AISB'06

Adaptation in Artificial and Biological Systems

Volume 3

Editors: Tim Kovacs and
James A. R. Marshall

University of BRISTOL

EPSRC
Engineering and Physical Sciences
Research Council

invent

Proceedings of AISB'06: Adaptation in Artificial and Biological Systems
Tim Kovacs and James A. R. Marshall (eds.)

ISBN 1 902956 97 5 – Volume 1
ISBN 1 902956 98 3 – Volume 2
ISBN 1 902956 96 7 – Volume 3

Published by the Society for the Study of Artificial Intelligence and the Simulation of Behaviour

Printed by the University of Bristol, Bristol, UK

Contents

Plenary Talks

Symposia

Narrative AI and Games

5th - 6th April 2006

Organisers

Sandy Louchart, Salford University
Judy Robertson, Glasgow Caledonian University

Programme Committee

Ruth Aylett, Heriot-Watt University
Paul Brna, University of Glasgow
Nicolas Szilas, Macquarie University
Stephane Donikian, IRISA
Marie-Laure Ryan
Daniela Romano, University of Sheffield
Marc Cavazza, Teesside University
Jonathan Schaeffer, University of Alberta
Sue Thomas, De Montfort University
Dan Sutch, Nesta Future Lab
Lisa Gjedde, University of Education
Dave Moffat, Glasgow Caledonian University
Michael Mateas Georgia Institute University
Laaksolahti, SISC
Craig Lindley, Gotland University
Stephane Bura, ElsewhereEntertainment

Contents

Edumarket game : Technocity

Julian Alvarez
IRIT/LARA
Toulouse II & III, FRANCE
alvarez@irit.fr

Olivier Rampnoux
CEREGE
Poitiers, FRANCE
olivier.rampnoux@univ-poitiers.fr

Jean-Pierre Jessel
IRIT
Toulouse III, FRANCE
jessel@irit.fr

Gilles Methel
LARA
Toulouse II, FRANCE
methel@univ-tlse2.fr

Abstract

The objective of this paper is to restore the process of development and installation of a project called "Technocity" which aims to develop technology courses of French State Education among the young people from 15 to 18 years old. We develop a gameplay to build a new school guidance tools. We choose to create a dynamic approach with video game and shortly report on different technical job. After, we explore the impact and the reception of this tool in different classrooms by childrens.

Keywords: **Advertising tools, Advertainment, Education, Innovation, Marketing, Reception, School guidance**

1 Introduction

Technical courses of study offered by the state education system, "Electro technical – Energy", "Electronics – IT", "System maintenance", "Mechanical engineering" and "Construction – civil engineering" find themselves in a dramatic situation in terms of attractiveness and recruitment as a result of their very poor image and reputation among young people. Even though there are real opportunities in terms of finding a job, attractive salaries and professions which are far from being routine, each year classes in the technological sector are being closed because there are insufficient numbers of candidates. Starting from this observation, the Sumotori communication agency has suggested to the state education authorities and the local education board in the Midi-Pyrenees Region (France) that they add to their communication plan and traditional tools through the development of a tool in order to boost the number of candidates and to improve the image of these courses.

"Technocity" is intended to participate in improving the image of five technological courses of study as seen by young people aged 15 to 18. The idea is to innovate in the communication strategy through the use of new technologies. Sumotori has suggested to the local education board to try to attract young people and to reinforce the impact of the message by using the principle of the video game as a communication mediator. The strategy is based on the idea of arousing interest and attracting the target with a recreational format and a medium which recalls their digital leisure activities and then encourages them to discover a message of a persuasive nature : discover and learn about the technological courses.

This experiment is a first for the local education board which has decided to give it a go and evaluate its impact. This choice has been driven by the urgency of the situation. The economic pressure on the school system is more and more significant in terms of performance and efficiency. "Technocity" combines in the same movement the recreational dimension and the informative dimension. In order to keep the attention for long periods, the video game is attractive and will offer mini reports at a later stage. During the game, the student will complete a quiz relating to the professions shown in the videos. This is this a radically new approach which seeks to combine the digital leisure activities of the child, videos and games. Given that our ultimate aim is to modify the child's attitude towards technological courses, we can talk of an advertising product. But, as the concept contains video games which require dexterity and experience of games, we can also position this product as entertainment. We see emerging an illustration of what is usually called advertainment. However, we must go beyond this term because, as a result of introducing into "Technocity" the quizzes which refer to the recreational educational space, we find ourselves in a more complex combination which brings together the stakes of entertainment, education and marketing. This innovative combination brings forth a new category of product which we could call Edumarket game.

Today, communication strategies already use video games to place products, the mobile phone to broadcast messages or ring tones which reproduce the sound associated with a brand. "Technocity" goes even further, because through this product, we are seeking to make video games compatible with a persuasive message which evokes studies and work. This antinomy is our major concern and is central to our thinking. Firstly, we will look at what is at stake in the development of "Technocity" and we will present the original facets of the product, and then we will present the results achieved in studies and inquiries carried out with children and teachers and representatives from the state education authorities.

2 Scenario development of Technocity

2.1. Games design

In order to develop the game scenarios, the first step was to list the "federating aptitudes" which are looked for in the child for each of the five predefined sectors. These "federating elements" are then associated with major existing families of video games (arcade, simulation, strategy) to determine the choice of gameplay for the five "Technocity" games. For example, in the case of "Mechanical engineering", whose "federating elements" are "spatial perception", "reflection" and adaptability", the principle of the game is an assembly game along the lines of "Tetris". We were guided in our choice by the idea that, beyond recognising skills, these games have to stand out in order to offer a choice to cover a widely as possible the different groups of players which make up the target.

Budgetary constraints and production deadlines played a large part in the development of the games. Developing a 3D video game for general release such as "Warcraft III", "The Sims" or "GTA", which could involve several hundred people over many months or years, requires a budget well in excess of 1 million euros. Moreover, video games for general release must make a profit which requires sales of thousands, or even millions of copies. In the present case, the means at the disposal of the education authorities are quite relative and make it necessary to consider cheaper development alternatives. We decided to develop a multimedia video game using programmes such as "Flash" (Adobe/Macromedia) which are simpler and quicker to use.

2.2. Style of graphics and sound in the games

The conscious choice of 2D, unlike the most recent video games on sale, aims at producing a break in perception. This type of contract can also be found in most small cartoon animations on Internet, "webtoons" such as "Happy Tree Friends", which push the concept to an extreme degree by depicting animals with personalities. Thus, a 2D graphic universe exists which is recognised and codified by children and we found it logical to use it as a source of inspiration. Moreover, as music and fashion are federating elements of identification for young people, the option of introducing variations with adolescent fashion models, associated with different styles of music also fits into the scheme of communal on-line practices. Each variation has its own music dedicated to its style of clothing and it accompanies the player through the different phases of the game

2.3. The video sequences in the mini reports

In order to present the various technical sectors, the local education board had several different video films. They present interviews with young technicians who describe their jobs for 2 to 3 minutes and explain exactly what they do. Presenting the message through explanatory videos constituted a difficulty both in terms of scenario and narrative in that the user has to move from an active state (interacting in the video game) to a passive state (watching video extracts) which creates a genuine break. Moreover, when the first video sequence is shown, we are at a crucial moment in the narrative structure, because the real intention of "Technocity" is revealed. In order to stimulate the attention and strengthen the recreational dynamic, a quiz with questions linked to the contents of the videos is proposed during the video sequences. A significant gain is made available in order to stimulate the memorization work of the student: access to new Flash games by a connection to 6 advertainments from the Dupuis's website www.spirou.com. The local education board is relying on word of mouth which might encourage young people to access games with well known enduring characters. This idea sends us back to the question of how "Technocity" will be received, to the formation of preferences which are going to have an influence on the attitude towards the professions presented and thus how it will affect the capacity to develop a behavioral intention for career guidance, which is the subject of the second part of our report.

3 Evaluation of Technocity

3.1. Research design

Distribution of the product takes place in two phases. First of all, through the intermediary of 1,000 CD-ROMs distributed in 600 colleges, secondary schools and career guidance centres in the Midi-Pyrenees region (France). Then an Internet site is set up with content identical to that of the CD-ROM (www.technocity.fr). The idea is that the site enables young people to continue exploring "Technocity"'s contents at home and can show it to their group of peers and to their parents.

In order to comprehend the reception of "Technocity", we have put in place a study design (Evrard et al, 2003) which aims at covering the entire target, but more particularly students in 3ème (grade 9), because it is a key part in career orientation. At the end of the college cycle many students can decide to opt for the technical sectors. First of all, we made participative observations with the students and interviews with education authority representatives and teachers in order to define a number of items and to be able to construct a questionnaire which will be put to a representative sample of the target population. The key idea is to assess the persuasive value of the concept thus developed on students after the orientation process through a comparison with pre-orientation statements, the perception of "Technocity" and the real orientation via the dossiers constituted by the students. Using the information thus collected, which will terminate at the end of the 2006 school year, we will be able to consider constructing a segmentation like that of Schaaper (1999) for the video game market. The identification of user profiles and the definition of a taxonomy of the students is necessary to enable the evolution of the communication tools during the next information and orientation campaigns put in place by the local education board. In this context, we considered it essential, in order to assess the reception of "Technocity", to meet all the different classes, while also taking into account the socio-geographical parameters such as the urban, peripheral and rural zones.

3.2. Results, attitude and reception

In the end, our results cover six reports of participatory observation in 8 classes, and 12 individual interviews with secondary school teachers and representatives of the local education board. Given that the entire study took place within a college or secondary school during sessions supervised by teachers, (we can consider that it is a kind of immersion) it is quite clear that this interacted with the reception of the product. Indeed, in the school setting, the child places himself in an apprentice situation and not in a play position. The institutional setting normalises the reception of "Technocity" and therefore requires distancing the speeches and the remarks made by the students. As a result, as the first sequence, after choosing a character, begins with a video game, the contrast with the school context produced a very positive and enthusiastic initial reaction from all the students. As Molinier points out, as a result of the labelling of the school around knowledge, skills and apprenticeships, and also a virtual refusal of the game in the school setting (Brougère, 1995), the recreational sphere appears as a contradiction, indeed a break with what the students expected, we are able to interpret this attitude in the light of the product.

As part of this research, we wanted to take a tripartite approach to the attitude and not a unidimensional approach. Indeed, by picking up on the work by Derbaix (1982) concerning the hierarchy of effects, which deal with advertising in particular, it appeared logical to us to seek to break down the attitude of the child into these three components, that is to say, the cognitive component, the affective component and the conative component. Knowing that the model which has been well-tested with children is an inverted sequence compared to the adult, we are in the presence of the following chain of events : affective – conative – cognitive. Thus, the initial enthusiastic reaction of the children relates to the affective dimension of the product. Moreover, as in the case of advertising in its traditional form, we know that the peripheral elements (music, characters, colours and graphics) play a key role in the construction of this affective dimension (Kapferer, 1985, Guichard, 2000). All the consideration, conception and development work carried out beforehand is justified by this initial reaction by the children.

In a second phase, during our observations, we noticed that playing video games followed by watching videos does not produce any weariness in the short term. Given the constraints linked to the school setting, our experimentation and observation protocols could not exceed one hour. In numerous cases, we had to interrupt the game or video because the students did not wish to stop it themselves. From this assessment, becoming familiar with the product has been achieved and raises interest. From the point of view of attitude, once again the product encourages support, and in particular in the behavioural component. Lastly, with regard to the cognitive dimension which sends us back to a better understanding of the professions and sectors, the fact that a large number of students were able to finish complete sequences (video games and quizzes on professions) leaves us to think that the adolescents have memorised the information, at least in the short term. The question of transferring the memory of work to the long term memory of knowledge was not part of the declared aims of the product.

The exploration of behavioral intention remains to be explored and understood. Currently, our results do not enable us to reach any conclusion regarding the impact and thus, efficiency of "Technocity" in relation to student orientation. This is the subject of the second phase of our research design which is currently being developed. If a real impact exists, we will have to test the hypothesis of knowledge memorization, of the modification of their vision of industrial professions, and the fact that students have made appointments with career guidance counselors or are seeking information about these professions. Lastly, an increase in the number of students in the classes previously out of favor may also be an indication of efficiency, but this would then require having control over all the variables interacting in this chain of events which leads to an actual enrollment in the training branch.

4 Conclusion

Our reflection will only be complete when we have tackled the question of student orientation in all its complexity. The mobilisation and development of new communication tools are not sufficient to guarantee the efficiency of the tool. Indeed, school orientation is also an eminently social process, in which parents seek to enrol their children on a trajectory which is in line with their social status and their representations of professions and sectors. Parents are strongly aware of the importance of school and training in order to ensure more rapid integration into the world of work in an extremely competitive context. As a result, the question arises concerning the choice of the target for state education communication campaigns, parents or children and hence the genuine efficiency of a tool such as "Technocity". School orientation is a syncretic decision for parents in which the children are closely associated. As a result by targeting secondary school students, we hope to exploit this double socialisation (parents – children and children – parents) or social interactions (Gollety, 1999). The question of the impact of "Technocity" and its reception by parents remains complete and does not come within our research. The other point to be analysed arises directly from the comparison between an 3D edumarkert game, like Food-Force (www.foodforce.com) and Technocity. Even though they use the same narrative plan, these two products do not make use of the same technologies (3D vs. 2D), and the influencing agent does not have the same legitimacy (United Nations vs. State Education) and they do not deal with the same themes. Having these two approaches available today makes us open to other study objects, to know if the different computer graphics approaches play a significant role in the reception of messages by young people, if edumarket game will open as many new paths to be exploited in commercial strategies for industrialists and marketing professionals.

References

Gilles Brougère, *Jouer/Apprendre,* Paris, Economica Anthropos, 2005.

Christian Derbaix, L'enfant, *la communication publicitaire et la hiérarchie des effets,* Recherches française de marketing,(89) p.31 – 47, 1982.

Mathilde Gollety, Lorsque parents et enfants s'apprennent mutuellement à consommer, *Décisions Marketing,*(18) p.69 - 80, 1999.

Stephen Kline., Nick Dyer-Witheford and Greig de Peuter *Digital Play. The Interaction of Technology, Culture and Marketing,* Montréal, McGill-Queen's University Press, 2003.

Tony Manninen, *Interaction Forms and Communicative Actions in Multiplayer Games,* Copenhague , Game Studies web site, 2004.

Pierre Molinier, *Enjeux d'une culture de l'image à l'école*, Paris, *An@é*,(2), 2000.

Petri Nets for Game Plot

Cyril Brom and Adam Abonyi
Faculty of Mathematics and Physics, Charles University in Prague
Malostranské nám. 25, Prague 1 – 118 00, Czech Republic
`brom@ksvi.mff.cuni.cz` `adam.abonyi@gmail.com`

Abstract

We have developed a technique for authoring a nonlinear plot and for managing a story according to the plot in an interactive story-based virtual reality application. The technique exploits Petri Nets that alter reactive plans controlling individual actors. Its main advantage is that it allows for iterative design and copes with large virtual worlds inhabited with tens of actors. In this paper, we describe the technique and a prototype application.

1 Introduction

This paper concerns itself with a problem of managing a story in an interactive application featuring a large virtual world inhabited by intelligent virtual humans. Above all, we think of a role-playing computer game underpinned by a narrative plot. A framework for driving virtual humans is already developed, as well as a prototype application for story management.

Several aspects need to be stressed. First, by *virtual human* (v-human in what follows) we mean a piece of software that *imitates* behaviour of a human in a virtual world and that is equipped with a virtual body. Even if the body is typically visualised by a graphical viewer, we will not consider graphical issues here. We are focused on *intelligent* v-humans, by which we mean that they carry out more complicated tasks than just walking, object grasping or chatting in an ELIZA-like manner.

Second, by *large virtual world* (v-world) we mean a really *large* artificial environment – not a single room, but a village or a region. There are tens or hundreds of v-humans acting in a large v-world. The story should last for tens of minutes.

Third, by *interactive* we mean that there is at least one user interacting with the simulation. Obviously, this will cause an interactive–narrative tension. The user can be embodied through an avatar, or can act within the v-world in an alternative way.

Fourth, a story is pre-given by a *plot specification.* The plot is not linear. It means that the story can evolve in several different ways according to a user intervention, and it can take place in several different locations at the same moment – not only in the location observed by the user.

We aim at developing a large long-lasting interactive computer game emphasising a story and intelligent artificial actors. From the artificial intelligence point of view, anyone with this goal must tackle at least the following issues:

1) How to control behaviour of an individual v-human?
2) How to simulate efficiently a large v-world when it is not possible to simulate it in its entirety due to enormous computational cost?
3) How to manage an interactive simulation according to a plot specification?
4) How to merge independent solutions of 1 – 3 into a single application?

We developed a toolkit for prototyping behaviour of individual v-humans (Bojar *et al.*, 2005) and, based on its concepts, a simulator of large v-worlds called IVE (*intelligent virtual environment*, Fig. 1) (Brom *et al.*, 2006). IVE presents an augmentation of Bryson's hierarchical reactive planning (2001), BDI theory (1987), and Gibson's theory of affordances (1979). These three main points help us to achieve several things. First, the creation of v-humans with relatively complicated behaviour is possible. Second, a level-of detail (LOD) technique for "AI" of v-humans and topology of virtual world (not for computer graphics) can be used; this allows for an automatic smooth simulation simplification on places unimportant at a given instant. Third, new objects and locations can be added into v-worlds as plug-ins and v-humans are able to adapt to them without using any machine-learning algorithm (similarly to The Sims). This feature facilitates a design.

IVE has been tested with a scenario comprising about 100 v-humans acting in four virtual villages; each with a pub, 5 mines, and 12 houses. The results have allowed us to conclude that the issues 1 and 2 have been solved. However, as the v-humans of IVE are driven only by prescripted reactive plans and schedules, no interesting story can emerge. Thus, the logical next step has been to address the issues 3, 4. In other words, our recent goal is to augment IVE with a story management module driving the simulation according to a high-level plot specification.

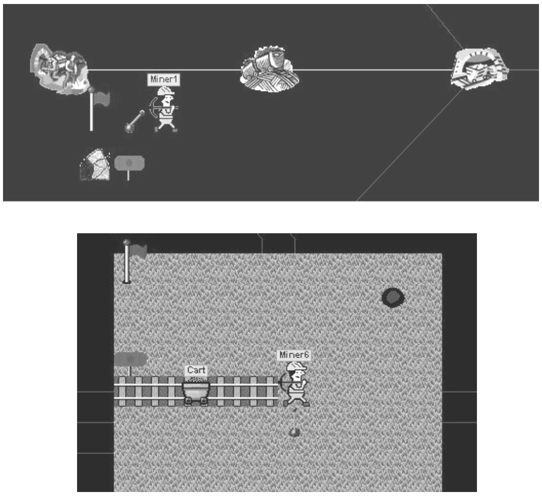


Figure 1: Mine scenario in IVE. The mine comprises three sublocations: the lower part, the tunnel and the upper part. There are two miners – one in the upper and one in the lower part. The lower part and the tunnel are not expanded (above), contrary to the upper part (bellow). The scenario illustrates an activity taking place in locations with different LOD value (*i.e.*, mining). The cart leaves the pit, which is simulated in less detail (LOD is 4), and enters the upper part, which is simulated in more detail (LOD 5).

Solving the issue 3 we have developed a technique for authoring the plots and for managing the stories, which uses Petri Nets. We have not fully augmented IVE with the technique yet, hence the issue 4 remains on the list. However, a test application (called TEST) for prototyping the plots and verifying the technique is already developed. In this paper, we describe the technique. We start with setting the ground of related works and detailing the requirements on the technique in Section 2. Section 3 introduces our technique, and describes TEST and a case-study story prototyped in TEST. In Section 4, a formal description of our method is given. Finally, we evaluate the technique and discuss directions for future research.

2 Related work and problem detailed

This section details requirements we had on the technique and discusses related work. With respect to our final goal, we were seeking a technique that:

a) copes with large v-worlds,
b) allows for describing high-level plot specifications, and yet allows for user interaction and for some degree of autonomy of v-humans,
c) copes with stories unfolding in several different places at the same instant,
d) allows for prototyping of stories.

The requirement b) means that we wanted a story manager to alter behaviour of individual v-humans, but not to drive them step by step. At the level of individual actors, we wanted the autonomy to be retained. Thus, our approach can be seen as a compromise between top-down scripting and emergent narrative. The requirement d) means that the method should allow a user to verify a plot in a test application that just *models* the course of the story. There should be no v-human and no 3D graphics in the application; it should serve just as a tool for testing the plot.

To be clear, we were not interested in automatic story generation, so the results of Cavazza (2002), who, generally saying, used an HTN planner for generating dramatically interesting sitcom-like episodes, is not what we were looking for. Similarly, we could not follow the approach of Aylett *et al.* (2005). In a nutshell, they used a continuous partially-ordered planner for generating episodes according to a high-level plot in an anti-bullying educational application. Notice, that we do not disregard the role of planners in storytelling applications. However, we think that it is hard to use them for large virtual worlds (issue a)) because of their exponential complexity.

We were interested, similarly to Mateas (2001), in authoring of plots and in keeping an unfolding story as close to an optimal plot as possible. Unfortunately, we could not use his solution, because it was not clear how to scale it to deal with the issues a), b), and d). Neither could we use the method of Szilas *et al.* (2003), because we did not find out how it could cope with issue a), b).

There is a branch of techniques used for specifying plots that exploits finite-state machines (FSM). Each state is a story episode, and a transition is a trigger that detects an end of the episode and starts a next one. FSM were described for example in (Sheldon, 2004) and used by Silva *et al.* (2003). Natural advantage of FSM is that they are formal, and yet graphical (Fig. 2), which facilitates a story design. However, a classical FSM is not suitable for us, since they cannot cope with issues c) and d). First, a (deterministic) FSM has just one active state at a given time, second, its triggers test for v-world events, which actually can not occur in a prototyping application for there is no v-world in it. Instead, we needed something like a FSM with more active states (to tackle c)) and with triggers that would have tested for events caused by the plot specification itself (not occurrences in a v-world – to tackle d). We found that this "something like a FSM" is a Petri Net (PN). They have been already used to describe story plots in (Natkin and Vega, 2003), however, only to a retrospective analyse of a story (of a computer game). We are interested in the reverse issue.

To recapitulate, there were 4 main requirements we had on the plot description technique. We found out that we could not use state-of-art work, especially FSM. We have realised that a PN similar to the PN used in (Natkin and Vega, 2003) is the solution. The method is introduced in Section 3 and detailed in Section 4.

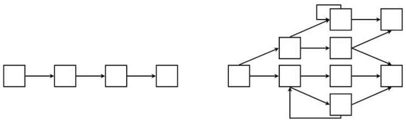


Figure 2: Story plots as finite-state machines. The linear plot is on the left, the nonlinear on the right.

3 PN-model and Story example

In this section, we introduce our technique, describe TEST and present an example of a story-plot prototyped in TEST. We recommend the reader to be familiar with PN – see (Natkin and Vega, 2003) for an introduction.

3.1 PN-model

To address the issues a) – c) (Sec. 2) we have refined a kind of PN, so called *timed coloured PN*, to serve for describing high-level plot specifications. To set the terminology, we say that *PN-model* is our type of PN, and *PN-plot* is a specification of an individual plot by means of the PN-model.

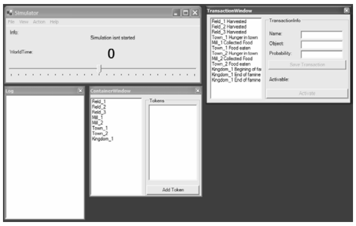
To address the issue d), we have developed two kind of PN-models, so called *draft PN-model* and *final PN-model*. The former serves for specification of a *draft PN-plot* and the latter for specification of a *final PN-plot*. Every draft can be verified by TEST, but can not be used for a real story manager. Every trigger of a draft PN-plot tests only for events caused by the draft itself, but not for any v-world events. Drafts serve for verifying the plots. Additionally, we have developed a method for converting a draft PN-plot to a final PN-plot. This conversion is to be carried out by the story designer after the draft is verified. Some triggers of the final PN-plot tests also for the occurrences of the v-world. The final PN-plot is aimed for a real story manager.

To summarise, our methodology for a specification of a plot is following: First, design a draft PN-plot. Second, verify the draft in TEST and adjust it. Third, convert the adjusted draft to a final PN-plot. In fact, this method is an example of iterative design. Section 4 details draft PN-model. Section 5 describes the conversion method.

3.2 TEST

We have developed a test application TEST, where a story is unfolded in an abstract manner according to a given PN-plot. TEST works as follows: First, a draft PN-plot is loaded from an XML-file. Second, when the simulation is started, abstract events start to occur according to the PN-plot. Third, a user can alter the story at real-time.

The purpose of the application is twofold. It serves as a "proof-of-concept" of our PN-model, and it allows for prototyping and verifying stories, including modelling of a user interaction. Actually, we are working on a real story manager of IVE.



Info:
Simulation isnt started
WorldTime: 0

Field_1
Field_2
Field_3
Mill_1
Mill_2
Town_1
Town_2
Kingdom_1

Field_1 Harvested
Field_2 Harvested
Field_3 Harvested
Town_1 Hunger in town
Mill_1 Collected Food
Town_1 Food eaten
Town_2 Hunger in town
Mill_2 Collected Food
Town_2 Food eaten
Kingdom_1 Begining of fa
Kingdom_1 End of famine
Kingdom_1 End of famine

Name:
Object:
Probability:
Activable:

Figure 3. A screenshot of TEST.

3.3 Story example

Here, we present an example of a story we have prototyped in TEST. Notice, that the story has no artistic value, it serves only for a demonstration purpose. The story is a simple fantasy narrative set in a village in an evening. It is a rather small story, just an episode from a larger tale. There are the following actors and groups of actors:

- MAGICIAN: a user-actor. She needs a puppet (for whatever reason).
- GUARDS: four bum-bailiffs. One of them is a friend of the magician; he has given her a note that there will be a puppet theatre coming this evening.
- PUPPET THEATRE: four artists. One of them is a twin of a guy who used to steal in the village several years ago, was arrested but managed to escape.
- ROBBERS: a band having pilfered in the village for a few weeks.
- VILLAGERS: citizens of the village. Some of them will mistakenly recognise the twin as his robber-brother.

Two things are going to happen in parallel. First, the robbers are going to pilfer this evening and the guards will try to capture them. Second, the troupe is going to perform a piece in a pub, and in the course of the play, some villagers will mistake the twin as his robber-brother and a brawl will flare up in the pub. Consequently, a fire might start, that would destroy the theatre and all the puppets.

Nothing is fixed, the events occur in a probabilistic manner. It is not sure whether or not the guards capture the robbers, or even notice them; whether or not somebody mistakes the twin; and whether or not the fire breaks out.

The magician is allowed to influence the story in the following ways: She can buy a puppet (before the theatre is reduced to ashes), or steal one in the course of the brawl. If she helps the artists in the brawl, she will be given a puppet for free, provided that no fire has broken out.

She can call the guards during the brawl using a spell. In this case, the guards interrupt the chasing and come to the pub. This reduces the probability of the fire breaking out significantly, especially if guards have just caught the robbers. She can also put out the fire immediately by casting a spell.

A part of the PN-plot for this story is depicted in Figure 4. The most important features of the portrayal are described in Section 4. The complexity of the drawing is discussed in Section 6.

4 Petri net model

This section gives the formal description of the draft PN-model and describes the most important features of the graphical representation of the PN-plot depicted in Figure 4. Notice however, that portrayals are informative only; what is important is a background formal specification. An algorithm for driving a story according to a PN-plot (both draft and final) is also presented.

A typical PN consists of places, tokens, transitions and transition function. In our model, we talk about containers (which are places – "the circles": ○, ◎, ⬡, ⯃), tokens (which are "the pellets": ●), actions (which are transition – "the rectangles": ▭) and triggers (more or less, they correspond to a transition function – to "the arrows": →, ⇢). The purpose of a trigger is to fire an action according to tokens' location in containers, or to add or to remove tokens from containers. Hence, tokens' locations evolve in time.

Tokens. Every *token* has a name, a colour, age, and a state. For every colour, there is a set of corresponding states. Let us denote T_{ALL} a set of all possible tokens, N_{ALL} a set of all possible names, B_{ALL} a set of all possible colours, and S_b a set of all possible states for a colour $b \in B_{ALL}$.

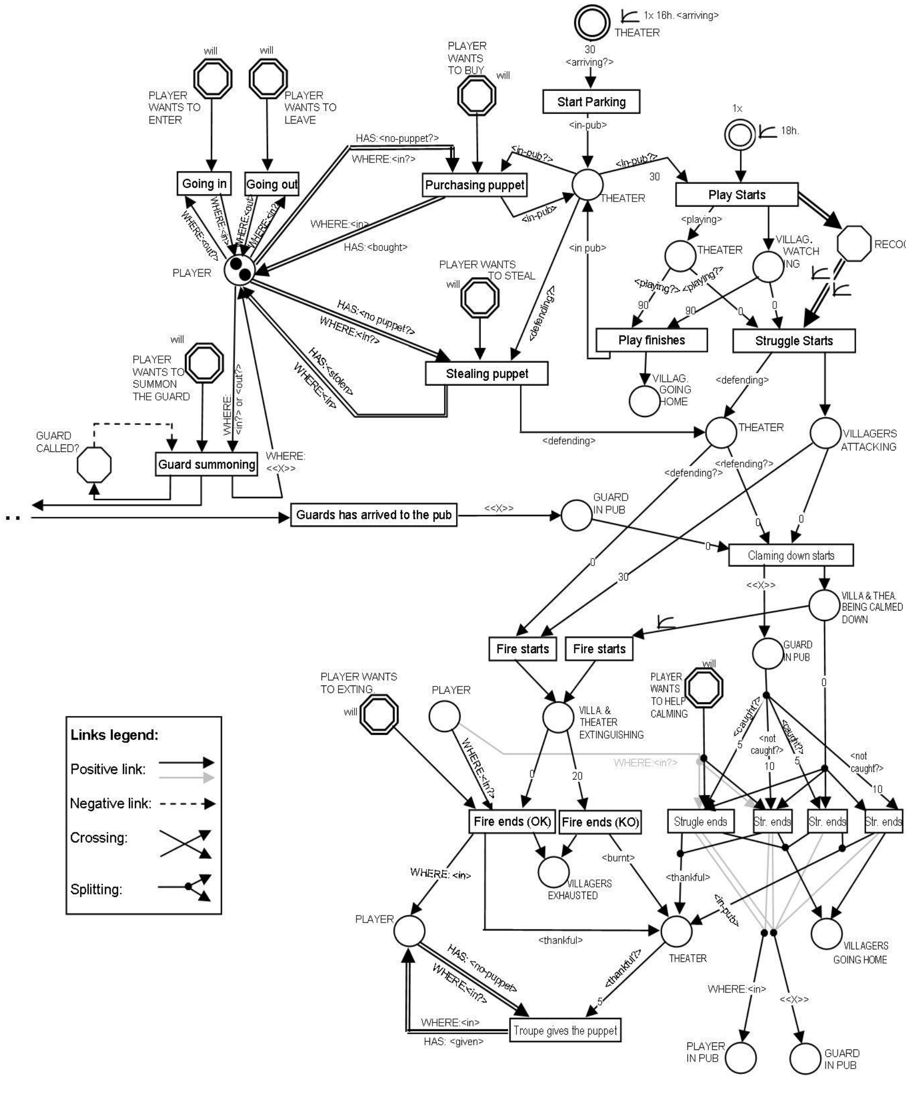


Figure 4: The "pub-plot" in the initial state is depicted. Notice that a plot can be much larger and several PN can run in parallel. In particular, in our case, the "catching-plot" is connected to the "pub-plot" on the left (indicated by "**...**").

Then, we say that *colouring* is a function: β: $T_{ALL} \rightarrow B_{ALL}$ – it gives a token's colour – and *token–status* is a function (defined for each colour): σ_b: $T_{ALL} \rightarrow N_{ALL} \times S_b \times \mathbf{N}$ – it gives a token's name, state, and age. If an S_b is empty, we say that the token is stateless. A token can be located in a container. The age stands for how long the token is located there. As Fig. 4 depicts the PN-plot before the simulation is started, only the initial tokens are shown. After the start, other tokens would begin to appear or could be removed.

Containers. Every *container* has a name, a type and can be associated with a set of triggers, which are actually if-then rules. Let us denote C_{ALL} a set of all possible containers, Y_{ALL} a set of all possible types and I_{ALL} a set of all possible triggers. Then, we say that *container–status* is a function γ: $C_{ALL} \rightarrow N_{ALL} \times Y_{ALL} \times \wp(I_{ALL})$ – it gives a container's name, type and triggers. A container can contain more than one token in a given simulation time. We say that *containing* is a function κ: $C_{ALL} \rightarrow \wp(T_{ALL})$ – it provides all tokens located in a given container in a given instant.

A semantic meaning of a token in a container is a denotation either of a state of a group of actors (*i.e.*, an actor-token) or of a satisfied precondition (*i.e.*, a prec-token). In a drawing, containers are denoted according to their triggers and tokens they can contain as ○ (actor-tokens/without any trigger), ◎ (actor-tokens/with triggers), ⬡ (prec-tokens/without any trigger), ⏣ (prec-tokens/with triggers).

There are following main token types in our PN-plot; *italic* denotes the initial state:

- "magician has" (colour m); S_m = {*no-puppet*, bought, stolen, given}
- "magician where" (colour i); S_i = {*in-pub*, out-pub}
- "guard" (colour g); S_g = {*not-caught*, caught}
- "theatre troupe" (colour t); S_t = {arriving, in-pub, playing, defending, burnt, thankful}
- "villagers", "robbers" (colour o); S_o = {}
- a prec-token: "has-called", "has-recognised" (colour o)

The meaning is obvious: for example, every "magician has" token represents that a magician has stolen, or has bought, or has been given a puppet, or does not have it, respectively. Similarly, the state of "magician where" stands for a magician in a pub or out of the pub. Notice, that these tokens can be located only in PLAYER container. Notice also, that a token is not a v-human itself, it is just a representation of its state for the purpose of a story management!

Triggers. The most important primitive of the PN-model is a *trigger*. A trigger can be associated with an action (*action trigger*) or a container (*container trigger*). Basically, a trigger is an *if-p-then-c* rule, where p is a precondition and c a consequence, which is to be performed when p holds. There are four types of triggers (both action and container).

- A *token-generating* trigger is a trigger that has a consequence of always adding some tokens to some containers and not removing any token.
- A *token-consuming* trigger has a consequence of always removing at least one token and possibly adding tokens to containers.
- An *action-firing* trigger neither generates nor removes any tokens, but fires an action.
- A *conflict-resolving* trigger's precondition tests whether two or more actions are to be fired at the same time, and its consequence resolves the conflict.

Actions and triggers. Every *action* has a name, an action-firing trigger, a token-generating trigger and token-consuming trigger, a "ready to fire" flag, and an effect. The precondition of the last two triggers tests whether the action has just fired (*i.e.*, whether or not the flag is set). Let us denote A_{ALL} a set of all possible actions. Then, we say that *action–status* is a function α: $A_{ALL} \rightarrow N_{ALL} \times \{0,1\} \times I_{ALL}^3$ – it returns an action's name, whether or not its flag is set, and its triggers.

If an action-firing trigger holds, its consequence sets a "ready to fire" flag. Then, the other two triggers can be triggered. Notice the word "can". In a given instant, more actions can be ready to fire, but not all of them can be allowed to fire for there can be a conflict between their token-consuming triggers (it is not possible to remove a token that has been just removed by another trigger). A conflict can be solved by a conflict-resolving trigger, which unsets the flag. Finally, when an action is really allowed to fire, its effect is performed, and its token-generating and token-consuming triggers are triggered. See Algorithm 1 below for details.

There is no graphical primitive corresponding to a trigger. However, a container with a trigger is denoted as ◎ or ⏣, and an action trigger is depicted as a set of arrows, each from a container to an action, or vice versa. For example, ⊙→▭ denotes an action-firing trigger and a token-consuming trigger, which means "try to fire the action if there is at least one token in the container" and "consume one token when the action is fired". An example of a token generating trigger is ▭→○ (it means "generate one token when the action is fired").

Details of a precondition or a consequence are indicated schematically next to an arrow, next to a container, or by a changed shape of the arrow. Precisely, a precondition can:

Algorithm 1. PN-plot driving in TEST and in a real drama manager.

Input: PN-plot.

The algorithm is performed in every time step *t*.

```
1) $I_{action,t}$ ← all action-firing triggers that fire in time $t$
2) Perform a consequence for all $i \in I_{action,t}$ in a random order (i.e.,
   mark the "ready to fire" flag of the respective actions)
3) $I_{conflict,t}$ ← all conflict-resolving triggers that fire in time $t$
4) Perform a consequence for all $i \in I_{conflict,t}$ in a random order (it
   unsets the "ready to fire" flag of some conflicting actions; it
   may include asking a user for selecting the action)
5) $I_{remove,t}$ ← all token-consuming triggers that fire in time $t$ (they
   test for "ready to fire")
6) for each $i \in I_{remove,t}$ (take $i$ in a random order) do {
       if all desired tokens can be still removed do {
           perform the consequence of $i$ (it includes removing the
           tokens)
           if $i$ is an action trigger do perform an effect of the action }
       otherwise,
           if $i$ is an action trigger do unset the "ready to fire" flag }
7) $I_{generate,t}$ ← all token generating triggers that fire in time $t$
8) for each $i \in I_{generate,t}$ (take $i$ in a random order) do {
       perform the consequence of $i$
       if $i$ is an action trigger do unset "ready to fire" flag }
```

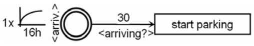
- test κ: *i.e.*, whether some containers contain (⟶) or do not contain (⋯⋯➝) some tokens; test β and σ_β: *i.e.*, colouring (<col?>) and token-status (<state?>),
- test simulation time (HH hours),
- compare age (MM minutes) or simulation time (HH hours) to a random value generated by a given probabilistic distribution (),
- test whether a user will alter the simulation somehow (will),
- test whether a trigger has fired *n* times (n ×),
- test whether two or more actions are in a conflict (======).

A consequence of a token-generating trigger can generate a token of a specific colour and a state. It is indicated as <state>, <<X>> or <col> above the arrow. <<X>> means that the generated token has the same state as the token that has been just consumed (see, for example, GUARD IN PUB container).

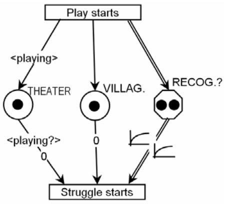
Notice that all abovementioned acts relate to the draft PN-plot itself, no v-world events are tested. This is not possible with any FSM. As TEST only models the course of a story, the effects of all actions are user notifications of an action performance.

Notice that not all conflicts are always solved in Step 4 – there might exist a conflict not recognised by any trigger. In this case, the action to execute will be chosen randomly from the set of the actions in the conflict (Step 6).

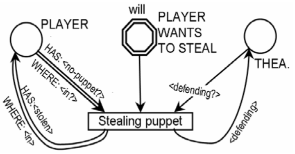
Now, consider several examples taken from Figure 4. First, focus on the container THEATRE and the action *Start parking* (in the pub). There are three triggers indicated in the drawing. The first one generates a token "theatre" with arriving state into the container once at about 4 p.m. (Step 7, 8). The second one is an action-firing trigger. It sets "ready to fire" flag when the token in the container is at least 30 minutes old (Step 1, 2). The third one is a token-consuming trigger, which removes "theatre" token when the action fires (Step 5, 6).



The next example concerns with the episode of the theatrical performance. The token-generating trigger of the *Play starts* action has generated four tokens; two "has-recognised" into the RECOGNISED? container, one "villagers" into the VILLAGERS WATCHING container, and one "theatre" with the state playing into the THEATRE container. The action-firing trigger of the *Struggle starts* action tests whether or not there is a token with the state playing in the container THEATRE, and a token in the container VILLAGERS, and two tokens in the container RECOGNISED? with appropriate age. It is determined randomly for the both tokens, whether or not the token is old enough. If the action is fired, the next trigger – the token-consuming one of *Struggle starts* – consumes all four tokens. Notice that all durative episodes are represented in a similar way using the age of the tokens. The container THEATRE is the same as in the previous figure, it is just depicted several times for convenience.



The next drawing represents the stealing of a puppet. If the magician is in the pub (denoted as WHERE: <in?>), if she does not have a puppet (denoted as HAS: <no-puppet?>), and if she wants to steal, and if the brawl has already flared up (the state of "theater" is defending), she may steal the puppet. Notice that a token "player wants to steal" is generated according to a choice of a user – this feature is achieved by checking a check box in TEST now. From the perspective of the high-level plot, the action of stealing is instantaneous: it takes no time.



Notice that there is no conflict-resolving trigger in PN-plot of this story. Notice also that all the properties that are only indicated in the portrayal, such as probabilistic distributions, are specified precisely in the XML plot specification.

5 Supplementing IVE with a story manager

In this section, we describe how to convert a draft PN-plot to final PN-plot. We first remember IVE. Then we extend the draft PN-model specified in Section 3 to the final PN-model. Finally, we describe how to convert a draft to a final PN-plot.

In IVE, almost all behaviour of v-humans (say, 90%) is driven by hierarchical reactive planning, in particular by production rules organised in tree-like structures (Brom *et al.*, 2006)[1]. There are two reasons for that: the reactive planning fits well in a large, unpredictable dynamic v-world, and hierarchical nature of plans helps us to achieve LOD simulation. A classical planning technique needs to be exploited only rarely (a case in point is a path-finding).

In order to extend the draft PN-model to the final PN-model, we need to specify how a story manager can alter a v-world and how a v-world can modify a final PN-plot. The extension is founded accordingly. First, the story manager can alter the story by performing an effect of a PN-action. In the draft, the effect is just a user notification. In contrast, in a final PN-model, an effect is allowed to:

1. create a new virtual object, a location or a v-human,
2. force a v-human upon a new reactive plan, hence modify its behaviour,
3. modify an internal drive of a v-human, hence change its needs,
4. increase or decrease LOD in a given part of a v-world.

For example, in Fig. 4, the effect of the action *Struggle starts* would be the case 2. Second, we allow preconditions of triggers in the final PN-model to test also circumstances of v-world. For example, each trigger of "will" would correspond to an event caused by a user actor.

Having the final PN-model, we can describe how to convert a draft PN-plot to a final PN-plot. Suppose we have a draft PN-plot of a story episode.

1. Replace every precondition of a trigger of "will" from the draft by a final precondition that tests whether or not a user has just caused a respective event.
2. Replace every effect of an action beginning the story from the draft by an effect from the list of possible final effects (above).
3. Test in the final application, whether the story episode begins as expected.
4. From the draft, take all action-firing triggers, conflict resolving triggers, token-generating triggers and actions not beginning the story. Take them one by one in the order of their presupposed time of firing: if it is a trigger, ponder on whether its precondition should be replaced; if it is an action, ponder on whether its effect should be replaced. If it should, replace it.
5. Test the final application.

Since we do not have a final story manager yet, we have converted some of our draft PN-plots verified in TEST (including the example from Sec. 3) to the final PN-plots by hand. Notice that this procedure is intended to be automated partially.

We have realised several things. Preconditions testing an age of a token are usually replaced by a test of occurrence of an event in a v-world. Typically, preconditions of token-generating triggers of containers are replaced. Token-generating and token-consuming triggers of actions need to be modified only rarely. In the example from Sec. 3, about 25% of preconditions from the draft were replaced.

6 Discussion and Future work

Although our work is in progress, the results we have collected so far show that our model allow us to achieve the four requirements from Sec. 2. Notice that for a story comprising several episodes, more PN-plots can be specified and all of them can run in parallel. The magician, for example, does not have to be in the pub for the brawl to flare up. She can be involved in the "robbers catching" episode (that would decrease LOD value in the pub). Notice also that while the story manager alters the course of the overall simulation from a high-level perspective, the autonomy from the point of view of individual v-humans is retained. V-humans act according to their reactive plans, which are only "attuned" to the story. Our approach presents a compromise between pure scripting and pure emergent narrative. We have also realised that it is extremely useful to have an application for a story prototyping. During the creation of a plot one can make a flaw. In TEST, the flaws can be detected easily. There are also several natural advantages of PN. They can be formally validated (which helps to detect unreachable parts for example), depicted graphically and compiled at runtime.

There are also several drawbacks of our approach. First, our tokens are state-based. That means a token can have exactly one state: *e.g.*, "magician" token can have stolen state, but not both stolen and, for example, in-pub state – we need two tokens. Tokens also can not hold variables. These are merely technical limitations. We did not need these features for our plots and therefore we did not specify them in the formal model. In principle, it is possible to formalise a PN-model so that the features are included.

A more inherent problem is that a portrayal of even a small story (see Fig. 4) is fairly complicated. Assume that there are tens of such episodes; how

---

[1] Notice that there is a distinction between pure reactive agents employed e.g. by Brooks (1986), and v-humans driven by reactive planning. The latter can for example exhibit goal-directed behaviour and use their memory. A v-human driven by a reactive planner is not "intelligence without representation". Notice also, that a reactive plan, even though pre-given, is not a computer game script.

could a story designer cope? We think that one possible approach to this problem is to extend the PN-model in a hierarchical manner. Microsoft Visio (2006) might be a tool for drawing of plots in this way. The advantage of Visio is that one can define in it new graphical components and routines for their conversion to XML specifications. These specifications are needed both for TEST and a real story manager.

Our future work concerns development of a real story manager and addressing the problem with large portrayals using a neat drawing tool. We remark that Algorithm 1 is designed to drive both TEST and a real story manager.

7 Conclusion

In this paper, we have described a method for specifying non-linear plots for a large computer game featuring intelligent virtual humans. The technique exploits Petri Nets specifications. It is designed to cope with four issues: with large virtual worlds, with stories unfolding in several different places at the same time, with allowing for some degree of autonomy of v-humans and with allowing for prototyping of stories. The natural feature of the technique is that the plots can be depicted graphically.

We have briefly described our application called TEST that serves to verify plots. We have also introduced IVE, which is our AI framework for programming large virtual worlds inhabited with tens of actors. Our future work concern supplementing IVE with a story manager that uses our Petri Nets and that is driven by the algorithm described in this paper.

Although the technique is aimed for IVE primarily, it can be used in another application as well, in particular in an application featuring a large v-world. TEST can be used independently as it is. Both TEST and IVE can be downloaded at http://urtax.ms.mff.cuni.cz/ive/.

Acknowledgements

This work is partially supported by the Program “Information Society”, the project 1ET10030051; and GA UK No. 351/2006. The authors would like to thank to Joanna Bryson, Lubomír Bulej and two anonymous referees for their valuable comments.

References

Ruth S. Aylett, S. Louchart, J. Dias, A. Paiva, M. Vala. FearNot! – An Experiment in Emergent Narrative. *Proceedings of Intelligent Virtual Agents*, 305–316, 2005.

Ondřej Bojar, Cyril Brom, Milan Hladík, Vojtěch Toman. The Project ENTs: Towards Modeling Human-like Artificial Agents. *SOFSEM 2005 Communications*, Slovak Republic, 2005.

Michael E. Bratman. *Intention, plans, and practical reason*. Cambridge, Mass: Harvard University Press, 1987.

Rodney A. Brooks. Intelligence without reason. *Proceedings of the 1991 International Joint Conference on Artificial Intelligence*, Sydney, 569–595, 1991.

Cyril Brom, Jiří Lukavský, Ondřej Šerý, Tomáš Poch, Pavel Šafrata. Affordances and level-of-detail AI for virtual humans. *Proceedings of Game Set and Match 2*, Delft, 2006 (to appear)

Joanna Bryson. *Intelligence by Design: Principles of Modularity and Coordination for Engineering Complex Adaptive Agents.* PhD thesis, Massachusetts Institute of Technology, 2001.

Marc Cavazza, Fred Charles, and S. J. Mead. Interacting with Virtual Characters in Interactive Storytelling. *ACM Joint Conference on Autonomous Agents and Multi-Agent Systems*, Bologna, Italy, 318 – 325, 2002.

James J. Gibson. *The Ecological Approach to Visual Perception.* Boston: Houghton Muffin, 1979.

Michael Mateas. *Interactive Drama, Art and Artificial Intelligence.* Ph.D. Dissertation. Department of Computer Science, Carnegie Mellon University, 2002.

Microsoft Corporation. Microsoft Office Visio 2003 *www.microsoft.com/office/visio/* [14th January 2006]

Stéphane Natkin and Liliana Vega. Petri Net Modelling for the Analysis of the Ordering of Actions in Computer Games. *Proceedings of Game-ON*, 82–92, 2003.

Lee Sheldon. *Character Development and Storytelling.* Chapters 7 and 14. Thompson Course Technology, 2004.

André Silva, Guilherme Raimundo, and Ana Paiva. Tell Me That Bit Again... Bringing Interactivity to a Virtual Storyteller. *Proceedings of Virtual Storytelling II*, 146–155, 2003.

Nicolas Szilas, Olivier Marty, Jean-Huges Réty. Authoring highly generative Interactive Drama. *Proceedings of 2nd International Conference on Virtual Storytelling*, 2003

Explicit Intent:
Shared Story Ownership between Player and Game

Stéphane Bura
Creative Director
Elsewhere Entertainment - 10Tacle Studios Belgium
sbura@pobox.com

Abstract

Per definition, Interactive Storytelling in videogames requires interaction, that is, communication emanating from both the player and the game. However, in current games, the player's ability to express his intent is severely limited, leading to poor control over storytelling. This article presents a different way of envisioning this interaction, one that is inspired by tabletop role-playing games.

1 Introduction

If interactivity is "the capability of acting on each other"[1], then the stories found in the current generation of videogames are not interactive. The player can trigger and sometimes select the events that compose these stories, but he can't act on the stories or affect the quality of the experience they offer.

This stems from two mental blocks that plague game designers and writers: That the player's avatar should be his sole mean of communication with the game and that the avatar's successes and failures are equivalent to the player's. Both these blocks are the consequences of a laudable goal: to increase the player's immersion in the game world as well as his empathy toward his avatar and, in effect, to emulate movies. But games are not movies. They are an interactive medium.

Caveat: The author is not a research scientist and this is not a scientific paper; these are just thoughts from a game developer who has been working on the problem of interactive storytelling in videogames for ten-odd years.

2 Game Patterns

While repetition is only used for stylistic purpose in non-interactive stories, it is at the heart of games. Indeed, Raph Koster (2005) argues that the opportunity for the player to learn patterns in a safe environment is what makes games fun. Learning can only occur if similar situations present themselves regularly and if the means to influence them can be understood and mastered. Such a situation and the interactions it affords is what we'll call a *pattern*.

2.1 Patterns limitations

The number and variety of patterns available in a videogame depends on the number of commands the player has access to and the meaning they can take depending on the context. For instance, shooting at a creature conveys the intent to harm it, while shooting at a lamp can mean trying to break it and make the area dark. Learning patterns and their uses is important since succeeding at game challenges usually require a certain level of mastery in that regard.

A player can only remember and master a limited number of commands and original patterns, that is patterns specific to the game world (such as the effects of a spell or piece of equipment never encountered before, in any other medium). This is why most games rely heavily on patterns taken from the real world or on well established game patterns (like the health pack) and why they use generic commands, like "shooting at something" in the example above. Such commands, therefore, can't convey meaning by themselves: only the patterns do.

The narrative artificial intelligences in the current generation of story-based games all implement the same interaction loop: the player acts, then the AI interprets what the changes brought to the game world by these actions mean and, finally, changes the game world too to reflect the inferred meaning or illustrate plot advancement, which can open new actions to the player. Inferring meaning from the use of patterns by the player is a complex task because, as we've seen, the player's vocabulary is fully action-oriented. Did the player mean to shoot at the

---

[1] As www.dictionary.com defines it.

foreign dignitary across the park or was it the result of a random shot? Does he want to strike fear in the villagers by walking among them fully armed or did he forget to put his weapons back in his inventory? There's no way for the game to know. So games usually assume that any occurring event linked to a recognisable meaning is purposely produced by the player (yes he tried to assassinate the dignitary, set the secret services on him; yes he wanted to scare the villagers, let the fearful healer hide). This can be very frustrating for the player because, except if the game explicitly warns him beforehand, he has no way of anticipating which of his actions convey implied meaning and which don't[2]. To minimize this frustration, game designers tend to limit implied meaning to actions that have easily understood quasi-mechanical effects linked to game creatures' senses and behaviours: The player walks slowly *in order to* move silently and remain undetected by the guards (Thief, 1998), or he makes a noise *in order to* draw a guard to the place where the noise was heard (Metal Gear Solid, 1998). There is meaning here but it is canned functional meaning.

Since the player can only use action-oriented patterns to express meaning[3], the narrative AI interaction loop becomes quite simple: it just checks if predetermined patterns are used or conditions met to trigger story events (for instance, when the dungeon door is opened; when the dragon is killed; etc.) The player is thus taken from one story point to another, living out the story but having no control over it. Action Adventure games use this method exclusively, usually allowing parallel story threads (or side-quests) to give back some control to the player (he can, for instance, try to find a lost dragon-slaying sword before confronting the dragon or try his luck without). Triggered events are not about understanding what the player wants to do; they just add a narrative frame to his actions. Such stories are not interactive.

2.2 Storytelling Patterns

Nowadays, the most successful models of interactive storytelling are to be found in tabletop Role-Playing Games (RPGs)[4].

Thirty years ago, RPGs started where story-based videogames are now: players did little more than live out pre-written adventures, their only freedom being the choice of their moment-to-moment actions. But RPGs quickly evolved, each new game bringing innovations such as eschewing the adventure model and replacing it with a fully developed world the players could explore and change (Steve Perrin, 1978); giving the players the means to control fate and luck (G. C. Klug, 1983); having the story affected by characters' passions (Greg Stafford, 1985) or flaws (Steve Jackson, 1986); encouraging all the players in the group to take turns as the storyteller (Jonathan Tweet, 1988); or even driving inexorably the characters toward madness and death (Sandy Petersen, 1981). RPGs continue to evolve today, still experimenting with form, rules, and themes. Yet computer games cling to a thirty year old model.

The evolution of RPGs follows a clear path: each new generation of games gives more narrative powers to the players, powers that were previously the sole province of the Game Master. Sharing story ownership between the players and the storyteller increases the quality of the narrative experience as the story meets the needs and desires of the players more often and more easily. In order to achieve this shared ownership, RPGs define rules about how players can affect the story beyond the actions of their characters. These rules are *storytelling patterns*.

3 Explicit Intent

Sharing story ownership means that the player is no longer a simple actor. He can change the game world beyond what his avatar can do and then slip back in his actor role and deal with or profit from the new situation. This is a huge paradigm shift.

Videogames are often conceived as a series of challenges presented not only to the player's avatar but also, through it, to the player. The player is thought of as playing *against* the game, his avatar's victories and defeats becoming his. But what if a defeat generates a more interesting story? What if some victories feel pointless and boring to the player (as can victories in random repetitive combats)? What if the player feels that only one outcome to a situation is interesting, however improbable or difficult it is to obtain? Shouldn't the player be allowed to tell the game about it - to play *with* the game - so that the perceived problem can be fixed?

---

[2] It's not reasonable to believe that they all do. Actually, the more implied meanings - and thus the more realistic the game world is - the more jarring inconsistencies appear to the player.

[3] Pre-scripted modal dialog with game characters can convey meaning. However, since it doesn't use sophisticated AI (most game dialogs are explorations of a dialog tree or graph) but instead relies on writing quality and choice variety, it is beyond the scope of this article.

[4] In a RPG, each player except one improvises the part of a character in a story. The remaining player is the Game Master (or storyteller). His role is to describe the world where the story takes place and the result of the characters' actions. He also plays the parts of all the other characters in the world (the "supporting cast"). The goal of a RPG session is not to win the game but to have a pleasurable storytelling experience.

3.1 Harnessing the Narrative Power of the Player

James Bond 007 Role Playing Game (G. C. Klug, 1983) presented its designer with a challenge: what kind of game rules could simulate the world accurately and still allow for the improbable stunts and lucky breaks the Bond-like agents need in order to survive and accomplish their missions? He solved the problem with the *Hero Points* storytelling pattern. Hero points are a limited resource players can spend to emulate Bond's luck and extraordinary skills. Hero Points let an agent find a sport car with its motor running just before a car chase scene; or shoot (without any chance of missing) an underling who is about to activate the villain's base self-destruct mechanism; or jump unscathed through a hail of bullets; and so on. Hero Points make the game more interesting as taking risks became fun, just like in a James Bond story.

A player in a videogame could have the same ability to bend the world rules when he feels it will make the story more interesting. This could include skewing the odds of random events (perfect skill use, random combats avoidance, etc.), finding an object his avatar needs, having it meet a character "by chance", etc. Conversely, the player could create problems for his avatar (jammed gun, additional guard patrols, sneezing bout, etc.) to increase the dramatic tension or the challenge level.

In such a model, since the game knows the purpose of the events chosen by the player, it can present their outcomes in a dramatic manner (the game sends a patrol near the avatar just when it's about to sneeze) or in the form of new challenges (for instance, the dungeon key wished for by the player appears dangling from a guard's pocket and the imprisoned avatar must grab it without being detected when the guard walks past its cell).

3.2 Story Contexts

While the above storytelling pattern allow the player to tell the game what he wants, *story contexts* let it know how the avatar feels and what the player's intent is.

A story context describes the current emotions, feelings or world views of the avatar and, through them, the player's goals. Using this information, the game can alter existing interactions, make new ones available and remove old ones. In current games, action-oriented patterns are routinely modified according to the avatar's physical state - a wounded avatar would move slowly and with a limp - but rarely according to its mental state since it's supposed to be the same as the player's[5].

Story-based games where the player can explicitly choose the emotions felt by his avatar are rare. Indeed, emotions are often used to get the avatar moving toward the next plot point ("They killed my partner. I won't rest before I avenge him."), so letting the player control them could ruin the storyline. Again, this mental block requires a paradigm shift. These games aim for passive player empathy (let him feel happy / sad / vengeful / afraid along with his avatar) instead of letting him actively shape the emotional landscape of the story. Choosing feelings and emotions for one's avatar could be a result of empathy or projection, but it could as well be motivated by curiosity or strategic thinking ("I won't let anger fill my heart and render me blind to the true reason why my partner died").

Used that way, emotions / story contexts can themselves trigger story events or be illustrated by the game to reinforce the chosen mood. For instance, let's say the avatar is alone at night in a deserted back alley. Two of the available moods are *nervous* and *confident*. If the player chooses to make the avatar nervous, the game reflects this by showing him the world *the way his avatar perceives it*[6]: its footsteps sound louder than usual, there are whispering winds, curls of mist, howling dogs in the distance, menacing shadows, and glittering eyes in the darkness. A nervous avatar would attract the attention of some low-life and would get mugged. Conversely, the player of a confident avatar could see a brightly lit alley, rats fleeing before him, and pass through it unchallenged.

Why then would the player want his avatar to be nervous? Is that the "wrong" choice? This illustrates the difference between playing against the game (trying to win every challenge) and playing with it to build an interesting story. Plot complications are interesting. Maybe the mugging would lead the beaten avatar to a hospital and start a romantic subplot with a compassionate nurse; maybe the avatar would overpower the mugger and acquire information from him or gain status in the underworld.

3.3 Explicit Player Goals

In the dead partner example above, wouldn't the experience be more intense to the player if he chose to seek revenge, instead of being told by the game that he had to? Choosing one's goal can also be done with the help of story contexts.

---

[5] For proof, the current trend of output alteration depending on the avatar's state, like blurring the screen when it is dazed.

[6] Subjective viewpoint is an effective technique to enhance empathy or suspense. It is used in countless movies.

Story contexts have different scales and can be nested. They govern the short term (emotional reaction to a situation that can lead to opportunities, as above), the mean term (lasting moods that can alter character relationships), and the long term (goals, values, beliefs). Rules link them logically or dramatically. For instance, a string of small victories can make the *optimist* mood available that, if chosen, augments the avatar's luck but can lead to sudden dramatic reversals of fortune. Or, if the player lets his avatar get angry, it could augment its strength and resilience but close the access to other possible interactions (like negotiation) and require special events to exit this mood (like a retreat in a monastery).

This opens a new emotional space to explore, beyond the physical space and the story timeline.

Using these different types of context, the player can inform the game of what he wants now, in the near future and what the global experience should be about. Is it a story about a character seeking revenge or seeking justice? Is it about attaining a material goal or preserving a relationship? Is it one of noble sacrifice or one of survival against all odds?

If the player were able to make such a choice early in the game, it could colour the whole experience.

4 Implementation

Making such games is hard because they must be designed and implemented holistically. Action-oriented patterns (the rules of the game world) and storytelling patterns (how to cheat) are interdependent and determine the kind of stories that can be told and, thus, also the kind of suitable game world.

In order for the player to use storytelling patterns, the relevant rule parameters (such as the probability of random attack by monsters) must somehow be known and accessible to him. In the case of object "creation" or character behaviour alteration, each potential event must be pre-scripted separately or follow precise rules. To ensure that these events can be combined, their game descriptions should use shared concepts and functionalities (like guards reacting to "localized noise" and not only to a sneeze, a cough, a snapped twig, etc.). This requires a well developed library with width (a number of interesting concepts without significant overlaps), depth (detailed behaviours that use these concepts), that can be used to describe all the possible interactions between objects in the world and that contains no inconsistencies. Furthermore, story contexts must be able to alter the characteristics of these events and concepts dynamically, as well as other contexts.

This means that *one cannot develop such a narrative AI outside of a specific game*. The designer has to make choices regarding the kind of experience he wants to provide the player with. These choices will limit the possibilities of interaction with the game world and with the story and decrease the design's complexity.

For instance, the game could focus on exploring the storytelling possibilities of a confined world (à la *Groundhog Day*); on James Bond-like rule bending (which would require a simple enough world to be fully described, such as the ones in the early Zelda games - like *The Legend of Zelda*, 1987); on letting the player set his own goals in a complex evolving world (with the dynamic creation of appropriate challenges); etc.

Conclusion

Current videogames give the player an interaction toolset that is too limited for him to express his intent. If the goal of interactive storytelling is to allow player and game to create together a pleasing narrative experience, such a game must share ownership of the story with the player. This means that the perception of what a player does in the game has to change. He should no longer be confined to playing the part of his avatar. Instead, he should be considered as a performer as well as a part-time director, writer, critic and prop master. The game should adapt the experience to fulfil the player's clearly expressed wishes - while still offering him challenges and surprising him. Videogame storytelling will then truly be interactive.

References

Steve Jackson, *GURPS*, Steve Jackson Games, 1986.

G. C. Klug. *James Bond 007 Role Playing Game*, Victory Games, 1983.

Raph Koster. *A Theory of Fun*, Paraglyph Press, 34-47, 2005.

The Legend of Zelda, Nintendo, 1987.

Metal Gear Solid, Konami, KCEJ, 1998.

Steve Perrin & al., *Runequest*, Chaosium, 1978.

Sandy Petersen & al., *Call of Cthulhu*, Chaosium, 1981.

Greg Stafford, *Pendragon*, Chaosium, 1985.

Thief: The Dark Project, *Looking* Glass, Eidos Interactive, 1998.

Jonathan Tweet & al., *Ars Magica*, Lion Rampant, 1988.

Individuality and Contextual Variation of Character Behaviour for Interactive Narrative

M. Gillies*
*Department of Computer Science,
University College London
Malet Place, London WC1E 6BT, UK
`m.gillies@cs.ucl.ac.uk`

I. B. Crabtree†
†BT plc
Adastral Park, Ipswich IP5 3RE, UK
`barry.crabtree@bt.com`

D. Ballin‡
‡BT plc
Adastral Park, Ipswich IP5 3RE, UK
`daniel.ballin@bt.com`

Abstract

This paper presents a system for generating non-verbal communication behaviour suitable for characters in interactive narrative. It is possible to customise the behaviour of individual character using a system of character profiles. This allows characters to have a strong individuality and personality. These same profiles also allow the characters' behaviour to be altered in different contexts, allowing for suitably changing behaviour as the story unfolds.

1 Introduction

Characters are vital to narrative and their behaviour is central to expressing the unfolding story. Their actions play a major part of creating a narrative but other, more subtle, behaviour is also important. Non-verbal communication (also known as "body language") plays a large part in defining characters and displaying the mood of a scene. In this paper we present a system for generating non-verbal communication behaviour suitable for interactive narrative.

Diversity of characters is vital to creating stories, the interplay and conflicts between contrasting personalities is one of the most important elements composing narrative. The differences between characters must be clearly visible in their behaviour, and non-verbal behaviour is one of the most important expressions of personality. As such, it must be possible to customise characters, giving each their own specific mannerisms and behaviour. However, characters should not always act in the same way. A key element of narrative is that as the story unfolds the characters' situation changes and in particular the emotional tone of the story alters. This means that the behaviour of the characters should be able to change to express their new situations. This can happen in many ways. For example, characters should behave differently depending on who they are interacting with, and the relationship between them. Characters can also take on different goals, and different behaviour is appropriate in different places and situtations.

We present a system of character profiles that both allows end users to customise characters, and also allows the characters' beahviour to change in different narrative situations. The profiles system is build on the demeanour framework. Demeanour is a system for generating non-verbal communication. It contains a behaviour language that is used to author autonomous behavioural controllers. These controllers contain a number of parameters that can alter the behaviour of a character.

2 Related Work

Our work builds on a body of work on autonomous characters for virtual environments, for example, Blumberg and Galyean (1995); Badler et al. (1993); Tu and Terzopoulos (1994); Perlin and Goldberg (1996), and Rickel and Johnson (1999). There has been extensive research on autonomously producing expressive behaviour of a number of types including facial expression (Pelachaud and Poggi (2002)), eye gaze (Cassell et al. (1999), Rickel and Johnson (1999) and Gillies and Dodgson (2002)), gesture (Cassell et al. (1999)), style of motion (Chi et al. (2000)) and, like our current implementation, posture (Cassell et al. (2001a), Bécheiraz and Thalmann (1996)).

Maya et al. (2004) have investigated how to create variation between animated characters. They use XML based profiles which are merged with an XML based specification of the affective content of a particular piece of speech, using an XSLT based system, to produce a final piece of behaviour. However, they do not provide any user friendly system for customising characters, nor does their system work in real time. The use of profiles and context dependence has also been used in other types of agent technology, for example, Soltysiak and Crabtree (1998).

3 Non-verbal behaviour

This section describes a behaviour network for non-verbal communication that we have developed. It models the way people relate to each other or their attitude to each other and is based on the work of Argyle (1975). In our model the attitude of one person to another is expressed through posture and, to a more limited degree, gesture. It is discussed in more detail in Gillies and Ballin (2003).

Though there is an enormous variety in the way that people can relate to each other Argyle identifies two fundamental dimensions that can account for a majority of non-verbal behaviour, affiliation and status. Affiliation can be broadly characterised as liking or wanting a close relationship. It is associated with high levels of eye gaze and close postures, either physically close such as leaning forward or other close interaction such as a direct orientation. Low affiliation or dislike is shown by reduced eye gaze and more distant postures, including postures that present some sort of barrier to interaction, such as crossed arms. Status is the social superiority (dominance) or inferiority (submission) of one person relative to another, we will not discuss it directly in our examples.

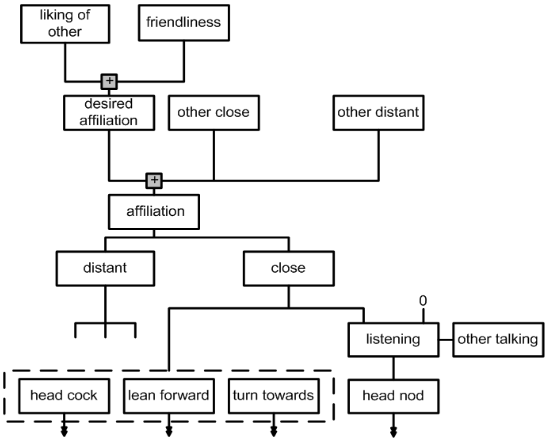


Figure 2: A section of a behavioural controller.

Figure 2 shows in diagrammatic form a fragment of the attitude behavioural controller that deals with affiliation (status is calculated in a similar way) and posture (eye gaze is discussed in section 3.2). At the top of the diagram the actual value for affiliation is calculated as a weighted sum of a number of factors (for the sake of clarity not all the factors used are actually shown). This is done in two stages, firstly the factors depending on the character itself are calculated. These factors are represented as parameters (here 'liking of other' and 'friendliness' are shown). Then factors depending on the other character's behaviour ('close' and 'distant') are added in, these are taken directly from the controller of the other character. As the behaviours associated with positive and negative affiliation are very different it is split into two terms, 'close' which is equal to the affiliation and 'distant' which is its negation. 'Close' is then mapped into actual behaviour (as is 'distant' but it is not shown in the diagram). At semi-regular intervals a new combination of the various behaviours ('head cock', 'lean forward' and 'turn towards') is produced, this combination is always proportional to the value of 'close'. These behaviour types are passed as parameters to the underlying animation system. Another affiliative behaviour is head-nodding, but this is only shown when the other person is talking. This behaviour is controlled by a switch node ('listening'), based on whether the other character is talking. If 'other talking' is true then 'head nod' is proportional to 'close' otherwise it is zero. Figure 3 shows examples of body language generated by the Demeanour framework.

3.1 Posture and Gesture

Human bodies are highly expressive; a casual observation of a group of people will reveal a large variety of postures. Some people stand straight, while others are slumped or hunched over; some people have very asymmetric postures; heads can be held at many different angles, and arms can adopt a huge variety of postures each with a different meaning: hands on hips or in pockets; arms crossed; scratching the head or neck, or fiddling with clothing. Computer animated characters often lack this variety of expression and can seem stiff and robotic; however, posture has been relatively little studied in the field of expressive virtual characters. It is a useful cue as it is very clearly visible and can be displayed well on even fairly graphically simple characters.

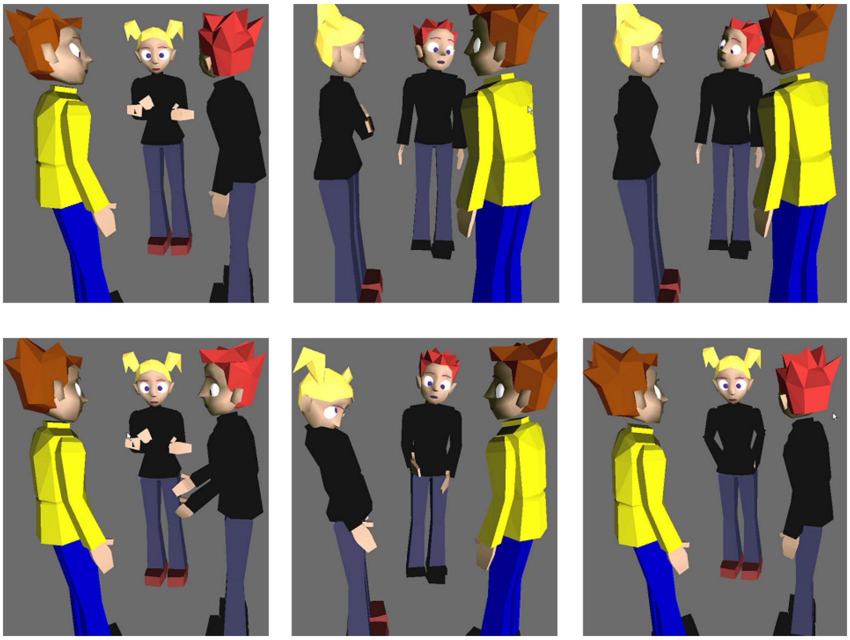


Figure 1: Three characters displaying expressive behaviour. The characters display gaze, posture and gestures behaviour and different attitudes to each other. The female character displays a negative attitude (low affiliation) whereas the male character dressed in yellow displays more positive attitude and the male character dressed in black has a mixed attitude.

Research on posture generation has been limited relative to other modalities. Cassell et al. (2001a) have investigated shifts of postures and their relationship to speech, but not the meaning of the postures themselves. As such their work is complimentary to ours. Coulson (2002) uses an OCC model of emotion to generate postures. Bécheiraz and Thalmann (1996) use a one-dimensional model of attitude, analogous to our affiliation, to animate the postures of characters. Their model differs from ours in that it involves choosing one of a set of discrete postures rather than continuously blending postures. This means that it is less able to display varying degrees of attitude or combinations of different attitudes.

The generation of gestures has been studied by a number of researchers. For example, Cassell et al. (1999) have produced a character capable of extensive non-verbal behaviour including sophisticated gestures. Chi et al. (2000) present a way of generating expressive movements, similar to gestures, using Laban notation. Gestures are closely related to speech and should be tightly synchronised with it. Cassell et al. (2001b) present a system that parses text and suggests appropriate gestures to accompany it. Gestures are less closely related to attitude than posture, though some connection can be made, for example head nodding while listening is a generally affiliative gesture.

As described in the previous section the attitude model generates a high level description of the behaviour of the character in terms of a value of each of a number of behaviour types. The behaviour modules themselves must translate this description into concrete behaviour. Each behaviour type can be expressed as a posture in a number of different ways, for example space filling can involve raising to full height or putting hands on hips while closeness can be expressed as leaning forward or making a more direct orientation (or some combination thereof). Actual postures are calculated as weighted sums over a set of basic postures each of which depends on a behaviour type.

The basic postures were designed based on the description in Argyle (1975) and Mehrabian (1972), combined with informal observations of people in social situations. The weights of each basic posture are the product of the value of its behaviour type and its own weight relative to the behaviour type. The weights of the basic postures are varied every so often so that the character changes its posture without changing its meaning, thus producing a realistic variation of posture over time. Each basic posture is rep-

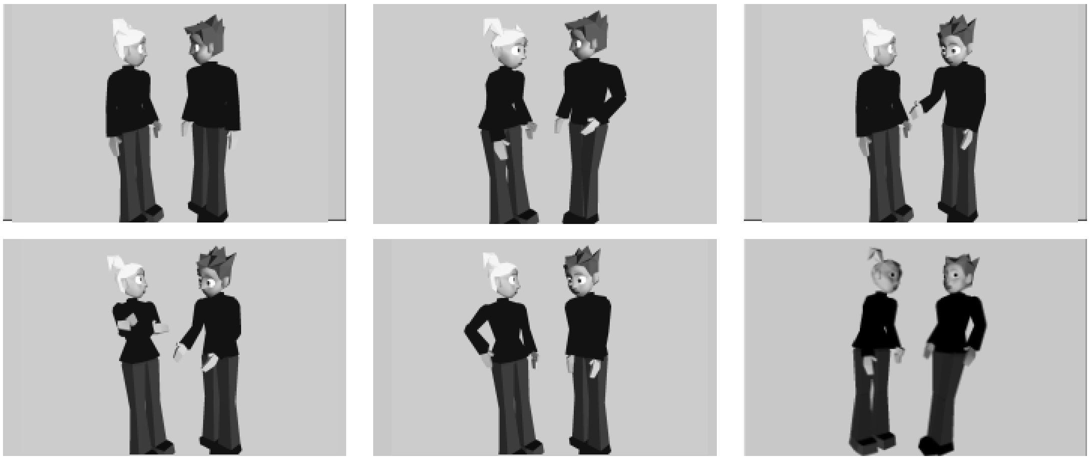


Figure 3: Examples of body language generated by the Demeanour framework to reflect different attitudes between the characters. Clockwise from top left: mutual gaze; close and relaxed postures; the male character is gesturing while talking; the female character has a distant, hostile posture; the female character has a high status, space filling posture the male character has a low status, submissive posture; the male character is relaxed (a high status posture) and the female character has a close posture.

resented as an orientation for each joint of the character and final posture is calculated as a weighted sum of these orientations.

Gesture is generated using the same body animation system as postures, the main difference being that gestures are multi-frame animations and so weighted sums must be performed over a number of frames. They are also no longer merely static poses that can be held for a period of time; they must be repeated at appropriate intervals. More importantly gestures are more closely integrated with the flow of conversation and so must be synchronised with conversation. Of course as the conversation is textual the synchronisation does not have to be as exact as it would be with spoken language. We also do not attempt to parse text so gestures are not strongly connected to the meaning of the text as in Cassell et al. (2001b). Our gesture model serves only to indicate when someone is talking and to express a degree of attitude. Figure 3 shows examples of postures and gestures.

3.2 Eye gaze

Natural eye gaze is critical to the realism and believability of an animated character. This is because eye gaze is fundamental in showing interest levels between characters and as means of anticipating events. Typically a person will look to another before exhibiting any behaviour, such as moving towards them or speaking to them. In conversation, a listener will typically spend a large proportion of their time looking at the speaker. A complete lack of gaze towards the speaker is a clear message of the lack of interest of the audience towards the speaker and will be picked up very quickly. Conversely, mutual gaze, in which two people are looking into each others' eyes is a powerful mechanism that induces arousal in the individuals, so typically mutual gaze is short (of the order of a second).

Argyle and Cook (1976) have done extensive studies with pairs of individuals to understand levels of eye gaze, and mutual gaze, and has detailed results covering (among other things) conversations and the level to which individuals will look at the other while speaking (35%) and listening (75%) etc. We have used these results to influence our model of gaze and mutual gaze in-group settings. Eye gaze is also related to attitude. Higher affiliation results in higher levels of eye gaze. Argyle and Cook have demonstrated compensatory behaviour for eye gaze. People react to higher levels of eye gaze by reacting with more distant postures, and conversely people will look at each other less if they are placed close together.

Existing simulations of eye gaze fall into two broad categories. Chopra-Khullar and Badler (1999) and Gillies and Dodgson (2002) simulate the eye gaze of characters navigating and performing actions in an environment but do not handle social factors of gaze between people. Our work is closer to the other type of simulation that deals primarily with social gaze.

Garau et al. (2001) and Colburn et al. (2000) simulate the patterns of eye gaze between pairs of characters based on frequencies of mutual gaze. Vilhjálmsson and Cassell (1998) use eye gaze to help regulate the flow of conversation by indicating when a speaker is about to finish talking, when someone wants to start or end a conversation and other similar information. Rickel and Johnson (1999), in their character based virtual reality tutoring system, use gaze primarily as a method of indicating to the user an area of interest in the environment. Thórisson (1998) simulates eye gaze in the context of more general work on multi-modal communicative behaviour during conversation.

Each character has a set of foci of interest, which are objects that it will look at. The level of interest is specified as the proportion of time spent looking at that object. So for example if the character is in conversation with another character, while talking the level of gaze will be set to (say) 35%, and whilst listening to about 75% to approximate the natural gaze levels in conversation between two people.

However, this base value is also affected by the affiliation attitude between the character that is looking and the one that is being looked at. A close attitude increases proportion of gaze (up to a maximum of 100%) and distant behaviour reduces it (to a minimum of 0%). The exact formula used to determine the actual eye gaze is:

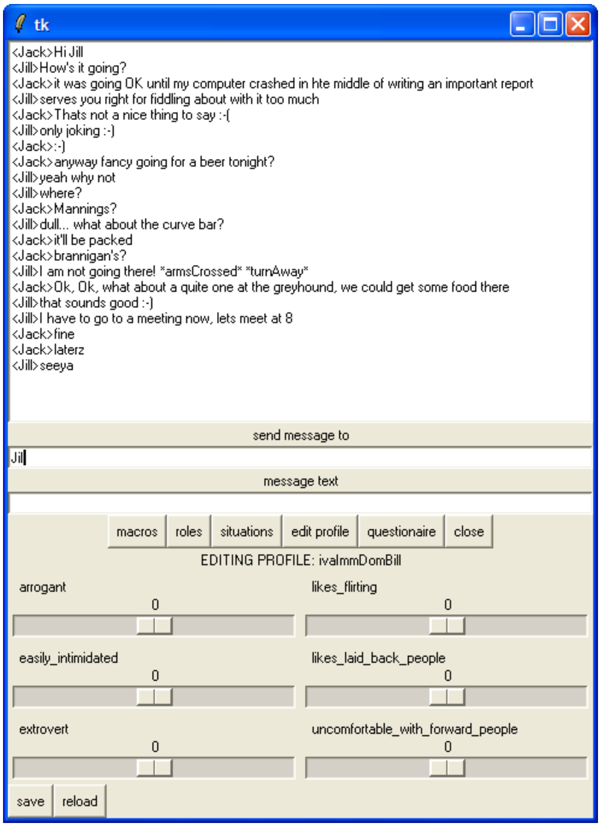
$$g = g_{cond} - g_{cond}\frac{distant}{d_{max}} + (1 - g_{cond})\frac{close}{c_{max}}$$

where g is the proportion of time spent gazing at the target on average. g_{cond} is the gaze proportion due to the condition (talking, listening or neither). $distant$ and $close$ are the values for the close and distant attitudes and d_{max} and c_{max} are the values at which the gaze proportion is either 0 or 1.

In conversation between people a person will look at another then look away, usually by averting their gaze rather than moving their head, but they are not looking specifically at any other object, just averting their gaze. In our model we achieve this by having a number of 'halo' points around the head of a character that can be selected to look at if we need to look away, and have no other object that demands our attention.

4 Profiles

Demeanour provides a system of character profiles for off-line customization by end-users or world designers. By world designers we mean expert con-



```
<Jack>Hi Jill
<Jill>How's it going?
<Jack>it was going OK until my computer crashed in hte middle of writing an important report
<Jill>serves you right for fiddling about with it too much
<Jack>Thats not a nice thing to say :-(
<Jill>only joking :-)
<Jack>:-)
<Jack>anyway fancy going for a beer tonight?
<Jill>yeah why not
<Jill>where?
<Jack>Mannings?
<Jill>dull... what about the curve bar?
<Jack>it'll be packed
<Jack>brannigan's?
<Jill>I am not going there! *armsCrossed* *turnAway*
<Jack>Ok, Ok, what about a quite one at the greyhound, we could get some food there
<Jill>that sounds good :-)
<Jill>I have to go to a meeting now, lets meet at 8
<Jack>fine
<Jack>laterz
<Jill>seeya
```

send message to: Jil

message text:

EDITING PROFILE: ivalmmDomBill

arrogant: 0
likes_flirting: 0
easily_intimidated: 0
likes_laid_back_people: 0
extrovert: 0
uncomfortable_with_forward_people: 0

Figure 4: The user interface used for real time control fo chara

tent creators with some programming skills or at least the ability to handle technologies such as XML used when defining character behaviour and adjectives for profile creation (see below). Interfaces for end users are aimed at typical computer game players, not experts but familiar with instant messaging and 3D navigation. Player character's are controlled mostly through a text chat interface, through which players can enter text to be spoke and emoticons which control the character's behaviour (as well as choosing profiles). The user interface is shown in figure 4.

A profile is a set of data that determines the unique behaviour of a character, i.e. how it differs from other characters. In Demeanour a character's behaviour is generated by a parameterised behavioural controller (the structure of the controllers is discussed in more detail in Gillies and Ballin (2004)). This controller can be the same for each character but changing the values of the parameters allows for different behaviour.. Customization is possible by altering the values of the parameters e.g the weighting for how the closeness behaviour of other character affects a char-

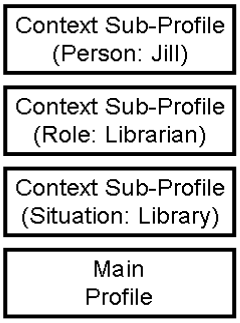


Figure 5: The profiles stack containing a number of loaded contextual profiles

acter's affiliation. A profile can set this weighting to a positive value to achieve reciprocating behaviour, negative for compensation and a low or zero value for indifference to the other's status.

Thus a profile consists of a number of values for parameters of the behavioural controller. These values are stored in an XML-based format separate from the controller definition. When a profile is loaded into a behavioural controller the values in the profile are used to set the parameters of the controller (profile values are matched to parameters by name). Profiles are used as a means of customising a character, and a means of providing contextual variation. This means there will be a number of profiles loaded in a controller at any given time. They are stored in a stack as shown in figure 5. The base of the stack is always the main profile that contains the context independent customisations of a character. Above this, a number of context dependent profiles are loaded as described in section 4.2. When a new context profile is loaded it is added above all the previously loaded profiles in the stack but below the temporary and conversation profiles. Profiles higher up the stack will override profiles lower in the stack, so recently loaded profiles override older ones and user input overrides other profiles. However, this process can be controlled by giving priorities to values within a profile. Values can have two priorities, *required* and *optional*. Required values always override values lower down the stack but optional values only override other optional values, and so are only loaded if no profile has a required value for that parameter.

4.1 Profiles for customisation

The primary function of character profiles is the customisation of characters. End users should be able to customise the behaviour of their character and designers of virtual worlds should be able to provide variety in the autonomous agents in their world. Each character has a main profile, at the base of the profile stack, containing values for the parameters of the behavioural controller that determines the unique behaviour of that character. This is the main customisation system for a given character.

To be an effective customisation method, easy to use tools must be provided for designing profiles. The most direct method is for the user to choose values to parameters whether by hand editing files or via a user interface. However, parameters are often closely linked to the internal workings of the behavioural controller and not necessarily intuitive to end-users, so this method should generally be confined to world designers and advanced users.

We propose the use of "adjectives". These are names in natural language that describe a particular character trait or group of traits that is understandable to end-users. These adjectives are mapped onto actual settings of the internal parameters, each adjective affecting a number of parameters. For example, 'extrovert' might combine dominance with high affiliation while 'easily intimidated' might indicate compensation behaviour to dominance (i.e. responding submissively to dominant behaviour). Each adjective is a fixed set of parameter values and therefore is itself a self contained profile. The adjectives themselves are chosen by world designers. They can be designed at the same time as the behaviour network, through direct profile authoring tools as above. An end-user designs their profile as a combination of the adjectives. They are presented with series of sliders each labelled with an adjective name, the values of the sliders represent the proportions of the various adjectives. The values contained in the adjectives are multiplied by the slider values and summed to obtain the final profile. This provides a customization tool that is easy to use, abstracts from the internal workings of the controller, and is itself easily customizable by world designers. Figure 4 shows an example of the user interface for choosing adjective weights.

4.2 Profiles and context

As described in the introduction the variability of human behaviour is not solely between individuals but within individuals. People behave very differently in different contexts and it is important to also model this sort of variability. The importance of this type of adaptation is brought out in work by MacNamee et al. (2002) and Maya et al. (2004). Goffman (1972) provides a fascinating description of how people's behaviour varies in different contexts. This is particularly true for characters in narrative where the character's behaviour must reflect the unfolding story.

In order to handle this sort of variability Demeanour uses a system of sub-profiles for specific context. A sub-profile is a small set of parameter values that are loaded in a given context to alter the behaviour of the main profile. These are loaded above the main profile in the stack as shown in figure 5, with more recently loaded sub-profiles overriding older ones.

The variation of a person's behaviour in different contexts can depend on a number of different factors and so these contexts themselves can have different meanings, for example, a relationship with a colleague may define a context for interaction with that colleague but the context would also depend on whether they are at work or in a social context. We divide contextual sub-profiles into three types depending on when they are loaded and to some degree who designs them. The system could be augmented to add a number of other types.

Person sub-profiles define a relationship to a particular individual and thus represent the attitude to that person. They are loaded when an interaction starts with that individual. The parameter 'Liking of other' in figures 2 is an example of a parameter that represents an attitude to a particular person and is suited to inclusion in a person sub-profile. Person profiles are generally designed by end users.

Role sub-profiles represent the behaviour when the character is performing a particular role. These roles are often related to work, for example, a waiter behaves very differently when actually serving customers than when not working or when interacting with other staff in the restaurant kitchen (Goffman (1972)). A practical example, might be eliminating all low affiliation behaviour when a waiter is at work to ensure politeness. Role sub-profiles can be loaded by the end-user or automatically loaded in a given situation. Role profiles can be created by world designers to give a user character specific behaviour in a role that the user might not have forseen or they can be created by an end user.

Situation sub-profiles are specific to a particular environment or narrative context, for example, flirting behaviour might be disabled in an office environment but re-enabled in an office party context. They are loaded automatically when the character enters a situation and applied to all characters in that situation.

5 Conclusion

We have described the Demeanour framework's system of profiles which is applied in a number of ways, end user customisation, context dependence and real time control. This system shows promise in producing some of the variability and adaptability of real human behaviour.

Acknowledgements

We would like to thank BT plc. for sponsoring this research. We would also like to thank the UCL Department of Computer Science Virtual Environments and Computer Graphics group for their help and support, and Amanda Oldroyd for the use of her character models.

References

M. Argyle. *Bodily Communication*. Routledge, 1975.

M. Argyle and M. Cook. *Gaze and Mutual Gaze*. Cambridge University Press, 1976.

N. Badler, C. Philips, and B. Webber, editors. *Simulating Humans: Computer Graphics, Animation and Control*. Oxford University Press, 1993.

P. Bécheiraz and D. Thalmann. A model of nonverbal communication and interpersonal relationship between virtual actors. In *Proceedings of the Computer Animation '96*, pages 58–67. IEEE Computer Society Press, June 1996.

B. Blumberg and T. Galyean. Multi-level direction of autonomous creatures for real-time virtual environments. In *ACM SIGGRAPH*, pages 47–54, 1995.

J. Cassell, T. Bickmore, L. Campbell, K. Chang, H. Vilhjálmsson, and H. Yan. Embodiment in conversational interfaces: Rea. In *ACM SIGCHI*, pages 520–527. ACM Press, 1999.

J. Cassell, Y. Nakano, T. Bickmore, C. Sidner, and C. Rich. Non-verbal cues for discourse structure. In *41st Annual Meeting of the Association of Computational Linguistics*, pages 106–115, Toulouse, France, 2001a.

J. Cassell, H. H. Vilhjálmsson, and T. Bickmore. BEAT: the behavior expression animation toolkit. In *ACM SIGGRAPH*, pages 477–486, 2001b.

D. Chi, M. Costa, L. Zhao, and N. Badler. The emote model for effort and shape. In *ACM SIGGRAPH*, pages 173–182. ACM Press/Addison-Wesley Publishing Co., 2000.

S. Chopra-Khullar and N. Badler. Where to look? automating visual attending behaviors of virtual human characters. In *Autonomous Agents Conference*, 1999.

A. Colburn, M. Cohen, and S. Drucker. The role of eye gaze in avatar mediated conversational interfaces. Technical report, Microsoft Research, 2000.

M. Coulson. Expressing emotion through body movement: a component based approach. In Ruth Aylett and Lola Cañamero, editors, *AISB workshop on Animating Expressive Characters for Social Interactions*, Imperial College London, April 2002.

M. Garau, M. Slater, S. Bee, and M. A. Sasse. The impact of eye gaze on communication using humaniod avatars. In *ACM SIGCHI*, pages 309–316, 2001.

M. Gillies and D. Ballin. A model of interpersonal attitude and posture generation. In Thomas Rist, Ruth Aylett, Daniel Ballin, and Jeff Rickel, editors, *Fourth Workshop on Intelligent Virtual Agents*, Kloster Irsee, Germany, September 2003.

M. Gillies and N. Dodgson. Eye movements and attention for behavioural animation. *Journal of Visualization and Computer Animation*, 13:287–300, 2002.

Marco Gillies and Daniel Ballin. Integrating autonomous behavior and user control for believable agents. In *Third international joint conference on Autonomous Agents and Multi-Agent Systems*, Columbia University, New York City, July 2004.

Erving Goffman. *The presentation of self in everyday life*. Pelican, 1972.

Brian MacNamee, Simon Dobbyn, Padraig Cunningham, and Carol O'Sullivan. Men behaving appropriately: Integrating the role passing technique into the aloha system. In Ruth Aylett and Lola Cañamero, editors, *AISB workshop on Animating Expressive Characters for Social Interactions*, Imperial College London, April 2002.

Vincent Maya, Myriam Lamolle, and Catherine Pelachaud. Influences on embodied conversational agent's expressivity. In Patrick Olivier Ruth Aylett and Marc Cavazza, editors, *AISB workshop on Language, Speech and Gesture for Expressive Characters*, pages 75–85, Univesity of Leeds, March 2004.

A. Mehrabian, editor. *Nonverbal Communication*. Aldine-Atherton, 1972.

C. Pelachaud and I. Poggi. Subtleties of facial expressions in embodied agents. *Journal of Visualization and Computer Animation.*, 13:287–300, 2002.

K. Perlin and A. Goldberg. Improv: A system for scripting interactive actors in virtual worlds. In *Proceedings of SIGGRAPH 96*, Computer Graphics Proceedings, Annual Conference Series, pages 205–216, New Orleans, Louisiana, August 1996. ACM SIGGRAPH / Addison Wesley.

J. Rickel and W. L. Johnson. Animated agents for procedural training in virtual reality: Perception, cognition, and motor control. *Applied Artificial Intelligence*, 13:343–382, 1999.

S. J. Soltysiak and I. B. Crabtree. Automatic learning of user profiles – towards personalisation of agent services. *BT Technology Journal*, 16(3):110–117, 1998.

K. Thórisson. Real-time decision making in multimodal face-to-face communication. In *second ACM international conference on autonomous agents*, pages 16–23, 1998.

X. Tu and D. Terzopoulos. Artificial fishes: Physics, locomotion, perception, behavior. In *ACM SIGGRAPH*, pages 43–49, 1994.

H. H. Vilhjálmsson and J. Cassell. Bodychat: Autonomous communicative behaviors in avatars. In *second ACM international conference on autonomous agents*, 1998.

A Study of Episodic Memory-Based Learning and Narrative Structure for Autobiographic Agents

Wan Ching Ho⋆, Kerstin Dautenhahn⋆, Chrystopher L. Nehaniv⋆
⋆Adaptive Systems Research Group, School of Computer Science, University of Hertfordshire
College Lane Campus, Hatfield, Hertfordshire, AL10 9AB, United Kingdom
`[W.C.Ho, K.Dautenhahn, C.L.Nehaniv]@herts.ac.uk`

Abstract

In this study we develop and compare the performance of different agent control architectures based on learning through episodic memory for the design of Non-Player Characters (NPCs) in computer games. We focus on the Categorised Long-term Autobiographic Memory (CLTM) architecture, utilising abstracted notions of human autobiographic memory and narrative structure humans apply to their life stories. We also investigate the influence of remembering negative experience on agents' adaptivity. A large and dynamic virtual environment is created to examine different agent control architectures in an Artificial Life and bottom-up fashion. Agents' lifespan is measured in the experiments. Results show that CLTM architecture including remembering negative events can significantly improve the performance of a single autonomous agent surviving in the dynamic environment.

1 Introduction

The design of Non-Player Characters (NPCs) has been a critical issue for many modern computer games since game players' expectations are getting higher and higher. Game developers put huge amount of efforts into computer graphics and animations. An additional alternative focus for enhancing their believability in character-based computer games is intelligence expressed by NPCs' behavioural patterns. Although Finite State Machines and Scripting have been two dominant techniques to craft NPCs' intelligence (Cass, 2002), in recent years researchers have attempted applying various AI learning techniques such as Trial and Error, Imitation, Neural Networks and Genetic Algorithm, to create reasonable and intelligent behaviours for NPCs. Among these techniques, Isla and Blumberg (2002) pointed out that, as a learning mechanism with a great potential, *episodic memory* has been only made explicit use by a few behaviour simulation systems. They also indicated that the advantage of using episodic memory for learning, compared to other mechanisms, is speed. NPCs can form usable hypotheses for making decisions or selecting behaviours to execute in the future, after just one observation of users or other agents.

The theoretical term *autobiographic agents* was first defined by Dautenhahn (1996, page 5): "we define the concept of an autobiographic agent as an embodied agent which dynamically reconstructs its individual 'history' (autobiography) during its life-time.". Nehaniv (1999) further pointed out three areas of temporal historical grounding of for narrative autobiographic agents: 1) recognition of narrative structure, 2) expressing narrative structure and 3) having a narrative structure.

Our work focuses on the above areas by building memory architectures for artificial autonomous agents. We expect that, by having narratively structured autobiographic memory as 'extrasensory' data, the increased temporal horizon may free agents from the rigid perception-action cycle. Therefore, the main issue in this paper is to design and validate the bottom-up narrative structure for autobiographic agents in remembering significant events experienced during their lifetime for adaptation to the dynamic environments.

2 Dynamic Virtual Environment and Agent Embodiment

In order to examine the performance of our agent control architectures and the utility of narrative storytelling features between LTM agents in the future work, we required a rich set of possible event sequences. We created a large, dynamic and complex 'nature-like' virtual environment has been created by using VRML and Java programming languages. This environment is fairly different from other simple and flat agent test-beds since it has various types of resources, most of them dynamically distributed on different kinds of

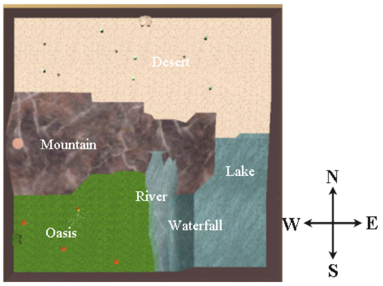


Figure 1: The dynamic virtual environment.

landforms. Figure 1 illustrates the virtual environment model.

All virtual agents are designed to have a finite lifespan and are required to wander in the environment as their basic behaviour. The survival of an agent depends on maintaining homeostasis for its four internal physiological variables, namely *glucose*, *moisture*, *energy* and *body temperature*.

Due to space constraintof this paper, details about environmental structure and agent embodiment in this study can be found in Ho et al. (2005).

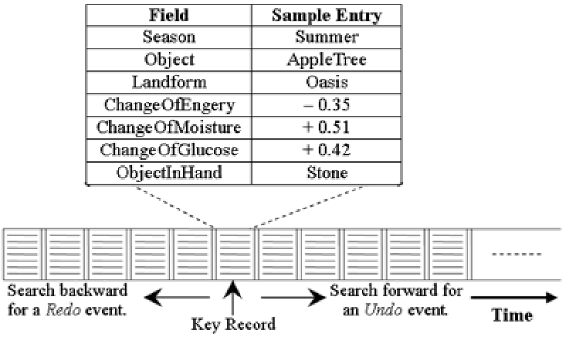
3 CLTM Control Architecture

We aim to develop appropriate autobiographic memory architectures on top of a basic subsumption control architecture in order to enhance the agents' performance in surviving in a dynamic environment. Inspired by human long-term memory (Alba and Hasher, 1983) and autobiographic memory models from related research in psychology (Conway, 1992), we developed a more sophisticated CLTM architecture compared with the Purely Reactive (PR), Short-term Memory (STM) and Long-term Autobiographic Memory (LTM) models used in our previous study (Ho et al., 2005). The CLTM architecture addresses our fundamental research issue in this paper – learning through *significant* episodic information. As humans remember experiences in a way that makes narrative sense, we abstract the important points from Linde's narrative structure (Linde, 1993) to create the underlying mechanism for agents to remember events experienced by themselves.

Linde (1993) formally developed a narrative structure for applying to life stories of humans. This narrative structure provides a clear picture regarding different features of narrative and how these features can fit together in a simple personal experience. It can be interpreted as in Table 2 together with how we implement these features into our CLTM architecture:

Feature	**Abstract**	**Orientation (optional)**	**Narrative**	**Evaluation**
Explanation from Linde	**Type of story**	**Abstractions about who, when, where and what happened**	**Detailed descriptions of the event(s) that happened in the story**	**Evaluation of the whole story or a part of the story**
Implementation in CLTM architecture	**Type of event (which category)**	**Event abstractions about object area, season, and type (positive or negative**	**Situations constructed the event (starting, intermediate and ending situations)**	**Evaluation value (*significance*) of the whole event**

Figure 2: Comparison between CLTM architecture and the original explanations for the main features of Linde's narrative structure



Field	Sample Entry
Season	Summer
Object	AppleTree
Landform	Oasis
ChangeOfEngery	– 0.35
ChangeOfMoisture	+ 0.51
ChangeOfGlucose	+ 0.42
ObjectInHand	Stone

Figure 3: Working Memory consists of situations in CLTM. Events are reconstructed in two directions: *Redo* and *Undo*.

3.1 Working Memory and Event Categories

An CLTM agent surviving in the dynamic environment has a finite list of recent records, called Working Memory; each record is an abstracted situations representing a situation of a particular moment when the agent tries to remember the event context – in this case, the objects and the landform of its surrounding environment and its internal variables. The name of each field in a record and sample entries are shown in Figure 3.

If a record contains a situation when agent was able to sense a resource in the environment, this record will become the *Key Record* for reconstructing events. Starting from a Key Record in Working Memory of CLTM, an agent will reconstruct events experienced in the past in both directions for *Redo* and Undo events. Each event contains an *Evaluation* value which derives from the measurement of the total change of internal variables. The significance of each event is determined by this value. The overall structure of an event reconstructed from Working Memory is similar to Linde's narrative structure.

The second part of CLTM is Event Categories, in which each category contains one type of events. Since the same type of events can be repeatedly experienced by an CLTM agent, only the most significant event (an event with the highest Evaluation value) among repeated events reconstructed from Working Memory is remembered in each category. Categories can be divided into two types: *positive event categories* and *negative event categories*.

3.2 Positive Event and Negative Event

One of the important features of narrative is *breaches* – a story should contain something unexpected, some problem to be resolved or some unusual situation, as described by the founder of narrative psychology Bruner (1991). By considering an event as a narrative which is worth remembering, an event should not only contain general positive information. Therefore we have developed *negative event categories* for CLTM agents to remember negative events which bring negative changes to internal states of them.

In *positive event categories*, an CLTM agent remembers different types of positive event which may guide this agent to locate useful resources in the environment.

Negative events in *negative event categories* can help an CLTM agent to avoid getting into a trouble area which may trap an agent and bring a considerable amount of decrease to internal physiological variables of this agent.

To execute an *LTM Trace* for either a *Redo* or *Undo* event from positive event categories, the agent will try to achieve the next situation from the current situation, until it reaches the target one.

4 Experiments

We aim to measure the performance of three types of agent architectures running in the dynamic virtual environment: Categorised Long-term Autobiographic Memory without remembering negative events (CLTM_Pos); and with remembering negative events (CLTM_Pos_Neg), and Short-term Memory plus CLTM_Pos_Neg (STM+CLTM_Pos_Neg). We carried out 10 experimental runs for each architecture. Each run takes approximately 30 minutes on a Pentium 4 2.0GHz PC with 512MB Ram. For the fifth type STM+CLTM_Pos_Neg control architecture, we have arranged the STM to have higher priority to execute its *Trace-back* process than *CLTM Trace* in the sense of decision making. The starting position for all agents in the experiments is in the center of the oasis area. At the beginning of each experimental run the agent performs a random rotation.

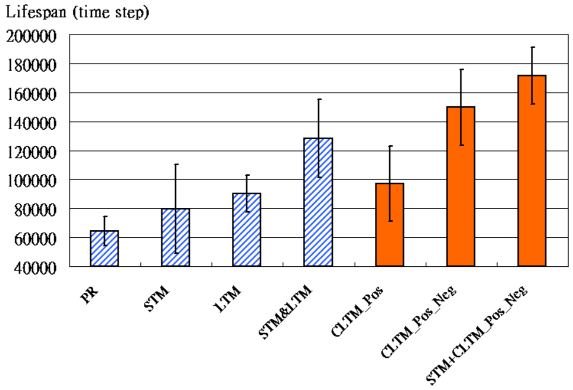


Figure 4: Average lifespan for different types of memory control architectures. Error bars are applied as confidence value to the results. Note that results for architectures PR, STM, LTM and STM+LTM are from our previous work.

In all experimental runs with STM and CLTM architectures involved in, the length of STM is set to 50 entries, and the length of Working Memory in CLTM is set to 30 entries.

4.1 Results

Figure 4 shows average lifespan of totally seven types of agents in 10 experimental runs. To make comparison with memory architectures we developed in our previous study, results for architectures PR, STM, LTM and STM+LTM are from Ho et al. (2005). Details of implementation of these two architectures are not described in this paper, but the simulation environment and experimental conditions in this study are exactly the same as we used in Ho et al. (2005).

4.2 Discussion and Analysis

Results for single-agent experiments in this study can be concluded as follows:

1. The average lifespan of the CLTM_Pos_Neg agent and the STM+CLTM_Pos_Neg agent outperform the PR, STM and CLTM_Pos agent. It implies that remembering negative events helps CLTM agents to be more adaptive in the sense of surviving in the dynamic environment, as agents remembering negative event can avoid repeating the same mistakes of going to some disadvantageous areas. CLTM_Pos architecture does not produce good enough result to significantly outperform PR and STM architectures.
2. Although from time to time the STM agent with *Trace-back* process is able to precisely undo all actions of an event and come back to the resource which was encountered previously; the performance of the *Trace-back* process from the STM

agent is sometimes affected by the environmental dynamics, such as the seasonal resource distributions. Therefore the average lifespan of the STM agent, with a high confidence value, cannot be considered as outperforming the PR agent.

3. The agents with STM+CLTM_Pos_Neg have the highest average lifespan. This result is reflected in agents' memory control architecture as it combines the precision offered by the *Trace-back* process from STM and the flexibility of CLTM to cope with the environmental dynamics.

In our previous study (Ho et al., 2005), we used an list with unlimited length as agents' Long-term Autobiographic Memory (LTM). Each time when an LTM agent needs to retrieve a significant event from LTM, it reconstructs a memory schema which contains a certain amount of useful events experienced in the past, and then rank these events by comparing their significances for selecting the most significant one to execute the LTM Trace process. Compared with LTM architecture, Categorized Long-term Autobiographic Memory (CLTM) we developed in this study has the following advantages:

- The most significant event is chosen from the same type of events and stored into a category. On one hand, this approach saves the computational memory space as agents can safely forget repeated and less significant events. On the other hand, agents can improve the performance by spending less time on searching useful events from the permanent Long-term Autobiographic Memory.
- CLTM agents are able to remember negative events. Some areas in the environment are less beneficial to the agents: either the useful resource are located in the corner which is difficult to be reached, or these areas simply do not have any resource. Remembering the negative experiences in the past may enhance agents chance to survive. In addition to avoid entering the disadvantageous areas, agents staying in 'safe' areas increases their chance to encounter resources.

Therefore, the results produced by the memory control architectures LTM and STM+LTM from our previous study (Ho et al., 2005) are generally not as good as CLTM architectures with remembering negative events.

5 Conclusions and Future Work

In this paper, we developed Categorized Long-term Autobiographic Memory (CLTM) architecture for establishing episodic memory- and narrative structure-based learning dedicated to NPCs design in computer games. Next, through enabling CLTM agents to remember their negative experiences, we showed agents' adaptivity is further enhanced, as shown in experimental results.

In the future, we are interested in investigating how the length of STM and Working Memory of CLTM influences agents' performance. We also expect that if CLTM agents can share both positive and negative experiences through story-telling, interesting results can be obtained through experiments and observations. Furthermore, STM and CLTM agents' further potential can be discovered by running them in a gaming environment with input from human players.

References

J W. Alba and L Hasher. Is memory schematic? *Psychological Bulletin*, 93:203–231, 1983.

J. Bruner. The narrative construction of reality. *Critical Inquiry*, 18(1):1–21, 1991.

S. Cass. Mind games. *IEEE Spectrum*, pages 40–45, 2002.

M. A. Conway. A structural model of autobiographical memory. In M. A. Conway, D. C. Rubin, H. Sinner, and E. W. A. Wagner, editors, *Theoretical Perspectives on Autobiographical Memory*, pages 167–194. Dordrecht, the Netherlands: Kluwer, 1992.

K. Dautenhahn. Embodiment in animals and artifacts. In *AAAI FS Embodied Cognition and Action*, pages 27–32. AAAI Press, 1996. Technical report FS-96-02.

W. C. Ho, K. Dautenhahn, and C. L. Nehaniv. Autobiographic agents in dynamic virtual environments - performance comparison for different memory control architectures. In *Proceedings of IEEE Congress on Evolutionary Computation - Special Session: Artificial Life*, pages 573–580, 2005.

D. Isla and B. Blumberg. New challenges for character-based AI for games. In *AAAI Spring Symposium on AI and Interactive Entertainment*, Palo Alto, CA, March 2002.

C. Linde. *Life Stories: The Creation of Coherence*. Oxford University Press, 1993.

C. L. Nehaniv. Narrative for artifacts: Transcending context and self. In *Narrative Intelligence*, AAAI Fall Symposium 1999, pages 101–104. AAAI Press, 1999. Technical Report FS-99-01.

Authoring Dynamic Storylines in Interactive Virtual Environments

Shahryar Kashani*
*Department of Artificial Intelligence
University of Edinburgh
Shah@Smallgodseve.com

Judy Robertson†
†Division of Computing
Glasgow Caledonian University
Judy.Robertson@gcal.ac.uk

Abstract

The last few decades has seen the emergence of a number of interactive virtual environments attempting to combine stories and interactivity. The aim of the research has been to find a suitable middle ground between interactive freedom and a dramatic and coherent story structure. One of the major shortcomings of past approaches has been the lack of focus placed on the human user, making the user in the environment an observer rather than an interactive participant with control over the direction of the story. We show that this can be addressed by populating the environment with believable agents that have personality and emotions targeted towards the user. The agents have a range of emotions and an attitude towards the participant based on their interaction with, and the actions of, the participant. In the paper, we demonstrate how this user-modelling approach, inspired by a range of different disciplines, addresses some of the issues of past approaches. Using a character-driven story structure based on traditional storytelling techniques, we allow the relationships between the agents and the user to determine the direction of the story, giving the user a sense of interactive freedom while maintaining a dramatic narrative.

Introduction

> “Each of us has been designed for one of two immortal functions, as either a storyteller or as a cross-legged listener to tales of wonder, love and daring. When we cease to tell or listen, then we no longer exist as a people. Dead men tell no tales.”
>
> Bryce Courtenay, 1998

Storytelling: an ancient art that has been an integral part of our existence since the beginning of time. The act of telling stories, or narration, is a faithful companion in all of our core activities; from education to entertainment, through music to theatre, all the way to the daily communication with our friends, family and even strangers. As an audience, the most memorable experiences take place when we are absorbed by the story; when the world around us morphs into the universe described by the storyteller. This immersion brings the desire of agency; to be able to interact with the characters and be a participant in the story being told.

Although classic stories are linear, throughout history we have been trying to break away from the covers of a book. Oral storytellers often adjusted their stories to fit the reactions of the audience (Murray, 1990), participatory theatre and improvisation acting has actively involved the audience in the unfolding events in the story (Boal, 1979) and hugely popular interactive books have been written where the audience has a wide range of choices in the progression of the story (Gamebooks, 2004). Similarly, interactive media, such as games, which by definition are based on the participation of a player, are placing an increasing amount of focus on the story to engage the audience.

There is a basic tension between the concept of interactivity and the structure of a coherent story. A classic story has a linear narrative that must be maintained for the purpose of the story to be fulfilled. On the other hand, interactivity is based on giving the participant control over the direction of the story, and hence there cannot be a structure to uphold. The challenge lies in finding the balance between the control given to the participant to alter the direction of the story, and the control assigned to the system to maintain the structure that makes the story coherent. In other words, the goal has been to find the suitable middle ground between interactive freedom and a dramatic story structure.

One of the main problems of past approaches has been the degree of involvement assigned to the human participant of the system (the *player*). The player should be given the sense of being the protagonist of the story, and not just another element in the environment. It is easy to neglect the player in an attempt to uphold the fine structure. However, this reduces agency significantly; another casualty caused by the aforementioned tension. In this paper, we shall see how cross-disciplinary user-modelling methods, combined with classic storytelling techniques, address these issues.

Related Work

The Oz project (1989-2002) was one of the first attempts to create interactive drama by employing and focusing on the behaviour and interaction of characters. The Oz architecture takes the approach of populating the environment with independent and autonomous beings called *believable agents* (Mateas, 1997). For an agent to be believable, it should give the illusion of life i.e. perform actions on its own initiative justified by its personality, emotions, self-motivation and social relationships. The overall aim is to produce an emergent narrative based on the interactions in the environment. In other words; very little is planned and it is hoped that the unfolding events will provide a dramatic experience, much like theatre improvisation. This differs from most other approaches because it is a simulation environment (the predecessor to titles such as *The Sims*) rather than a planning architecture. The general limitation of the latter is that they quickly become unscalable, as every action and consequence has to be accounted for. Assigning such responsibility to the author will inevitably remove some of the freedom the user is granted and thus reduce the level of interactivity. This limitation is present in several planning architectures, such as Lang (1999), Charles, et al. (2002), Gordon and Iuppa (2003) and to some extent the Liquid Narrative approach (Riedl et al., 2003, Riedl and Young, 2004 and Liquid Narrative, 2004), although their focus is not primarily on interactivity, but telling a story and providing a cinematic experience complete with dynamic control of camera. In planning architectures, the stories are carefully scripted, and although the player has some part in the unfolding events, care is taken to restrict the chances of interrupting the plans.

Methodology

In the following section, we shall describe how our method differs from past research such as the Oz Project, using an approach inspired by oral storytelling techniques and cognitive emotion models. The most significant issues, and the approaches applied to address these, will be presented in three parts: player agency, drama management and agent authoring.

Player Agency

For true interactive experiences, the human player within the environment should feel a sense of *agency*. As defined by Murray (1997:126); "Agency is the satisfying power to take meaningful actions and see the results of our decisions and choices." Interactivity and agency are related terms; for a story to be interactive, the player should feel that his or her actions have some effect on the outcome, i.e. be given a feeling of agency. Therein lies one of the problems of the Oz architecture; it is a system created to populate believable agents that display interesting behaviour, but not necessarily based on the actions and decisions of the human player. In other words, player interaction with the agents is not prioritised when determining the agent behaviour. The main focus of the agents is solving their own personal goals, a fact that often reduces the player to either an obstacle or an observer. The lack of focus on the actions of the human player reduces the player from protagonist to interactive observer; the same problem faced by Façade (Mateas and Stern, 2003). We address this by concentrating the generated emotions on the actions of the human player. In other words, we are increasing user modelling and hope to achieve more agency by monitoring the player and adjusting the behaviour of the agents accordingly. The model chosen for our emotion synthesis is that of Ortony, Clore, & Collins (Ortony et. al, 1988). Their computational model has established itself as a standard for emotion synthesis and is applied to a broad range of different fields. The model specifies 22 emotions in different categories, based on reactions to certain goal-related events, acts of other agents or likeability of objects. This feature makes *OCC* suitable to employ in an interactive story environment, because stories are mainly focused around characters and objects. The *OCC* model was used in a similar environment by Elliott (1992), who successfully used an expanded version to generate emotions in a simulation called TaxiWorld. Bartneck (2002) describes a guide for emotion processing, i.e. how one can use the *OCC* model for character development. It discusses the terms that characters should follow from the initial categorisation of an event to the resulting behaviour of the character. The process is divided into five phases; classification, quantification, interaction, mapping and expression.

Drama Management

Much like live role-play dungeon masters progressively construct a story based around events that occur in a game of e.g. Dungeons & Dungeons, we need a way of ensuring that the unfolding events adhere to a coherent storyline and build up toward a climax. This is where the aforementioned narrative dilemma is introduced. We do not want to pose a great number of restrictions on what could unfold in the environment, because this removes the user agency. However, restrictions are essential if we wish the actions of the human player and the agents to be meaningful with regard to the story. What we need, then, is to remove some of the burden of understanding the inner workings of the well-written stories and evaluation functions, and not least hard-coding story content, from the authors and place it in some of the mechanics of the system. Traditional storytelling techniques can provide some clues on how this can be achieved.

In Propp's analysis of a corpus of 100 Russian fairytales it was found that in each of the fairytales a series of story points based around the protagonist were performed by the other characters (Propp, 1958). These story points, or *functions*, were recurring, and Propp narrowed the list down to 31 plot elements that consistently occurred in a uniform sequence. No story contained all functions, but he proposed that these functions encompassed all of the plot components from which fairy tales were constructed. These include absence of family member, interdiction to hero, violation of interdiction, attempted reconnaissance by villain and so on. This suggests that characters in stories are defined from a dramatic point of view. In other words, they are there to fulfil a purpose and to perform a function that is necessary for the plot. This reminds us that some characters are simply there to progress the narrative. Most importantly, these functions give us some indication on how a story should be structured, and removes the need of manually creating an evaluation function.

From Propp's functions, 8 different narrative roles that are needed to act out the story points or functions have been identified: hero, villain, helper, princess (sought-for-person), father of the princess (task assigner), mission dispatcher, donor and the false hero. These are for a particular example story, however, similar types were found in most of the stories analysed by Propp. In our architecture, as the simulation environment runs, events will occur that change the inner emotions of the agents. These events are mainly based on the actions of the human player. In other words, these emotions can be used as a way of describing the relationship between the player and the individual agents. For instance, if on several occasions the player has found an item that one agent wants, the agent will become increasingly jealous. The relationship between them can then be described as negative. We can use these emotions to determine which agent is suitable for what role. This decision is influenced by the choices of the player so agency is maintained.

What remains is the actual construction of the agents, which brings us to the next part of the methodology: defining personalities.

Defining Agents

The bulk of the work in creating agents lies in the defining of personalities. Ideally, this task should not be overly difficult for the author. For an interactive environment to be fully immersive, it needs to be inhabited by a number of agents with different and distinct personalities, much like real life. This poses two requirements: firstly, the architecture should be scalable, meaning that the construction of the story should not be strongly affected by the number of agents present in the environment. Secondly, the agents themselves should be easy to create and their representation should be powerful enough to allow for the creation of a vast number of distinct personalities. In other words, for the architecture to be accessible for non-computer experts, the creation of a simple, but powerful, user interface is required. This applies to both the construction of the agent and the authoring of the story. Addressing these requirements, the problem with the Oz architecture lies mostly in the latter. The personalities of the agents are defined in terms of the sub-functions, or methods, they have available to solve goals.

Since the agent do not use any domain knowledge, all these functions must be constructed from scratch, which is a time-consuming task even for authors that are familiar with the programming language. One simple cat agent, Lyotard, consisted of 2,000 lines of declarative code, indicating the amount of work it would require to complete a fully immersive environment. In previous work, we implemented a simple variant of the Oz architecture in Java, and found that creating even a simple character could take two weeks of work.

This suggests that a simplification of the Oz approach of defining personalities might be needed. To make our agents easier to create, we suggest shifting the focus to the emotional model when defining the personalities. Hopefully, by making the personality directly related to the emotions, we will make the characters more intuitive for the human player to relate to without having to observe their actions (Oz sub-routines) over a longer period of time. We do this simply by assigning what we call "emotion weights" to each individual agent. These weights determine what emotions are focused on when determining the dominant emotion of the agent. For instance, consider this example. The agent had experienced 6 events that have generated "joy" and 5 that had generated "distress", making the overall percentage of each emotion 54% and

46% respectively. Now, assuming we are defining this agent to not have a sunny disposition, we would place larger focus on the emotion "joy". Let us say the weights are distributed towards the negative emotion, i.e. 0.25 to "joy" and 0.75 to "distress" (adding up to 1). After the personality has been applied to the emotions, the distribution would now be adjusted to 28% for "joy" and 72% for "distress"[1]. These two measures work together in describing the emotions and events that the agent is receptive to. Although the emotions clearly state that the agent should be joyful, the personality steers it in the direction of distressful, which becomes its dominant emotion. The personality weights simply place more the emotions that are considered more important in the personality of the agent and less on the ones that are not.

The dominant emotions determine the agent's relationship to the player, and decide which set of dialogue the agent chooses to express this emotion. For a non-computer expert, it is easier to write dialogue to reflect emotion rather than programming behaviour. More importantly, the dominant emotions are the features that decide which of Propp's character this agent will play in the story as story points change and new functions are introduced to the player. The *auditioning* of characters is a direct result of the actions of the player, hence playing her in the focus of the system. The complete internal process can be described by Figure 1.

Creating and Playing Stories

In this section, we shall bring the architecture to life by implementing a complete virtual environment. Our graphic environment of choice is Epic's Unreal Tournament (UT). A decision was made to perform most of the implementation externally to UT and communicate with the game engine using message passing. From here on, we will call the external program which handles the behaviour, the client, and the virtual environment the server. The foundation for the server part of the implementation is the Gamebots tool developed by the Information Sciences Institute at the University of Southern California[2].

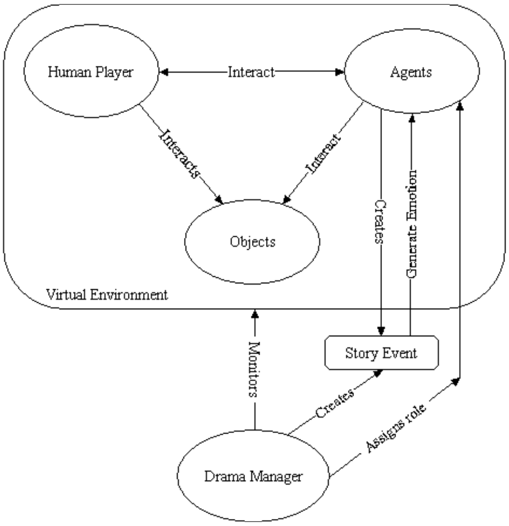


Figure 1

Essentially, the authoring tool (fully written in Java) provides a way of specifying the variables and properties of the elements described previously. It allows the user to define the agents and provide the narrative circumstances around, and the ordering of, Propp's functions (referred to as story points in the software). As one of the goals of this research is to provide a platform for non-programmers to design interactive stories, this section will focus on how the tool works and how software can be used to implement the ideas outlined in the previous sections.

Story Editor

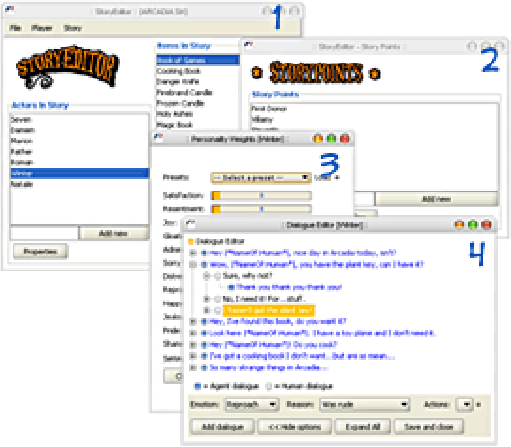


Figure 2.

Initially, the user is given access to an interface only containing the form marked "1". Here the author can add agents (left box) and story items (such as keys, candles, helper's items etc in the right box). Each agent has numerous properties (not shown on the illustration), including name, gender, friends (to create a social network of other agents to

---

[1] Applying the weights: "Joy" = 54% * 0.25 = 13.5%. "Distress" = 46% * 0.75 = 34.5%. Normalisation factor then becomes 13.5 + 34.5 = 48. Applying normalisation: "Joy" = 100 * 13.5% / 48 = 28.125% and "Distress" = 100 * 34.5% / 48 = 72.875%.

[2] http://www.planetunreal.com/gamebots/

communicate with), inventory, wish list (of items to sought for, making them happy if one is found and jealous if they are obtained by the player) and not least: a starting role. The starting role is the Propp role that the author wishes the agent to play at starting time (for instance a family member). For the development of the story, the friend list is perhaps the most interesting feature. In our implementation, the agent will stop and gossip with the other agents in its friend list. It will pass down events that it has seen, events that have triggered emotions, which will in turn trigger events in the agent being told the events. These will trigger other emotions, for instance if a "sad"-gossip/emotion is being conveyed to the receiving agent, it will generate a "sorry-for"-emotion. This way, the actions of the player indirectly influences the way agents she may not yet have communicated with perceive her. The social network is used fully to generate emotions used to audition for roles.

The weights that each agent focuses on when stating their dominant emotions can also be specified through the interface, as partly seen in "3". Here the weights add up to 100% and the authoring tool normalises the weights automatically. As another way of defining a personality, there is an option that allows the author to specify how strongly each agent should react to events that occur in the story environment. This dialogue is similar to "3" but is not shown. However, the main source of emotion creation comes from the agent's direct interaction with the agents through dialogue. Not only is this how most emotions are generated, but also how most of the emotions are displayed.

The dialogue editor can be seen in "4". It is built in a branching tree manner, with the player given a choice of ways in which to respond to each agent dialogue line. The agent line may consist of static text or numerous variables such as the name of agents cast for a particular role, the name of a person in possession of a particular item and a number of other domain knowledge that makes the player perceive the agent as informed and dynamic. Each agent dialogue choice may either trigger an emotion, an event (such as "give key to player") or start the processing of an internal agent goal (for instance, if the player tells the agent to go look for a lamp, the dialogue line may actually create this goal within the agent and start acting upon it). Most importantly, dialogue choices may also change story points, and thus the story can be moved along through dialogue (as well as other ways, as we shall see).

Each *agent* dialogue branch can also trigger an event, but additionally, they can be given a precondition. This way, the agent can choose to speak in a completely differ manner if it is, for instance, sad. Thus, it can display its internal state in a convincing manner. There are numerous preconditions that may be assigned, and if several hold true, one is selected by random statistically (meaning there is a smaller chance that particular branch will be chosen next time). In the current implementation, the preconditions include "X has dominant emotion Y", "X wants item Y ", "X has friend Y", "X has item Y" and "story is in story point Y". The X can either be agent itself or any other agent or role. The conditions can be negated and any number can be applied to a certain branch of the story. The final condition makes sure that the agents give the player information that is related to the current state of the story. This will enable the author to write dialogue that reveals the right information at the right time.

Authoring Roles

Once all the agents and items have been created, the author can move on to designing the story in which these agents will act (although one could have started by creating the story structure, the order is arbitrary). Following Propp, there are two elements that define a story: Story Roles and Story Points (functions).

"Roles" are created in a similar fashion as the agents, with their own unique dialogue, friends (consisting of agents or other roles), emotion weights (a villain, for instance, might have a strong weight towards jealousy) and most importantly – goals. These all ensure that whichever agent is cast for that role steps "into character" and goals in particular move the story along by allowing the agent to complete tasks and act out their internal motivation. At run-time, tasks are split into domain-understandable subtasks that are solved chronologically by a task-solving component. Once all the roles have been defined, the author can assign them to story points.

Authoring Story Points

The architecture allows for any number of story points to be acted out in any order. Hence, one does not necessarily need to follow Propp's ordering. After a story point is given an ID/name (for instance "Interdiction"), the author can decide which roles need to be cast, and which agents should be considered for the audition. This gives the author the possibility of excluding certain agents from being cast for roles, which might be feasible in stories where certain agents are pre-defined to be certain types, e.g. the best friend of the player. Each role can be given a property of either “active” or “passive”, meaning the agent will either move about or stand still. One may want agents that give the most important information to stand still, as to maximise the chance of the player finding them. The author is also given the option of spawning new items (created in the first screen, labelled “1” in the illustration) relevant only to this story point.

There are two ways a new storypoint can be triggered. One is, as we have seen, through dialogue. The other is by specifying a change after a certain number of minutes, an optional property set in each storypoint. This gives the author the chance of creating a paced story, one that does not come to a halt if the player fails to find the correct line of dialogue. Each story can be given some narrative, consisting of either static text or narrative. Finally, one can set a storypoint as the “end point”, meaning it does not advance past the narrative.

In the current architecture, each story point is acted out in a different part of the *Unreal* level, the centre of which chosen at random with a minimum distance specified in the editor. The radius in which agents belonging to a specific story point can stray can also be set in the editor. This will make sure that in a larger story/level, the player does not have to walk an unfeasible distance just to find the relevant characters for a particular part of the story. Following a change in story points, the agents cast to play will start to walk towards the centre. They will not all start walking simultaneously, as it might seem conspicuous. Having these story centres ensures that the story is dynamic and moves from location to location. In future implementations, one can allow the author to choose the location rather them being chosen at random.

Once a story point is triggered, the audition process starts. Here, the architecture looks at the emotion bank of all the agents auditioning that have not yet been given a role. The emotion bank is a list of all possible emotions as well as how many times each emotion has been triggered. The number increases based on the *reaction* weights discussed previously. (For instance, one agent might increase the “sad” emotion by 4 is someone takes an item it wants, and by 10 if the player is cruel to it through dialogue.) The *emotion* weights, as seen in “3”, are then applied to this emotion bank and compared to the emotion weights of each role to find the role with the closest profile. This will be the role that the agent will be cast to, as it is the one that has an internal state/personality closest to the one defined by the role. Once all the roles are cast, the agents have moved towards the storypoint centre and the new items have been spawned, the narrative will be presented to the player and the storypoint will be in effect.

Finally, it is worth noting that the authoring tool throughout the process performs a number of consistency checks. For instance, should the author decide to delete an object; the editor will check for occurrences of this object and inform the author. The story can be saved at any point in the authoring. The file format chosen is *XML* (Extensible Markup Language) format, a robust and logically verifiable format based on a number of international standards.

Story Engine

In essence, the story engine controls everything that occurs in the *Unreal Tournament* environment remotely. Its main tasks include creating agents and moving them around the environment, creating objects and monitoring their pickup, transferring events when there is gossip and eavesdropping occurring and generally all the other tasks not handled by the agents themselves. The 3D environment communicates with the story engine through message passing and allows the author to see the behind-the-scenes details of the story. This gives the author the chance of observing a story remotely (even see the lines of dialogue passed between the gossiping agents), as the story engine can connect to an Unreal Tournament client in any location using the IP address of the machine running the game.

Once a game file has been loaded, each agent will be given an individual tab showing his or her position and sensory information within the environment. In these tabs, one can see and alter their internal state, as well as emotion weights and dialogue. This is particularly useful for debugging and real-time altering of the story. In other words, event intensities, the task list, the dialogue bank, and so on, can be changed and saved real-time. This allows the author to alter the settings of the agent while the simulation is running to rectify errors or make dialogue additions. Additionally, there is a GameMaster tab that gives the authoring a global overview and allows her to see where the agents are in relation to each other. In addition, one can see the sensory range of each agent. Furthermore, the author can change the settings for the story structure, just like in the Story Editor, or set story points if they wish to debug or progress the story. Most of these options, both for the GameMaster and the agent, are for debugging purposes as all the story point changes and generation of emotions are automated. There is a textbox where the messages that are passed between the server and the agent client-side are logged. In addition, one can see the agent moving around the environment in the upper window.

When a story point is set, and agents have been cast for roles, the emotions (after adjustment/weighting) are compared to the personality weights of the roles and whichever agent has the closest match gets the role. Following this, the agent tab will be changed to reflect the role currently assigned. The details of the Story Engine are not necessary for the author as long as care is taken when authoring the story and no debugging is needed. However, aside from the cryptic debug messages, its usage is straight-forward and similar to the authoring tool.

Results

An informal user-evaluation with five users was performed, in which several quotes suggested this could be a successful system. Regarding agency;

"[The actions] affected everything. After being in the game for 2 minutes everyone was gossiping about me, and the things I have done. People were rude just because I talked angrily to someone else. The 'try funny dialogue choices first' approach is not getting you anywhere."

"The most enjoyable part of the experience was seeing my actions and attitude towards players contribute to the story, which became apparent by the way they interacted with me and each other"

"I felt that the decisions I made and actions I took contributed to the direction of the story. I realised I was taking part in a story with direction rather than just running around a world"

Due to the limited number of participants, the results are not conclusive and would have to be confirmed with further user evaluation. However, the feedback seems to suggest that local agency has been achieved and that the participants did feel their actions were reflected in the behaviour of agents. They also stated that they could see the consequences of their actions being manifested in the gossiping of agents. The players expressed a sense of global agency; feeling that they had some control over the direction of the story. This is an improvement with to the Oz project and Façade where the player is given the status as interactive observer rather than protagonist. In our approach, none of the players felt that they were simply observers and stated that their actions were sufficiently expressed in the emerging narrative. The player agency was reduced only in cases where Propp's function required a specific action from the participant for the story to progress. However, the authoring tool allows the writer to give the player a choice even in these situations.

Judging by the user evaluation, the overall moral and narrative of the story was sufficiently communicated to the player and dramatic story structure was maintained. In their retelling of the story, Freytag's Triangle was followed; i.e. the story had a beginning, a middle where some conflict was introduced and a climax was reached and an end where the conflict was resolved. The users stated that they felt the majority of the agents were not only in the environment to gossip and eavesdrop, but also took part in the story, were aware of the events that were occurring and in some cases communicated the narrative. Most expressed that the reasons for events occurring (such as someone become a villain) were logical based on past events. This kind of comprehension is important, because if the users do not understand the premise, the drama and is lost and it becomes difficult to prevent the story from becoming confusing. The players stated that they were immersed in the environment, and became connected with the agents to the extent of being able to describe their personalities. This not only shows that that the structure of the story maintained a sense of drama, but also that the authoring tool written for the architecture was flexible enough to create a number of personality-rich agents. Agents add to the immersive quality of the story, and thus it is important to have created an environment which is scalable and where agents can be written using a simple interface. The agent personalities and emotion were reflected to some degree, but mostly through dialogue and less through action. With respect to previous work, we have managed to maintain a story structure with less work from the author. In other words, since we are using an established storytelling technique, no evaluation function is needed to assess what constitutes a coherent narrative structure, as in the Façade drama manager. Furthermore, compared to Façade, the need for content creation has been reduced, and only the most important plot themes (i.e. Propp's functions) need to be specified by the author.

Contributions of Research

With respect to our original goals, we have managed to develop an architecture that allows for the creation of flexible stories in an environment inhabited by believable and autonomous agents. The architecture is scalable and allows for a large number of agents to coexist and form social relationships; exhibited by their interaction, gossiping and eavesdropping. The use of such agents has proved to be successful as the participants were immersed in the story and felt a connection with the characters involved. By focusing the attention of the agents to the human player, we have managed to increase the feeling of agency and user freedom compared to previous research. The stories can be designed using a tool that allows authors, unfamiliar with programming, to create interactive narratives and personality-rich agents. We have simplified the process of authoring stories, while still maintaining story structure, by drawing inspiration from traditional storytelling techniques. By justifying the motivation behind the behaviour of the characters, we hope to have presented the participant a story where her actions have a consequence on the narrative developed, and seem logical with respect to the premise. We believe we are one step closer to finding the suitable middle way between narrative structure and interactive freedom, and hope to have created a platform upon which future work can be built.

References

Augusto Boal. Theatre of the Oppressed. London: Pluto Press, 1979.

Christoph Bartneck. Integrating the OCC Model of Emotions in Embodied Characters. Proceedings of the Workshop on Workshop on Virtual Conversational Characters: Applications, Methods, and Research Challenges, Melbourne, 2002.

Bryce Courtenay. A Recipe For Dreaming. *Penguin Books*, 1998.

Fred Charles, Marc Cavazza, and Steven J. Mead. Generating Dynamic Storylines through Characters' Interactions. International Journal on Intelligent Games and Simulation, 2002.

Clark Elliott. The Affective Reasoner: A Process Model of Emotions in a Multi-agent System. Ph.D. Thesis, Technical Report No. 32, Institute for the Learning Sciences, Northwestern University, Evanston 1992.

Gamebooks (2004). The Extensive Gamebooks Archive. www.gamebooks.org, 2004.

Andrew Gordon and Nicholas Iuppa. Experience Management Using Storyline Adaptation Strategies. Proceedings of the First International Conference on Technologies for Digital Storytelling and Entertainment, 2003.

Raymond Lang. A Declarative Model for Simple Narratives. AAAI Fall Symposium on Narrative Intelligence, 1999.

Liquid Narrative Group. Computer Science Department, North Carolina State University, www.liquidnarrative.csc.ncsu.edu, 2004.

Michael Mateas. An Oz-Centric Review of Interactive Drama and Believable Agents. Technical Report CMU-CS-97-156, School of Computer Science, Carnegie Mellon University, Pittsburgh, 1997

Michael Mateas and Andrew Stern. Façade, an Experiment in Building a Fully-Realized Interactive Drama. Game Developers Conference, Game Design Track, 2003.

Murray, J. (1997).. Hamlet on the Holodeck (The Future of Narrative in Cyberspace). MIT Press.

Andrew Ortony, Gerald Clore and Allan Collins. The Cognitive Structure of Emotions. Cambridge University Press, 1988.

Oz Project. Carnegie Mellon University. www.cs.cmu.edu/afs/cs.cmu.edu/project/oz/web/oz.html, 1989-2002.

Vladimir Propp. The morphology of the folk tale (L. Scott, Trans.). Bloomington, IN: Research Center, Indiana University, 1958

Mark Riedl and Michael Young. An Intent-driven Planner for Multi-Agent Story Generation. Proceedings of the 3rd International Conference on Autonomous Agents and Multi Agent Systems, 2004.

Mark Riedl, CJ Saretto and Michael Young. Managing Interaction between Users and Agents in a Multiagent Storytelling Environment. Proceedings of the Second International Conference on Autonomous Agents and Multi-Agent Systems, 2003.

Understanding emotions in drama, a step towards interactive narratives

Sandy Louchart[1]
[1]Centre For Virtual Environment
University of Salford
Salford M5 4WT
s.louchart@salford.ac.uk

Ruth Aylett[2]
[2]MACS
Heriot-Watt University
Riccarton, Edinburgh EH10 4AS
ruth@macs.hw.ac.uk

Sibylle Enz[3]
[3]Institut fuer theoretische psychologie
Universitaet Bamberg
sibylle.enz@ppp.uni-bamberg.de

Joao Dias[4]
[4] INESC-ID
IST, Taguspark
Porto Salvo, Portugal
joao.dias@gaips.inesc-id.pt

Abstract

This paper discusses the importance of emotion modelling if one is to deal with interactive drama. As a position paper, it describes the emotion mechanisms in place in traditional drama and considers the possibilities to transpose these in regard to interactivity. We also, in this document, look with a critical eye at the necessary changes to be made in the general perception of drama, more particularly, the role of the user if dramatic enacting has to be rendered interactive. Beside the very much-discussed topic of narrative considerations, we draw attention in this document to the necessities of considering the preponderant role that emotions have in regard to the articulation, comprehension and perception of drama. For these reasons, this paper aims at understanding the emotional dynamics involved in interactive storytelling and establishes links between emotional rendering of information and the appreciation of the drama itself.

1 Introduction

With growing technology allowing for faster processing, higher graphic definition and greater realism, digital entertainment seems likely to represent the next step in entertainment. Whilst the video-games industry has already firmly established itself in our everyday life and culture, game developers are now starting to reach new markets and new audiences. In a society relying more and more on computational technology and with a population increasingly computer-literate, everything seems set for the next generation of digital entertainment to finally take off. Indeed, we have witnessed in the last few years a growing acceptance of the digital media to the point that digital entertainment does not seem anymore to be restricted to certain educated communities (i.e video-games, digital art). The range of video games has also been broadened and new audiences are being targeted (i.e. the women audience for Nintendogs) with clear apparent success. Also, sign of the times, in a typical cinematic fashion, (i.e. more often than not, with no scientific grounding), contemporary science fiction has also been exploring the theme of Virtual Reality in several successful productions (i.e. the matrix [1], existenz [2], 13th floor [3]). The general public seems also to be well aware of the realities and possibilities of the virtual reality technologies.

When combining the potentialities of diverse technologies, we seem to be a small step away from an entertainer's dream. Indeed, one could imagine with not too much difficulty, one day living or experiencing an interactive drama from the inside, within the boundaries of a realistic-looking virtual environment. One could also think of an environment where a user could be free to explore a digital experience and play an active part in an immersive narrative production.

Despite futuristic projections over the possibilities of developing technologies, we, however, have to question whether or not it is indeed the right time for such a cultural and technological revolution in entertainment to take place. Are the digital media companies ready and willing to develop digital experiences with the capabilities to move a user emotionally in the same manner as a movie does? Is this all technically feasible? Are we, as part of a potential audience, ready to interact with the characters of such dramas? For a certain number of very good reasons, very few people would risk predicting how close we are from such an entertainment vision. Some might reckon 5 or 10 years, others might think it much further in the future. As of today, there is however one thing we know with certainty. The development cost (i.e. research, technological innovations) for such a digital experience would be far too great for any video-games company to even attempt it.

As often with technology related innovations, making the vision a reality depends very much on the availability and costs of technology. Moreover, there are other obstacles far greater than technological issues to contend with. Greatest of them all is the ability to do as classic narrative media (i.e. cinema, literature, theatre) do and generate powerful emotions for spectators or readers. We believe that the overall issue of emotion generation should be at the centre of investigation into engaging digital interactive experiences. In addition to studying the different links between emotions and drama, we will look, in this paper, at identifying the emotional mechanisms involved in the assessment and interpretation of dramatic narratives (i.e. emotional empathy, believability). We will also consider with particular interest certain issues surrounding the overall concept of interactive narrative, and more particularly, the importance of understanding emotional dynamics when designing such applications.

1 Interactive dramas and the case for considering emotion

When reflecting on the nature of dramas and the way we appreciate and remember them, one must recognise the tight ties that link drama and emotions together. This is indeed a very well documented relationship [4], and we do not here attempt to offer anything new on that topic. What intrigues us is this relationship seems relatively neglected in research into interactive narrative. On the other hand the agent-based approach undertaken in our emergent narrative research [5,6,7] was bound in our view to consider the representation of emotion as an integral part of intelligent agent technology. However we found that there have been surprisingly few publications, at least in this particular research field, on the preponderant role played by emotion in the understanding and communication of drama.

Much has been said about the major changes required to make drama interactive in terms of structure, perspective, narrative articulation, protagonist roles and story management [8,9,10,11,12]. In comparison, very little has attempted to assess the emotional bonds we, as part of an audience, build with characters. Emotional empathy is however essential to audience bonding with characters in any drama; it is what makes us like or dislike a particular character and it is also a tool used by directors to build on cinematic effects and techniques such as surprise, suspense or tension. Our ability to emotionally relate to characters also partially determines the degree to which we enjoy or dislike a whole narrative production, be it a novel, a film or a theatre. Indeed, we tend to remember movies for instance; according to how much they made us laugh or cry. In the same way we like actors for their talents in making us believe in the characters they play and for their abilities in generating emotions via a character interpretation as if the character were a real person. From the criteria of predicted interest graph charts in cinema magazines **[figure 1]** [13] to comments made by moviegoers at the exit of a showing (i.e. it made me laugh, cry, - kept me on edge all along), emotional response summarise the whole impact of a piece.

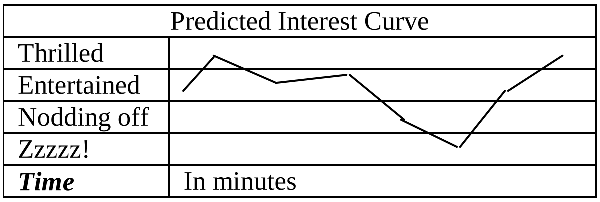


Figure 1: Example of movie evaluation in film magazines

Emotion seems so important in determining the outcome of a narrative experience that one should indeed question the worth of evaluating stories purely from an approach based on the interest of the spectator in the movie. In fact would seem that an emotion-based assessment might reproduce more accurately the impact a production has on its subject.

Independently of the particular personality or emotionally sensitivity behind individual emotional reactions to a narrative production, there are techniques and mechanisms built into a scenario or story whose purpose is to influence the audience into certain affective states at different points within the unfolding of the drama. These mechanisms have been developed specifically for each narrative medium. The techniques used to generate tension and terror for instance in literature are different from the ones used in cinema. Where literature would most certainly rely on describing the internal emotional state of terror of the character a movie would rely on more visual clues such as shadows or a close-up shot of the terrified character's eyes.

The shift in perspective from a spectator in traditional drama to a participant (or user) in interactive drama, coupled with the introduction of interactivity, suffices to define Virtual Reality (VR) as a new emerging narrative medium. We believe that, as with the other narrative media previously mentioned, VR will need its own specific set of techniques and mechanisms in order to manage itself structurally, which is what the virtual storytelling community is currently trying to achieve [7,8,9,10,11,12]. Also, it will certainly need to establish mechanisms and techniques to manage emotional input, outputs and emotion generation with great efficiency if it is to develop.

2 Emotional empathy and the understanding of drama

The way, in which we relate to drama, as spectators, is determined and influenced by many factors on different levels (i.e. context, personal history, political affiliations, cultural environment, beliefs, social situation etc.). It is up to the director or producer to assess and tune up a potential scenario in order to generate the scenario-specific right types and levels of emotions. The level of emotion generated is highlighted there and shows how difficult such an exercise is. Not only the types of emotions must be prompted in the first place in an efficient manner if they are to reach the majority of the audience, but they also should not rise to a level where they become overwhelming. For instance, many people would have found it highly controversial if a terrorist plane hijacking drama had been released straight after the tragic event of 9/11. The potentially high emotional response generated by such a production would have been too great of a risk for any production company to even consider such release.

Independently to the storyline and overall plot, as members of the audience, our reaction to a narrative production relies on how we feel about it, on how much we felt entertained, frightened, sad or amused. These emotions are brought to the spectator via several narrative elements, mainly, the context, the characters and the plot events. The emotion planning for any production could actually be seen as a combination of two different factors. The techniques generating these emotions vary and are balanced according to the different production genres.

One particular approach is to provoke an emotional response from the spectator by manipulating the image, the sound and/or context. This role is assumed by the director and lies in his/her ability to produce images or sounds prompt in raising certain emotions for the user. A very good example of exploiting sound to this effect is the manner in which it is used in Jaws [14] and how effective the theme tune was actually in generating apprehension and tension whilst the camera would only show an empty see. The imagery in Seven [15] also reflects this by constantly showing in the background rainy weather. It played a big part in setting up a slightly depressing feeling attached to the movie, so that the spectator would be more prompt in developing negative and dark emotions as the events unfolds. Finally, French director Laetitia Colombani illustrated the importance of contextualisation in regard to emotional responses from the audience in her 2002 debut movie He loves me, he loves me not [16]. In this movie the audience witness a love story depicted from the angle of one of the protagonist and is lead into forging an opinion and emotional bound with the characters, only to find out in the other half of the movie that these emotions were indeed biased by the context in which the characters were presented. These techniques are common in today's cinema and can be observed in many movies.

The other approach commonly used in drama for provoking emotional responses from the audience lies in characterisation, the art of writing interesting and compelling characters. This process assumes that a well-designed character would, if properly interpreted, be able to carry the audience emotionally through its own different emotional states when required by the director (i.e. if the director is not using context information to manipulate the audience's feelings). Such practice lies entirely on our natural tendency to feel for others and our "ability to understand and share in another's emotional state or context" [17]. This natural aspect of the human being is described as affective empathy. This can be demonstrated when analysing an audience's reaction to the tragic death of Alfredo (Philippe Noiret) in Giuseppe Tornatore's "Cinema Paradiso" [18]. Although viewers have bounded with the character emotionally, it is not their sadness for the loss of the character that they respond to. Tears from the audience are indeed reflecting upon the terrible melancholia and sadness that inhabit Alfredo's friend, Salvatore (Salvatore Cascio) rather than the death of Alfredo itself.

Empathy is very much the mechanism that allows a spectator to fully engage with a character. It allows a spectator to connect emotionally with a character and understand and sometimes feel the same emotions as the ones inhabiting the character. Writers continuously exploit such phenomenon in order to impact emotionally on the reader or spectator. Characters are purposely written in a way that they can display a rich emotion set and engage with the spectator's natural ability to care for others.

3 Emotional considerations for Interactive narratives

Building interactive narratives certainly represents a challenge in many aspects and appears not to be only restricted to the sole domains of narratology and story structures. As described in the previous sections, we believe that the emotions felt by the user are of prime importance in producing compelling and engaging interactive dramas. Although we argue in this paper for the necessity to design generative mechanisms that would allow a narrative system to reflect on the user's emotion state, it is however important to consider the meaning of interactivity in narratives. There are indeed two different ways to view the term interactive narrative. One could see there the possibility for a user to create a story or participate in the authoring of a story, whilst others could interpret the term from a character-based perspective in which the user is playing an active role in the unfolding of the story. In fact the very nature of the necessary changes to be made to the way we approach story design in regard to emotions depends directly on the type of interactivity considered. Indeed, the emotion requirements for an authorial consideration of interactivity will necessarily differ from a character-based approach. Whereas in the first approach, the users would be involve in designing and creating emotional planning and mapping for a production, they would be, in the case of a characterisation, experiencing and acting the drama from an insider's perspective, with all the emotional involvement it requires.

An authorial take on interactive narrative would aim at providing the user with tools and techniques that would allow for the generation of certain emotions from a static point of view, in a similar manner to the way in which cinema manages emotions such as tension, fear, passion or laughter. Since in such an approach, the users are indeed playing the role of a director. One could imagine that an interactive storytelling system would aim at interacting with the users in such a way that they ultimately produce with the help of the system a story involving characters, sets and actions that could rival on the same emotional scale the impact of its cinematic counterparts. Although complex to build, such system could probably be developed as of today as either an entertaining and compelling experience from a user's point of view or as a professional tool in assisting creative writing and scenario or plot development. The fact that such approach does not differentiate itself too much from already existing practices in the cinematic domain means that the techniques and practices necessary for the development of such systems are known and can potentially be applied or modified to fit the requirements for an automated generation via the use of technology. An intelligent system, from an emotional perspective, would aim in either identifying emotions from user input (i.e. actions, plot event) and suggest a course of actions in order to reach such actions in a scenario along with the intended emotions; or analyse user input and drive the scenario creation from an emotional point of view, recommending the next emotion or set of emotions to aim for. Such an approach implies although not explicitly, production of dramas that would conform to the traditional concept of storytelling as seen in cinema, theatre or literature. Works in this area are generally sharing a vision of drama where the audience occupies a rather static role (i.e. no possibilities for intervention or interfering in the plot development) and onto whom the dramatic performance is displayed and presented. The fact of the matter is that, currently, research in the area tends to be divided in between authorial and character approaches. However, one should not dismiss one for another in the sense that in one way or another, research carried out in one sub-branch of the field will benefit the other and vice-versa. From a character-centred perspective, it is an interesting prospect to indeed contemplate the possibilities to at some point in a near future, use authoring tools for the generation of interactive narrative that would manage characters and story world creation in a similar manner in which authorial approaches are today undertaking research in the field. Indeed, such a division could prove to be a beneficiary factor for the whole community as time unfolds.

The consideration of a character-based take on the interactive narrative issue brings however more complexity to the problem. The main obstacles to successfully communicate emotions to the users lie in the fact that users are by definition, in this concept, non-static and that the lack of apparent structure in the unfolding story does not allow for the use of most traditional emotion building techniques such as the ones developed for static audiences (i.e. cinema, theatre, literature). Indeed, if we take for instance the way in which tension is built in a movie (i.e. via camera angles, music, and montage), it seems that the very idea of a non-static users that could possibly not witness any of the tension building elements because exploring other areas of the story-world, does not allow for the use of such techniques. This feeling is reinforced when one considers that emotion-building techniques are integrant to the process of scenario development and that they are determining the final form of the scenario and the way in which the storyline will unfold. With such a prominent role in the development of a scenario, it seems difficult to conciliate such practices with the idea of emerging and non-deterministic storylines and structures. Attempts to address this particular issue can be observe in video games and the manner in which emotions and plot events are in no subtle ways forced down to the user. Although players can enjoy a certain freedom in terms of movements, the story structure is generally such that players will not be able to progress in the game unless certain events have happened. When it does, the players are generally force into actions or have to reverse into a static role, losing any freedom allocated to them by the developers, whilst a cinematic or scripted animation is played to them. Emotions are generally conveyed with no subtlety via the use of cinematics. The way in which emotions are generated and managed in video games provides reasons for the research and development of narrative and emotion building techniques and solutions adapted to an interactive user or player.

It is apparent that the commonly used techniques with regard to emotion generation and communication in drama do not provide a viable and ideal solution to the introduction of interactivity from a non-authorial perspective. We believe that character-centred concepts should be looking at the characters' internal states of emotions and their reasons, causes and consequences in order to communicate emotions and feelings rather than relying on agency techniques. Indeed, although plot event or even random narrative events should still be means of generating interest in an unfolding story, the focus should be on the reaction of the characters rather than the overall story. The fact that the users are immersed in the story-world and have no access to the overall plot, should allow for the development of story or interest relationships between users and other characters hence, providing the users with an understanding of characters' motivations, personalities or back stories. In a sense, what is expected from the users is not to grasp the overall plot described to them and absorb the drama emotionally as we are used to in films or books, but to actually understand the characters emotionally as well as strategically in regard to a real-time unfolding narrative. By bringing the users closer to the characters, we argue for a shift of emphasis on the means of conveying emotions to the users. Indeed, we argue for an active role of the users where they do not only receive the information from visual, audio and contextual clues, but instead interpret and understand the emotions and feeling of characters and personal contexts, while still being conscious if the impact of their own actions on others. Our emergent narrative system focuses on making the transition between the passive role of a spectator and the active role of a character.

In order to achieve success with such approach, one should regard to main research topics with interest. Particular interest should be oriented towards the way in which the character or agent in case of computational application, assess and process emotions. In general terms, independently of the acting method followed, a good actor is able to feel the same emotions of his/her character and express them in front of a camera or on stage. More often than not, the quality of an interpretation depends on the actor's ability to identify with the character emotionally. Therefore, it is conceivable that a virtual actor (i.e. intelligent agent), if in a position to interpret and feel emotions in regard to a particular character (i.e. personality, emotion tendencies, emotional responses), could indeed make decisions that would reflect on its internal emotional state in the same way an actor would do in acting a given scene. Although, lots of ground has been covered in the last few years in emotion modelling and intelligent technology, it seems that it still would take a lot of effort and investment in order to get an agent reacting and behaving convincingly, in the same way an actor would. However, the technology is currently being developed and such vision appears totally feasible. An agent-based approach would allow for a dynamic assessment of situations and stimuli prompt in generating emotions and would also be a determining element in managing context related actions or reactions. We believe that an agent-based solution would be particularly suitable for this type of application as opposed to a more generic framework that would manage all of the characters. It would be more complex for a generic, non-agent framework to actually keep track of feelings and emotions for each character in a given story experience and process them in real-time. Affective agents have been built to this purpose in the European framework V project VICTEC [19] and make their decisions according to their emotional state and their feelings within each other. We will describe some of the implementation work on these agents in the next section.

The other important consideration in terms of emotion in regard to a character-based approach to interactive storytelling lies in the way user emotions are assessed. As far as research in the field is concerned, it doesn't appear as if there are any other solutions than assessing the user's emotional state via its action-decision behaviour. Indeed, for the characters of such interactive drama to react emotionally to events and user interventions, they must therefore be able to interpret and analyse the user's decisions and actions. This approach is similar to the one we ourselves undertake everyday when making decisions. We make decisions in regard to our knowledge, mood and needs. This can be simulated and reproduced within an intelligent agent providing that it has been given the means to interpret user input and refer to its own emotional state in order to react or build plan in regard to internal goals. For this to be possible, one must consider emotion modelling and implementation within an agent-based framework.

There are grounds in determining the emotional impact of actions or plot events within an agent framework prior to a performance. Also, although this approach allows the characters to get a feel of the user's emotional state and allow them to display a certain number of emotions and behaviours seemingly adapted to the situation in real-time; one could raise the issue of contextualisation and highlight the potential problems linked to the use of deterministic reactions.

4 The FearNot! agents, building emotionally driven characters

The research work carried out for the VICTEC project aimed at building an agent framework whose functionalities would allow for the creation of emotionally driven characters, with a particular view towards interactive drama and virtual storytelling. The agent architecture **[figure 2]** developed for the project focused on the importance of emotions in regard to action-selection and decision-making mechanisms. The agents in the FearNot! (Fun with Empathic Agents to Reach Novel Outcomes in Teaching) Software are emotionally driven and react or plan their next actions in regard to their dominant emotional state. Such an approach allows for the articulation of an emergent narrative approach [5,6,7] and provides flexibility in regard to story articulation as envisaged in this particular narrative concept.

In reference to what has been mentioned already in this paper, we believe that for such a character-based system to function successfully, the agents need to be able to relate to each other emotionally. This particular idea is indeed at the centre of the FearNot! agent architecture and has been implemented in regard to the emotion definition presented by Ortony, Clore and Collins (OCC) [20].

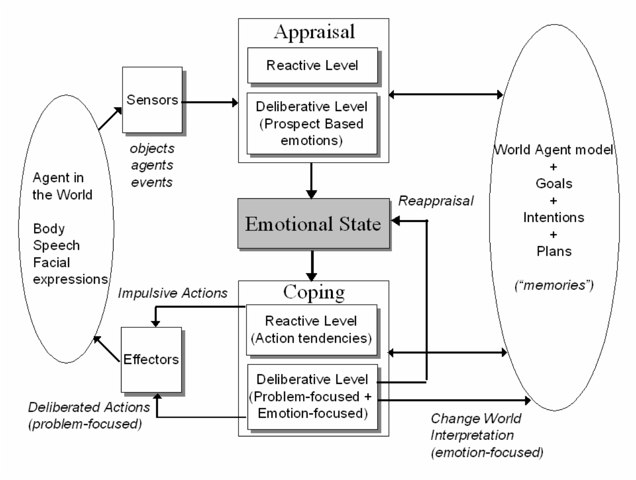


Figure 2: The FearNot! Agent architecture

In this architecture, each agent or character in the world perceives the environment, through a set of sensors (allowing the perception of events, objects, etc. in the world) and acts on the environment though its effectors, allowing different actions to be performed (for example, a bully may hit the victim and the victim may cry). Upon receiving a percept (for example, be the presence of another agent or an object, or even an action from another agent) the agent appraises its significance and triggers the appropriate emotions. Additionally, if a goal has become active, it will add a new intention to achieve the active goal.

The appraisal process feeds the resulting emotional state into an action-selection mechanism at two different levels: that of action-tendencies and that of coping behavior. For example, if the victim character starts to cry when bullied, it is not because s/he has a goal that involves crying – this is an innate reaction to a particular distressed emotional state and the inability to fight back. On the other hand, other actions, such as begging the bully to stop, do result from the internal goals of the agent and are planned.

This second layer defines two kinds of coping: problem-focused coping, involving planning and acting to achieve goals; and emotion-focused coping in which the character's interpretation of the environment is altered. For example, an agent that feels distressed by being unable to achieve a given goal, may lower the goal's importance as a way of reducing its distress. In this way, emotions will not only influence the agents' reactive behavior, but also guide the planning process, since emotional focused coping changes the agent's interpretation of its plans. The continuous partially ordered planner used in FearNot! selects the intention associated with the currently most intense emotion from the intention structure.

By assessing and reacting on each other's action and reactions, the agents in FearNot! truly interpret their role in the same way an actor would interpret a character. Each agent is defined according to its role in the drama and its personality (i.e. emotional triggering thresholds) and is attributed a set of actions, goals and emotional reactions. These are developed and written in regard to the agent's personality and role.

5 Conclusion

In this paper, we have highlighted the necessity to investigate the mechanisms needed for the generation of emotions for a spectator or user. We believe that this topic is highly relevant to the research on interactive storytelling and interactive dramas. Whereas it is certain that progresses are being made in this particular field on the form of the interactive drama and on mechanisms adapted to user interventions, there are still many questions and issues unsolved with regard to the content of the drama. Moreover, the means to reach the user emotionally have not yet been studied and should help in determining techniques and mechanisms in order to reach the user emotionally. We are hoping to conduct some test based on the VICTEC agent framework to illustrate this point in a near future.

Acknowledgements

References

[1] http://www2.warnerbros.com
[2] http://www.allianceatlantisvivafilm.com
[3] Centropolis Film Productions
[4] Bordwell D. Some theories of narration.Chapter in: Narration in the fiction film. Routledge. London. 1986
[5] Louchart S, Aylett R, 2004. *Narrative theories and Emergent Interactive Narrative*. IJCEELL Journal-V14 N6 pp 506-518.
[6] Louchart S, Aylett R, 2004. *The emergent narrtive theoretical investigation*; Proceedings Narrative and Learning Environments Conference NILE04 pp 25-33
[7] Aylett R, Louchart S, 2003. *Towards a narrative theory of VR*. Virtual Reality Journal, special issue on storytelling, Volume 7 January 2004
[8] Cavazza, M; Mead, S; Charles, F. 2001. Characters in search of an author: AI-based Virtual Storytelling.International Conference on Virtual Storytelling 2001
[9] Szilas N, Marty O, Rety J-H (2003) Authoring highly generative interactive drama; Proceedings Virtual Storytelling, Using virtual reality technologies for storytelling ICVS 2003 pp 37-46
[10] Riedl M, Young M, From Linear Story Generation to Branching Story Graphs, First International Conference on Artificial intelligence and Interactive Digital Media, 2005
[11] Michael Mateas, Andrew Stern, Structuring Content within the Facade Interactive Drama Architecture First International Conference on Artificial intelligence and Interactive Digital Media, 2005
[12] Swartout, W and al. Toward the Holodeck: Integrating Graphics, Sound, Character and Story. *Proceedings of 5th International Conference on Autonomous Agents* (Montreal, Canada, June 2001).
[13] http://www.totalfilm.com/
[14] http://www.universalpictures.com/
[15] http://www.newline.com/
[16] http://www.tf1.fr/
[17] Cohen, D. & Strayer, J. (1996). Empathy in conduct-disordered and comparison youth, Developmental Psychology 32(6), 988-998.
[18] ARIANE Distribution
[19]http://victec.org
[20] A. Ortony, G. Clore, and A. Collins. *The Cognitive Structure of Emotions*. Cambridge University Press, New York, reprinted 1994 edition, 1988.

Learning Behaviour-Selection Algorithms for Autonomous Social Agents living in a Role-Playing Game

Maria Malfaz Miguel Angel Salichs
RoboticsLab, Carlos III University of Madrid
28911 Leganés, Madrid, Spain
mmalfaz@ing.uc3m.es salichs@ing.uc3m.es

Abstract

In this paper, we describe and show experimental results of a control architecture of behaviour selection designed for autonomous and social agents. Using a role-playing game we have created a virtual world where the agent interacts with different objects and other agents to survive. This game was built using a text based MUD (multi-user dungeon). The control architecture has a motivational model, and the motivations perform homeostatic control of the internal state of the agent. Since our goal is to implement this architecture on a social robot, we consider a social drive. The behaviour selection is learned by the agent using a Q-learning algorithm while there is no interaction with other agents. In situations where social interaction exists, different multi-agent reinforcement learning algorithms are applied. All these learning algorithms use the variation of the wellbeing of the agent as the reinforcement. Therefore, in this architecture behaviours are not selected to satisfy the goals determined by a dominant motivation but to optimize the wellbeing of the agent.

1 Introduction

The goal of our project is to develop social robots with a high degree of autonomy. The social aspect of the robot will be reflected in the fact that the human interaction will be not considered only as a complement of the rest of the robot's functionalities, but as one of the basic features.

For this kind of robots, the autonomy makes them to behave as if they were "alive". This feature would help people to think about these robots not as simple machines but as real companions. Evidently, a robot that has his own "personality" is much more attractive than one that simply executes the orders that he is programmed to do.

This paper presents a control architecture for an autonomous agent based on motivations. In this first stage of the project, due to the obvious physical difficulties of making experiments on a real robot and on a real environment, we decided to implement our architecture on virtual players, who "live" in a virtual world. This game gave us the possibly of creating 2-D different environments to play in, as well as a graphic interface.

In our frame, the player uses reinforcement learning algorithms to learn its policy while interacts with its simulated world. The reward for this learning algorithm will be the variation of the wellbeing of the agent, due to the previous selected behaviour, calculated at each step of the process. This wellbeing is a function of the internal needs of the agent (drives). Therefore, the behaviour selection is a motivation-based process, but it is the agent by itself who learns which behaviour is better for each situation. This means that the agent will learn which behaviours will produce the highest increase of the wellbeing at the current agent's state. Therefore, behaviours are not related with a specific dominant motivation. i.e. The agent can "eat" although the "drink motivation" is the dominant one, but obviously, the action "drink" will produce a higher positive variation of the wellbeing.

This idea of using the wellbeing of the agent as the reinforcement in the learning process for behaviour selection has been also used by Gadanho in the ALEC architecture, obtaining quite good results (Gadanho,2003).

The paper is organized as follows. Next two sections give a description of this control architecture and the reinforcement learning algorithms. Then the following section introduces the experimental setting chosen to implement our control architecture. To continue, we will show the experimental results of the agents living on the virtual world. Finally, we will draw future works and conclusions.

2 Control Architecture

An independent system should not have to wait for someone to maintain, succour, and help it (Frijda and Swagerman,1987). Therefore, an autonomous agent should be capable of determining its goals, and it must be capable of selecting the most suitable behaviour in order to reach its goals. Similarly to other authors (Avila-Garcia and Cañamero, 2004), (Breazeal, 2002), (Gadanho, 2003), (Velasquez, 1998), our agent's autonomy relies on a motivational model. Figure 1 shows this proposed control architecture for behaviour selection.

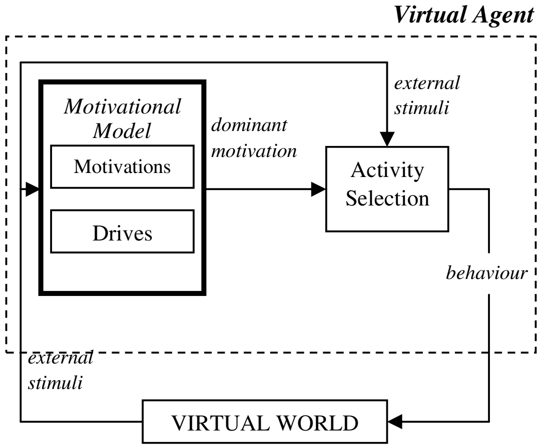


Figure 1: Control architecture for autonomous agents

2.1 Motivational Model

Motivations can be seen as homeostatic processes which maintain a controlled physiological variable within a certain range. Homeostasis means maintaining a stable internal state (Berridge,2004). This internal state will be parameterized by several variables, which must be around an ideal level. When the value of these variables differs of the ideal one, an error signal occurs: the drive. These drives constitute urges to action based on bodily needs related to self-sufficiency and survival. External stimuli, both innate and learned, are also able to motivate and drive behaviour (Cañamero,1997).

In order to model motivation we used the Lorentz's hydraulic model of motivation as an inspiration (Lorentz and Leyhausen,1973). Lorentz's hydraulic model is essentially a metaphor that suggests that motivational drive grows internally and operates a bit like pressure from a fluid reservoir that grows until it bursts through an outlet. Motivational stimuli from the external world act to open an outflow valve, releasing drive to be expressed in behaviour. In Lorenz's model, internal drive strength interacts with external stimulus strength. If drive is low, then, a strong stimulus is needed to trigger motivated behaviour. If the drive is high, then, a mild stimulus is sufficient (Berridge,2004).

Following this idea, the intensity of motivations (M_i) is a combination of the intensity of the related drive (D_i) and the presence of the related external stimuli (w_i), as it is expressed in the following equation:

$$M_i = D_i - w_i \tag{1}$$

The drives used are: Energy Thirst, Health and Sociability. And their related external stimuli are: food, water, medicine and other agent. These drives have been selected taking into account the role of the agents in the virtual world used to implement our architecture. A typical player in the selected game needs to eat and to drink in order to survive. The health and social drive has been added to make the working frame more complete. Since our final goal is to construct an autonomous social robot, it must show social behaviours. Therefore, as it is shown, social motivations are included as robot's needs.

The ideal value for all the drives is 0. At each simulation step these values are incremented a certain amount. Obviously, these drives are reduced to zero, never a negative value, when some behaviours such as eat are executed. The external stimulus, as we can observe, are the different objects that the player can find in the virtual world during the gametime. If the stimuli is present the value of wi is 1, otherwise is 0.

According to (1), the intensity of a motivation is high due to two reasons: 1) the correspondent drive is high or 2) The correct stimulus is present. The dominant motivation is the one with the highest intensity. This model can explain the fact that we do not always eat when we are hungry but because we have available food in front of us. We have also introduced activation levels (L_d) for motivations such that:

$$\begin{aligned} &\textit{if } D_i \le L_d \textit{ then } M_i = 0 \\ &\textit{if } D_i > L_d \textit{ then } (1) \text{ is applied} \end{aligned} \tag{2}$$

Therefore the possibility of no dominant motivation exists.

2.2 Wellbeing

As shown in (3), the agent's wellbeing is a function of the values of the drives (D_i) and some "personality" factors (α_i).

$$Wb = Wb_{ideal} - (\alpha_1 D_{energy} + \alpha_2 D_{thirst} + \alpha_3 D_{health} + \alpha_4 D_{social}) \quad (3)$$

Wb_{ideal} is the ideal value of the wellbeing of the agent. The personality factors weight the importance of the values of the drives on the wellbeing of the agent. The value of the wellbeing and its variation (ΔWb) are calculated at each step. The variation of the wellbeing is calculated as the current value of the wellbeing minus the wellbeing value in the previous step. Therefore, according to (3) the biggest positive variation will occur when the drive related to the dominant motivation is reduced to zero.

2.3 Behaviour Selection

For other authors (Avila-Garcia and Cañamero,2002), (Avila-Garcia and Cañamero,2004), (Cañamero,1997), action selection consists in making decisions as to what behaviours to execute in order to satisfy internal goals and guarantee survival in a given environment and situation. These internal goals are determined by a dominant motivation and therefore, behaviours are linked with motivations.

In this frame, behaviours are going to be related to the objects not to the motivations. This means that the agent can perform a set of limited actions with each object:

- Food: Eat/ Get/ Go to a room where there is food/ Explore.
- Water: Drink/ Get/ Go to a room where there is water/ Explore.
- Medicine: Take/ Get/ Go to a room where there is medicine/ Explore.
- Playmate: Steal/ Give food-water-medicine from/to other player / Chat/ Explore.

The interaction with another player is at the same level as with the rest of the objects. At each moment, depending on the relation of the agent with the objects, some behaviours can be performed and others not. Only the "explore" action can be performed at any time.

Among the previously mentioned behaviours there are some of them that reduce (ΔWb>0) or increase (ΔWb<0) some drives:

- Eat food: reduces to zero the Energy drive.
- Drink water: reduces to zero the Thirst drive.
- Drink medicine: reduces to zero the Health drive.
- Chat: reduces to zero the Social drive.
- To be taken something by other player: increases a certain amount the Social drive.
- To be given something from other player: reduces a certain amount the Social drive.

The agent interacts with the environment through the objects and the execution of some behaviour that modify the wellbeing of the agent. Obviously, the biggest positive variation of the wellbeing will be obtained when the selected behaviour helps to reduce the drive, related to the dominant motivation, to zero.

In this architecture the agent learns, using different reinforcement learning algorithms, the best behaviour at each step using the variation of the wellbeing as the reward. Therefore, in this architecture behaviours are not selected to satisfy the goals determined by the dominant motivation but to optimize the wellbeing of the agent.

It is important to note that finally, the agent will learn that when the dominant motivation is Eat he must select among the behaviours related with the object food, instead of those related with water or medicine. What is new about our approach is that those behaviours were not previously linked with that motivation.

3 Reinforcement Learning

3.1 Introduction

Reinforcement learning (RL) is about learning from interaction how to behave in order to achieve a goal. The agent and the environment interact over a sequence of discrete time steps. The actions are the choices made by the agent; the states are the basis of making the choices; and the rewards are the basis for evaluating the choices. Everything inside the agent is known and controllable by the agent; everything outside is incompletely controllable but may or may not be completely known. A policy is a stochastic rule by which the agent selects actions as a function of states. The agent's objective is to maximize the amount of reward it receives over time (Sutton and Barto,1998).

Q-learning is a value learning version of RL that learns utility values (Q-values) of state and action pairs Q(s,a). It provides a simple way for agents to learn how to act optimally in controlled Markovian domains (Yang and Gu,2004).

The theory of Markov Decision Processes (MDP's), assumes that the agent's environment is stationary and as such contains no other adaptive agents (Littman,1994). Therefore, while the agent is not interacting with the other agent, we will consider our virtual world as a MDP environment.

On the other hand, if the agent is interacting with other player, the rewards the agent receives depend not only on their own actions but also on the action of the other agent. Therefore, the individual Q-learning methods are unable to model the dynamics of simultaneous learners in the shared environment.

Currently multiagent learning has focused on the theoretic framework of Stochastic Games (SGs) or Markov Games (MGs). SGs appear to be a natural and powerful extension of MDPs to multiagent domains (Yang and Gu,2004).

Taking into account these considerations, in the proposed architecture the agent will use the standard Q-learning algorithm as the RL algorithm when the agent is not interacting with the other player. In the case of "social" interaction, the agent must use a multiagent RL algorithm. The following subsections explain in more details these two scenarios.

In our system, the state of the agent is the aggregation of his inner state S_{inner} and the states S_{obj} related to each of the objects, including external agents, which can interact with him.

$$S = S_{inner} \times S_{obj_1} \times S_{obj_2} \cdots \tag{4}$$

For the RL algorithms the states related to the objects are considered as independent. This means that the state of the agent in relation with each object is $s \in S_{inner} \times S_{obj_i}$

In our game the inner state of the agent is then:

$$S_{inner} = \{Hungry, Thirsty, Ill, Bored, OK\} \tag{5}$$

In relation with static objects the agent can be in the following states:

$$S_{obj} = Have\_it \times Near\_of \times Know\_where \tag{6}$$

In relation with other player:

$$S_{obj} = Near\_of \tag{7}$$

All these variables can be evaluated as $\{yes, no\}$.

3.2 Q-learning Algorithm

As mentioned previously, in MDP environments the agent will use the standard Q-Learning as a learning algorithm. As described in (Gadanho,2002), through this algorithm the agent learns iteratively by trial and error the expected discounted cumulative reinforcement that it will receive after executing an action a in response to a world state s, the Q-values for each object is:

$$\begin{aligned} &Q^{obj_i}(s,a) = \\ &(1-\alpha)\cdot Q^{obj_i}(s,a) + \alpha\cdot\left(r+\gamma \max_{a\in A_{obj_i}}\left(Q^{obj_i}(s',a)\right)\right) \end{aligned} \tag{8}$$

where A_{obj_i} is the set of actions related to the object *i*, *s'* is the new state, *r* is the reinforcement; γ is the discount factor and α is the learning rate parameter.

The optimal policy, chooses the action that maximizes $Q^{obj_i}(s,a)$ this means

$$a^* = \arg\max_a Q^{obj_i}(s,a) \tag{9}$$

3.3 Multiagent reinforcement learning

The difference between single-agent and multiagent system exists in the environment. In multiagent systems other adapting agents make the environment no longer stationary, violating the Markov property. In the learning framework of SGs, learning agents attempt to maximize their expected sum of discounted rewards. Unlike single-agent system, in multiagent systems the joint actions determine the next state and rewards of each agent. After selecting actions, the agents are transitioned to the next state and receive their rewards.

Nash equilibrium is a very important solution concept for the problem of simultaneous finding optimal policies in the presence of other learning agents. Nash equilibrium is a collection of strategies for each of the players such that each player's strategy is a best response to the other player's strategy (Yang and Gu,2004). Littman (1994) proposed a Minimax-Q learning algorithm for zero-sum games in which the player always tries to maximize its expected value in the face of the worst-possible action choice of the opponent. The player's interests in the game are opposite. Later, Littman (2001) proposed the Friend or Foe Q-learning algorithm, for the RL in general-sum SGs. The main idea is that each agent is identified as being either "friend" or "foe".

The Friend class consists of SGs in which, throughout the execution of the algorithm, the Q-values of the players define a game in which there is a globally optimal action profile, i.e. the game has coordination equilibrium. The Foe class is the one in which the Q-values define a game with a saddle point, i.e. the game has an adversarial equilibrium. The Friend-Q updates similarly to regular Q-learning, and Foe-Q updates as does minimax-Q (Shoham et al ,2003).

All these algorithms extend the normal Q-function of state-action pairs $Q^{obj_i}(s,a)$ to a function of states and joint actions of all agents. Taking into account this fact and that each agent can select among n actions while they are interacting, the Q-values to be calculated are $Q^{obj_i}(s,a_1,a_2)$ where a_1 and a_2 belong to the set of *n* actions of each agent.

4 Virtual Scenario

4.1 MUD: Coffeemud

MUD stands for "Multi-User Dungeon", and refers to a text-based multi-user game centred on a fantasy adventure genre such as Dungeons and Dragons. The election of this text-based game instead of using a modern visual one is that we need to simulate a robot with sensors and actuators living in a real world. Therefore, we wanted a virtual world with a very easy way to send and acquire information. Using this text based game, for our virtual robot or agent, acquire information is equivalent to read text and acting (move, take, etc) is to send text. For our purpose, a MUD offered the perfect way to create virtual scenarios (areas), with all the objects (food, water, medicine) to interact with. Among quite a lot of different MUD codebase, we finally decided to choose CoffeeMud, based on Java, due to the well documented software and clear explanations.

In order to make our experiments, we decided to create the area Passage. Passage was designed in a similar way as a department plant, with offices and corridors since the future application is to implement the architecture in our robot, where the scenario will be quite similar.

In this paper we will show the experiments made on the Passage Area, so we will describe it in detail later. In a typical MUD, a person would connect to a MUD Server using a Telnet client, and play. Since we want our agents to play, we have created several programs, in C language, that connect to our mud server, simulating different players. These agents will behave according to our control architecture.

4.2 Agents at the Passage Area

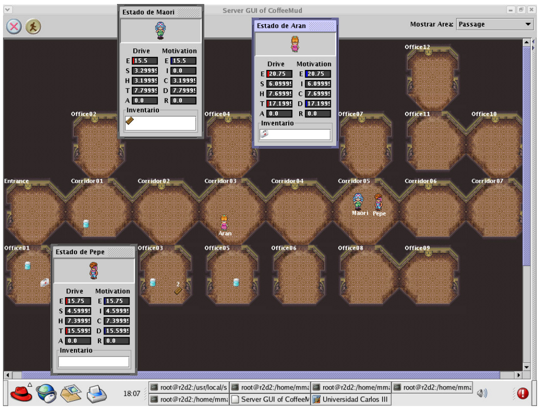
This area is formed by 20 rooms, 8 of them constitute a corridor and the rest of the rooms are offices distributed at both sides of the corridor. In this area the player can find food, water, medicine and other players. Except for the players, the rest of the objects are fixed in different rooms while the players obviously are changing their position over the time game.

Different objects can be found together in the same room, if the agent does not take some object, he remembers the location, and the state of the agent in relation with that object changes.

There are no doors in this area, the way the agent moves in the world is giving "direction" commands: north, south, east and west. With one movement command, the agent passes from one room to another. The commands used for interacting are quite similar too: "get food", "eat food", "take water from Maori".., etc. In the case of the "explore" and "go to" behaviour, each player has implemented a DFS (depth first search) algorithm and a Dijkstra algorithm. The DFS gives a route to explore all the rooms of the area. The Dijkstra algorithm solves the shortest path problem between two known rooms.

4.3 Graphic Interface

Due to the nature of a MUD, as we have said, the interaction between a player and the game is text based. Although it is quite easy to detect all the objects in the area, it is quite difficult to have a global view of the game since you can only "see" the room where you are placed. Furthermore, when there are several players connected at the same time, the only way to watch the player's evolution is using a graphic interface developed by us for this purpose.



Estado de Maori

Drive		Motivation	
E	15.5	E	15.5
S	3.29999	I	0.0
H	3.19999	C	3.19999
T	7.79999	D	7.79999
A	0.0	R	0.0

Inventario

Estado de Aran

Drive		Motivation	
E	20.75	E	20.75
S	6.09999	I	6.09999
H	7.69999	C	7.69999
T	17.1999	D	17.1999
A	0.0	R	0.0

Inventario

Estado de Pepe

Drive		Motivation	
E	15.75	E	15.75
S	4.59999	I	4.59999
H	7.39999	C	7.39999
T	15.5999	D	15.5999
A	0.0	R	0.0

Inventario

Figure 2: The Graphic interface

As we can observe in Figure 2, using this interface we can follow all the player's actions, as well as their drives and motivations values. Moreover, all the items that the player is carrying are showed on the little sub-windows developed for each player.

5 Experimental Results

5.1 Experimental setting

In this paper we present the results obtained by testing six different multiagent RL algorithms on one agent to observe their influence on the final learned policy. Obviously, these algorithms only affect to the way they interact with other players, but in fact we will show the influence on the overall performance of the agent. The performance of the agent was studied by analyzing the wellbeing of the agent which gave us a perfect feeling about how well the performance of the agent was using each of the algorithms.

To test these algorithms we need to recreate for each experiment the same environmental conditions. Therefore six experiments where carried out varying the multiagent RL algorithm implemented on one agent, while the other players had fixed policies.

Those fixed policies are the same in relation with the objects (food, water and medicine), but differ from one player to another in relation with social interaction.

One of them will be the "good" one, therefore when other player appears he will give him the objects that he is carrying as well as interact with him. Another player will be the "bad" one so he will try to take objects from his companion (steal). Finally the other player will be "neutral" and will choose randomly the action to execute with other player.

As it was explained in previous sections, the goal of this kind of games is to survive. The agent needs to eat, to drink water and to take medicine to satisfy its drives: Energy, Thirst and Health. Moreover in order to satisfy the Social drive the agent must interact with other players.

At the initial step of the simulation, all the drives are set to the ideal value 0, increasing their value at each simulating step time. Obviously, at the beginning there will be no dominant motivation till one of the motivation exceed its threshold limit which is set to $L_d = 2$ for all motivations.

When there is no dominant motivation, the agent can perform any of the available actions although apparently, there is no a clear goal. This is consistent with our architecture since the behaviour selection process has been designed to optimize the wellbeing of the agent, not to fulfil a motivational goal.

5.2 Procedure

For these experiments, five multiagent RL algorithms and the normal Q-learning algorithm have been implemented into the agent. The following equations present the way to obtain the $Q^{obj_i}(s, a_1, a_2)$ values for the five multiagent RL algorithms:

$$Q^{obj_i}(s,a_1,a_2) = (1-\alpha)\cdot Q^{obj_i}(s,a_1,a_2) + \alpha\cdot\left(r + \gamma\cdot V^{obj_i}(s')\right) \tag{10}$$

- Friend_Q learning:

$$V^{obj_i}(s') = \max_{a_1\in A_1} \max_{a_2\in A_2}\left(Q^{obj_i}(s',a_1,a_2)\right) \tag{11}$$

- Foe_Q learning:

$$V^{obj_i}(s') = \max_{a_1\in A_1} \min_{a_2\in A_2}\left(Q^{obj_i}(s',a_1,a_2)\right) \tag{12}$$

- Average:

$$V^{obj_i}(s') = \max_{a_1\in A_1}\left(\sum_{a_2\in A_2} Q^{obj_i}(s',a_1,a_2) \Big/ n\right) \tag{13}$$

- Q+Q$^{\mathrm{T}}$:

$$V^{obj_i}(s') = \max_{a_1\in A_1} \max_{a_2\in A_2}\left(\beta_1\cdot Q^{obj_i}(s',a_1,a_2) + \beta_2\cdot Q^{obj_i}(s',a_2,a_1)\right) \tag{14}$$

- Q-Q$^{\mathrm{T}}$:

$$V^{obj_i}(s') = \max_{a_1\in A_1} \max_{a_2\in A_2}\left(\beta_1\cdot Q^{obj_i}(s',a_1,a_2) - \beta_2\cdot Q^{obj_i}(s',a_2,a_1)\right) \tag{15}$$

Where a_1 and a_2 are the actions executed by the agent itself and the opponent respectively. A_1 and A_2 are set of n actions for each agent. r is the reinforcement, γ is the discount factor and α is the learning rate parameter.β_1 and β_2 are limited to [0,1].

The last three algorithms (13)(14)(15) are three possible variations of the Q-learning algorithm extended to the multiagent case. In fact, the Q+Q$^{\mathrm{T}}$ and Q-Q$^{\mathrm{T}}$ algorithms remind to the Friend and Foe ones, respectively, and the Average algorithm is similar to the Q-learning one.

Each of the experiments can be divided in two parts: the learning phase and the steady phase. At the beginning of the learning phase all the Q-values are initialized to zero. In order to choose an action, we set a fixed interval [0, k] to generate a random number, being k a positive number. This random number is added to all the calculated Q-values, and the selected action will be that one with the biggest resulting Q-value. A big value of k will favour the exploration of all the possible actions while a small value of k makes the algorithm to exploit those with the highest Q-values. During the learning phase this number k will be decreased along the game time, to ensure the exploitation of the correct actions. When the agent has learned its policy, then starts the steady phase where the agent "lives" with the learned Q-values.

5.3 Results

In order to analyze the performance of the agent using each of the six RL algorithms, the wellbeing of the agent was recorded.

Moreover, we define four indicators to measure the performance of the agent. The first one is the Average Value of the wellbeing during the learning phase. The second one is the same value during the steady phase. Both indicators give us useful information about the performance of the agent, but they cannot tell us anything about the quality life of the agent during the time life. An average value may be good but it does not consider if it is due to a general good performance or if it was a consequence of the existence a combination of very good moments and very critical ones.

Therefore to make a complete analysis of the performance of the agent, we define two other indicators: The percentage of the total time that the wellbeing was inside the Security Zone (SZ), and the same percentage during the steady phase.

According to the (3), the ideal value of the wellbeing is W_{ideal}, which is set to 100, since this implies that all the drives are satisfied. As mentioned previously, there are activation levels for all motivations $L_d = 2$, based on that we defined the Security Zone for the wellbeing to be SZ = [100,92]. We consider that if the wellbeing is inside this SZ, the drives are not too high and therefore the agent is doing right.

Table 2 summarizes the values of these indicators for each RL algorithm applied on the agent.

Table 2: Results

	Average Wb Steady Phase	Average Wb Learning Phase	Wb<SZ Steady Phase %	Wb<SZ Learning Phase %
Friend	95,15	90,3	94,29%	51,56%
Foe	95,36	92,65	92,48%	66,67%
$Q+Q^T$	94,49	90,25	79,26%	47,78%
$Q-Q^T$	95,16	87,26	88,68%	34,87%
Average	95,59	90,97	93,89%	52,72%
Q-L	94,52	90,3	84,25%	31,3%

These results show that the agent with the Foe-Q Learning algorithm implemented obtained the best performance during the learning and the steady phases. With this algorithm, the highest average values were obtained in both phases as a consequence of a very good performance over the game time. This fact is also reflected in the high percentage of Wb<SZ obtained in both phases.

The results obtained using the other algorithms show that there is no a significant difference between them. We can observe that a good performance is obtained in general, during the steady phase, while the performance during the learning phase is not relevant. These results show that the agent is always learning a correct policy in order to survive in our game with all algorithms

6 Conclusion and Future work

In this paper we have presented a control architecture designed for autonomous and social robots. In the first stage of our project this architecture is implemented on virtual agents. The advantage of using virtual agents instead of real robots is that we do not need to worry about the obvious hardware related problems (actuators, sensors, etc). In order to test this control architecture we designed a simulated world using a text-based role-playing game.

The agent learns the correct policy in order to survive in that simulated world, where he can find food, water, medicine and other players. For this learning process we have implemented several RL algorithms, using the variation of the wellbeing of the agent as the reinforcement.

According to the shown results we have proved that our architecture is able to accomplish the final goal. The agent finally learns the right behaviours to execute depending on the world state.

In the future work, it is expected that the agent learns not only the right policy but also to identify its opponent. So far, the agent treats all its opponents as if they were all the same, and this is not true. In future scenarios, the agent will be able to behave different with the "good" opponent than with the one that tries to steal its objects every time that interacts with it.

On the other hand, one of the main objectives in our project is to include emotions in the behaviour selection process. It has been proved that two highly cognitive actions are dependant not only on rules and laws, but on emotions: Decision making and perception (Picard,1998).

Emotions can act as a control and learning mechanism, driving behaviour and reflecting how the robot is affected by, and adapts to, different factors over time (Fong et al,2002).

In previous works (Malfaz ans Salichs, 2004), an emotional system has been already proposed. In this architecture emotions will be generated from the evaluation of the wellbeing of the robot. Happiness is produced because something good has happened, i.e. an increment of the wellbeing is produced. On the contrary, Sadness is produced because something bad has happened, so its wellbeing decreases. Fear appears when the possibility of something bad is about to happen. In this case, we expect that the wellbeing drops off. Finally, Anger is produced when a decrement of the wellbeing of the robot happened due to an other-initiated act.

Therefore, according to these definitions Happiness will be produced when $\Delta Wb > 0$ and Sadness when $\Delta Wb < 0$. These emotions will be used as positive/negative reward in the reinforcement learning algoritms.

Anger will be produced when sadness arises due to the interaction with another agent. Fear is produced when the agent knows that something bad may happen. This means that the wellbeing of the agent might decrease. To cope with fear the action that produces the negative effect is going to be considered. We will distinguish between actions executed by the agent and exogenous actions carried out by other elements of the environment such as other agents.

Acknowledgements

The authors gratefully acknowledge the funds provided by the Spanish Government through the projects named “Personal Robotic Assistant” (PRA) and “Peer to Peer Robot-Human Interaction” (R2H), of MEC (Ministry of Science and Education).

References

Avila-Garcia, O. and Cañamero, L. A Comparison of Behavior Selection Architectures Using Viability Indicators. *In Proc. InternationalWorkshop Biologically-Inspired Robotics: The Legacy of W. Grey Walter*(WGW'02). 2002.

Avila-Garcia, O. and Cañamero, L. Using Hormonal Feedback to Modulate Action Selection in a Competitive Scenario. *In Proc. 8th Intl.Conference on Simulation of Adaptive Behavior* (SAB'04). 2004

Berridge, Kent C. Motivation concepts in behavioural neuroscience. *Physiology & Behaviour* 81 , 179-209. 2004

Breazeal, C. *Designing Sociable Robots*. The MIT Press. 2002

Cañamero, D. Modeling Motivations and Emotions as a Basis for Intelligent Behavior. In W. Lewis Johnson, ed., *Proceedings of the First International Symposium on Autonomous Agents (Agents'97)*, 148-155. New York, NY: The ACM Press. 1997.

Fong, T., Nourbakhsh. I., Dautenhahn K. *A survey of socially interactive robots: Concepts, design, and applications*. Technical Report CMU-RI-TR-02-29. 2002

Frijda, N. and Swagerman, J. Can computers feel? Theory and design of an emotional model. *Cognition and Emotion.* 1 (3). pp 235-357. 1987

Gadanho, Sandra Clara. Emotional and Cognitive Adaptation in Real Environments. *In: Symposium ACE'2002 of the 16th European Meeting on Cybernetics and Systems Research*, Vienna, Austria. 2002

Gadanho, Sandra Clara. Learning behavior-selection by emotions and cognition in a multi-goal robot task. *The Journal of Machine Learning Research.* Volume 4 Pages: 385 – 412. MIT Press Cambridge, MA, USA. 2003

Littman, M. L. . Markov games as a framework for multiagent learning. *In Proceedings of the Eleventh International Conference on Machine Learning*, San Francisco, California, pp. 157--163. 1994

Littman, M. L. Friend-or-foe Q-learning in general-sum games. *In Proceedings of the Eighteenth International Conference on Machine Learning*, pages 322--328, Williams College, June 2001.

Lorentz, K, Leyhausen P.*Motivation of human and animal behaviour; an ethological view*. New York: Van Nostrand-Reinhold;xix, 423 pp. 1973

Malfaz, M. and Salichs, M.A. A new architecture for autonomous robots based on emotions. *Fifth IFAC Symposium on Intelligent Autonomous Vehicles*. Lisbon. Portugal. Jul, 2004.

Picard, Rosalind W. *Los ordenadores emocionales*. Ed. Ariel S.A. 1998

Shoham, Y., Powers R., and Grenager T. *Multi-agent reinforcement learning: a critical survey*. Technical report, Computer Science Department, Stanford University, Stanford. 2003.

Sutton, Richard S. and Barto Andrew G.. *Reinforcement Learning: An Introduction*. MIT Press, Cambridge, MA, A Bradford Book. 1998

Velásquez, J. When Robots Weep: Emotional Memories and Decision Making. *In: Proceedings of AAAI-98*. 1998

Yang, Erfu and Gu, Dongbing. *Multiagent Reinforcement Learning for Multi-Robot Systems: A Survey* (2004)

Narrative, agency and observational behaviour in a first person shooter environment

Dan Pinchbeck[*] Brett Stevens[*] Darren van Laar[†] Steve Hand[*] Ken Newman[*]

[*]Dept. of Creative Technologies
University of Portsmouth, UK

[†]Dept of Psychology
University of Portsmouth, UK

dan.pinchbeck@port.ac.uk brett.stevens@port.ac.uk darren.van.laar@port.ac.uk
steve.hand@port.ac.uk ken.newman@port.ac.uk

Abstract

A pilot study was carried out which, in part, aimed to identify patterns in observational behaviour amongst players of Half Life 2, a contemporary first person shooter (FPS) game. Participants played for ten minutes in one or two environments and their eye movements were recorded. Subjective observational questionnaires were also filled in. Although the small number of participants (n=8) precludes drawing any firm conclusions, a number of interesting potential patterns are discussed that may have important ramifications for game design and theory, particularly in regard to agency and narrative.

1 Introduction

The relationship between narrative and play is frequently debated amongst game researchers, often in relation to the discussion of narrative's role in knowledge acquisition, memory, perception, consciousness and subjective reality (Ryan, 2001; de Mul, 2005; Pinchbeck, 2006). Although some researchers have concluded that it is a mistake to apply narrative models to the ludic experience (Juul, 2005), an understanding of how different narrative devices and content affect play is nevertheless crucial to the ongoing development of the field.

It has been argued that perception and narrative are fundamentally linked and, in theory at least, manipulations of narrative devices should result in alterations to perception and play. This appears to be a common sense argument that is exploited regularly by game designers. Schank (1990) suggests that both memory and consciousness can be anchored within the process of telling stories, to others and oneself; Sarbin (1986) speaks of the "storied nature of human conduct" and Dennett (1991) conceptualises consciousness as the illusion of a linear, causal output of a massively parallel system. All of these models find resonance with Tulving's conceptual distinction between episodic and semantic memory (1972). In particular, Schank and Sarbin consider autobiographical memory as narrative in nature. If one accepts the basic premise of schema as behavioural management systems (Bartlett, 1932), then a relationship between perceptual behaviour and the schemas that translate information into knowledge seems to follow naturally. However, whether such a relationship is reciprocal and whether an inherently narrative memory means observational behaviour is somehow attuned to narrative devices and structures in the environment is another matter altogether. Simply because we remember narratives well, it does not follow that either narrative information is privileged in perception, or that narrative has any impact whatsoever on the act of perception.

A pilot study investigating various aspects of game play in First Person Shooter (FPS) environments was recently carried out. In particular, the study was an initial investigation into the relationship between presence, narrative and player behaviour in such environments. FPS games were selected as a form of mass market virtual environment, where the avatar is minimally intrusive and thus a greater projection of the player into the game world is potentially enabled. The relationship between the narrative of the game and the experience of the player is likewise more direct than third person perspective game experiences, as the avatar is less explicitly implicated or impacted by game events. In other words, it is less a case of controlling an avatar than undertaking actions oneself. Under such circumstances, it is reasonable to expect a less abstract engagement with the game than would be expected with the filter of a third person, visible avatar.

Although FPS games have historically tended to forgo much in the way of narrative, using only a very superficial framework to initiate action; more recent games have explored the value of more complex, even intrusive, narrative. Deus Ex (2000) effectively combined FPS action with a highly successful branching narrative structure, actively encouraging ethical engagement in the player over mindless violence, and is still frequently cited as one of the pinnacle achievements of the genre. The sequel, Invisible War, is a failed attempt to replicate this success, suffering primarily from both an overly obtuse and intrusive narrative which results in constant interruptions to play and over-frequent environment loading screens. Both Half Life (1998), and Half Life 2 (2004) reduce the active disruption of play by excluding narrative cut-scenes entirely, but nevertheless rely on strong narrative to drive play forwards and have been applauded for their narrative strength and focus. Half Life’s developers, Valve, have also hinted strongly that the follow-up to Half Life 2 will involve a greater emphasis on interaction with non-player characters, something which suggests an ongoing focus on narrative importance. Regardless of the ongoing popularity of minimally narrative, single-player FPS games such as Black (2006), Doom 3 (2005) and Quake 4 (2005), narrative is certainly not a discounted aspect of successful FPS design.

Eye tracking was initially put forward as an additional, objective, real-time measure that could be used to cross-validate subjective measures, in the form of post-test presence questionnaires. However, the data yielded some intriguing suggestions of patterns that are worth examining in more detail. In particular, three sets of patterns stand out:

1. Differences in perceptual behaviour between experienced players and novices;

2. Relative importance of architectural and agent-based objects in the environment;

3. Shifts in perceptual behaviour in action- and exploration-based sequences.

These patterns will be discussed in more detail, and some inferences regarding the relationship between play, narrative and agency will be put forwards. It will also be discussed whether the results obtained suggest whether it is indeed possible to predict and manipulate player behaviour on a long and short term basis by making alterations to the distribution of narrative elements within a temporal and environmental span. Such manipulation and prediction would, it is suggested, enable more engaging and rich experiences in game worlds to take place.

2 Embedded Devices

Jenkins (2003) postulates four types of narrative inherent to game experience: embedded, enacted, evoked and emergent. The terms correspond to:

1. Devices distributed throughout the play environment that convey narrative information;

2. The narrative act formed by experiencing the environment as a whole;

3. Pre-existing narratives from the player's previous experiences, which are not limited to game play and that are triggered by events or objects during play;

4. The narrative that is created as a result of play.

It is the relationship between the first, second and fourth that this study will focus on, a relationship which has elsewhere been identified as the most obvious nexus between narratology and ludology (Brand and Knight, 2005). In particular, several pertinent questions are raised by the initial study: can a coherent, stable narrative emerge from play, without play itself being conducted in a consciousness narrative manner? In other words, does the player play narratively, or just become aware of the narrative after the experience? Secondly, how holistically does the player engage with the narrative embedded in the environment, or do they privilege certain types of device and delivery? Is there a correlation between the perceived post-experience importance of narrative and behaviour during play; are players who have strong recollections or empathic responses to in-game narrative distinguishable from those with little narrative retention or empathy during the act of play? This may be re-phrased slightly; does enacted narrative really exist to the player, as opposed to the observer?

Narrative must be distinguished from story in order to enable a discussion of the comparative influence of short-term goal achievements on longer term, plot-based goal achievements upon player behaviour. Narrative is defined here as a linear sequence of actions or events, linked in a causal manner, which yields meaning. Two points need further clarification: firstly, that this causality may be projected or implied, rather than inherent in the actions or events themselves; secondly, that narrative is to be seen as a type of schema, an information management device. The definition therefore draws upon the distinction first made by the Russian Formalist school, between fabula and sjuzet, or the events and the telling. This has been alternately described as a relationship between story and discourse. However, we suggest that the conceptual baggage of story is both inappropriate and misleading when considering narrative within an environmental context. The story and discourse distinction implies that linearity or causality is inherent in the former, that it carries its own internal logic. Game and virtual environments, however, are comprised of a network of related elements, with not just potential for the emergence of linear causality, but usually a deliberate predilection for its emergence. However, the sequence remains only a probable emergent factor in the experiencing of the environment, not a given. Story, according to the former model, always contains the sequence; distinct discourses may be applied but will not fundamentally affect it. Game environments frequently do not share this functional nature. Given this, we can either attempt to describe games such as Deus Ex as a large number of identifiable branching narratives interwoven with an overall ludic architecture, or seek an alternative conceptualisation of narrative, where story is embedded as a potential within an environment. Rather than a story being something which, when combined with discourse, creates a narrative, we define a narrative as a particular type of structure occurring when events are linked causally into a sequence. Thus, narrative precedes story, which we see as a stable set of narratives capable of repeat, independent enactment. By defining narratives as structures that may exist both as nested and independent elements within a story, therefore, narrative structures can be used to examine both long and short-term goals within an environment. Within an overall story none, some or all narratives may be engaged with, yielding a subjective outcome, which nevertheless falls within a constrained framework. It is the stability of this outcome, how closely one experiences resembles another, which would defines it as a story or not. A story is stable and exists as an abstract media object; a narrative by contrast is unstable and subject to reformation or reinterpretation each time the events are called into sequence.

Thus, in the case of Half Life 2, the overall story is the player's ongoing battle to free a city from an occupying alien force. This can also be seen as a long-term goal, but it is impossible to conceptualise without engagement with the narrative material contained within the story. Within this are episodic structures, each with a distinct narrative (i.e. linear sequences that are mediated by subjective experience) and long-term goal, such as escaping hostile pursuit, or rescuing a scientist. The narrative elements of these episodes may be more or less crucial to their successful completion. For example, the episode “We Don't Go to Ravensholme” can be completed without the slightest attention being paid to the narrative goal of joining up with rebel forces elsewhere in the city. Each episode can then be broken down into short-term goal sequences, defending a computer terminal or crossing a bridge, which reduce the need for narrative further. Finally, there are micro-goals, too short to contain any real narrative function, such as shooting at an enemy or picking up a box. As Jenkins suggests, the carrying out of actions within the game also yields a subjective, emergent narrative to the player: the story of their game. It is therefore entirely possible that an emergent narrative may be formed in retrospect, following the accumulation and combination of non-narrative actions and micro-goals rather than through explicitly engaging with any of the narrative structures and devices.

Nevertheless, narratives are powerful schema. It could be argued that, as such, they share common features and potentially overlap those schema underpinning a great deal of human knowledge acquisition and even perception (Neisser, 1976). As a means of codifying information and imposing causality, their ability to evoke pre-determined, controlled sets of responses to events underlies most media artefacts. It should be expected that strong narrative will influence subjective experience in games, and evidence suggests that this is just the case (Schneider et al, 2004). It is both the relationship between goal and narrative and the relative importance of differing types of embedded narrative devices that are under scrutiny here.

We propose that embedded devices can be split into four categories: non-narrative, passive, active and dynamic. Each of these categories may be further defined by the temporal scale it predominantly affects. For example, Fig 1 contains a huge number of passive conveyers of narrative: the architecture positions the overall game world in time and space; the trees suggest a season; the huge alien citadel in the background (1) sets up both a long term narrative intrigue and sows the seeds of a long term goal.

Fig 1: Screenshot of “Point Insertion”, HL2

The public broadcast screen in the mid-ground is a good example of an active narrative device (2). Narrative information is actively supplied by this device; the player can, by listening to the broadcast, gain additional understanding of the situation. Whilst the device does not contribute anything to immediate goals, or short-term narrative, it actively establishes the game world further.

In the foreground, a humanoid agent (3) will dynamically respond to the player, usually aggressively. Whilst contributing towards long-term narrative as a collective, or type, the individual agent's role in FPS games is ordinarily short term and micro-goal orientated, such as fulfilling a combat function. Friendly Non-Player Characters (NPCs), such as Half Life 2's Alyx, operate dynamically and actively contribute to long and short-term narrative.

Finally, non-narrative items include barrels, crates and cans. These items can be manipulated and used to achieve goals, but really only exist as tools in the environment and do not carry any particular narrative significance in themselves.

A pertinent question when considering player behaviour and game design, therefore, is the relative impact of these diverse elements upon play. For example, do the narrative devices come together with evoked, pre-existing narrative information to form a coherent whole? Or, is Jenkin's emergent narrative defined only by a linear chain of events without any internal reference to its component narrative relationships? How much are players conscious of narrative devices during play, if at all? Or are they processed less consciously, with the overall game narrative functioning more as a memory structuring system, anchoring goals and actions together in a coherent and accessible way?

3 Description of Study

Eight participants were recruited to take part in the pilot study. Of these, three were novices, with little or no game experience and no FPS game experience. The other five all described themselves as regular game players with FPS experience. Two of the five had played Half Life 2 previously.

The game engine was run on an RM PC with a 3.20GHz processor and 2GB of RAM. Participants used keyboard and mouse controls, which they configured to their preferred settings to interact with the game. The display was projected onto a screen approximately 6' from the player, with the overall display dimensions being 6' x 3'. Eye movements were recorded by an ASL 501 head mounted Eye Tracking System. The data is integrated in real time with the display, resulting in a positional graphic overlaid upon the environment, outputted to VHS. In addition to the eye tracking data, a short observational questionnaire was filled in, post-test, by all participants. All eight participants played the opening sequence of the game, “Point Insertion”'. This is a non-combat, exploratory sequence with one embedded ninety second narrative scene. Four of the eight participants, of whom all were experienced gamers, also played an additional, combat orientated sequence, “Route Kanal”. Each sequence was played for ten minutes.

4 Observations

4.1 Novices and Experts

A number of key differences to observational and exploratory behaviour were immediately apparent between novices and experienced players. The most immediate and expected was that the novices looked off-screen to the controls more frequently or positioned their point of focus (hereafter referred to as the focal point) at the bottom of the screen to keep the controls in their peripheral vision. Interestingly, however, two of the three novices stated in their post-test questionnaires that they felt they had not looked offscreen. This would suggest a moderate sense of presence was being generated, according to parameters to be found in the literature (Youngblut, 2003), and was corroborated by post-test subjective response. In the case of the third novice, the test was stopped after seven minutes due to simulator sickness. However, it was apparent that the participant, even though reporting that they were conscious of looking outside the screen often, felt a degree of spatial presence, as evidenced by whole head movements to attempt to gain better perspectives on the environment, and moderate scores in post-test presence questionnaires.

Fig 2: Novices tend to track objects with focal point whilst avatar perspective remains fixed.

The second key difference was in the relationship between eye movements and avatar perspective. The novices' eye movements were jerkier and less tied to the avatar perspective, with a tendency to fix the latter and then use the focal point to explore the field of view in more detail (Fig 2).

Fig 3: Experienced players tend to centralise focal point and use avatar perspective to track objects.

By contrast, experienced gamers restricted their gaze to predominantly the central section of the display, using avatar perspective to explore their surroundings, i.e., moving the window around the virtual world. When coupled with their prioritising of dynamic elements, this seems to suggest that a specific set of observational schema for the medium are indeed in operation. Experts keep their focus close to the aiming mechanism (Fig 3).

Likewise, there was a clear structure to observation: the focal point saccaded from object to object and the avatar perspective followed if the object attracts attention. The lag between focus and shift of avatar perspective was visibly reduced in all the experienced gamers compared to the novices. In a full study, it will be interesting to compare this lag with fixation and inter-fixation lengths and durations, something the small number of participants in the pilot study disenables.

None of these observations are particularly surprising. It is also important to note that manual, interface-ability criteria cannot be discounted from any discussion of how a player reads an environment. The suggestion that experienced gamers operate according to learned schema must be tempered with the understanding that it may be due in part to their simply being more adept at using the interface. Interestingly, as novices became more confident with the controls, it appeared that their gaze was beginning to centre. Whilst this is a largely unsupported observation, it is nevertheless an indication that it would be interesting to conduct a full study, allowing novices a greater time in an environment and seeing how long it takes for their more freeform explorations to settle into the more recognisable patterns of experienced gamers.

4.2 Hierarchies and Schema

There appears to be a clear hierarchy of embedded devices, as might be expected. NPCs and humanoid agents receive a disproportionate share of visual attention and were generally the first object within any area that was attended to. For some of the subjects, this seemed to be a deliberate tactic; they used agents to navigate through the environment, literally hopping from one to the next as they made their way through. For most of the participants, however, this attention appeared automatic. In all participants, there was a characteristic bobbing motion: when the focal point moved close to an agent, attention quickly shifted to that agent and back again. This behaviour appeared to be lacking in combat situations and may therefore be part of a threat assessment pattern. This idea is lent weight by the tendency of players to lock onto hostile targets and deal with them one at a time; a kind of tunnel vision. In both cases, examining the X and Y co-ordinates of fixations and determining whether this bobbing motion can be tracked through fixation and inter-fixation co-ordinates and durations requires full statistical analysis, something the sample size precludes.

Non-human but moving objects seemed secondary in terms of attached importance. The flying Scanners found in both sections of the game were attended to, but generally after human agents had been assessed, or in their absence. The giant television screens found in three separate locations in "Point Insertion" received comparatively little attention from most participants, and only one participant spent any more than a second or two observing them. It was noted that perhaps the information transmitted by the devices is primarily auditory, but they do remain a striking and active visual features of the environment. Security cameras were likewise observed only sporadically - all participants seemed to rely predominantly upon agents to push the action along and steer them forwards in the environment. There was a marked lack of attention from four of the eight participants to passive devices. This included the Half Life logos sprayed onto walls, which are designed to steer the player's attention to important parts of the environments, such as supply crates and hidden entrances. This could be argued to be something that players need to learn within the game, but it is interesting nevertheless that a highly visible marker remained so frequently unnoticed by participants.

Fig 4.: The post-cut scene environment with ladder.

Perhaps the most evident example of this hierarchy of importance was found following the narrative 'cut-scene' a few minutes into "Point Insertion". Barney, a key NPC is introduced, who establishes the mid-term goal of the section: escape the station and find a secret laboratory. The scene ends with the player being ushered through a door with the verbal instruction from Barney to "pile up some stuff and get through that window". The room beyond is split level, with a ladder reaching the upper level where the window is located. There are a number of moveable objects on both levels. With one exception, all participants immediately began stacking objects. No attention was paid to the ladder, even though it was in clear view (Fig 4). It appeared that direction from an agent, establishing a micro-goal was prioritised over the exploration of the immediate environment.

The evidence of hierarchical attention, especially in the experienced gamers, is more suggestive of an FPS schema than the focal point and avatar perspective relationship. The taxonomy of narrative devices appears to function as expected: those objects which enable the greatest projection of agency are prioritised in perception. Background narrative, however, is relatively minimal in importance. During "Route Kanal", a looped announcer's voice is clearly and regularly audible, but participants struggled to remember if the voice was male or female, and one did not remember a voice at all. Likewise, the lack of attention paid to the television screens by most participants meant that a substantial amount of background material was not picked up on. Posters and signs were barely looked at; players were far more eager to approach non-player characters, such as the civilians in the station. This focus on those objects within the environment that enabled greatest projection of agency may even be interpreted as the player seeking out the most likely source of narrative information.

Another example of lack of attention to passive narrative devices occurred on leaving the station, into a large open city square. Behind the immediate buildings, in clear view, framed by the architecture and directly ahead of the player when they exit the station is a large alien structure that extends right into the clouds, way above the surrounding city (see Fig 1). Of the four who exited the station, only one participant paid attention to it. Given the effort Valve have recently expended into the extraordinary graphical enhancements of the Half Life 2 demo, Lost Coast, the lack of attention paid to non-dynamic objects is somewhat striking.

4.3 Exploration and Action

There appeared to be noticeable differences in perceptual behaviour in the combat orientated "Route Kanal" sequence in contrast to the exploratory "Point Insertion". Again, these differences ranged from the expected to the less immediately apparent. The point of visual focus in the former tended to fluctuate more between being fairly static and central over the avatar perspective cross-hairs and rapid, staccato movements throughout the display area.

On entering each new area, participants could clearly be seen to be carrying out an assessment of the environment. This usually involved relatively static avatar perspective and highly mobile focal point. Any object that stood out from the background: boxes, crates and barrels were included in this sweep. Once the assessment was carried out, gaze movement tended to centralise and avatar perspective was used to visually navigate. Tellingly, these objects, characterised by being non-narrative but potentially useful for the completion of micro-goals, received very little attention in the exploratory sequence, with the exception of the box stacking sequence described above, which required specific player attention.

All participants tended to alternate between two distinct eye movements when faced with hostile dynamic objects, i.e. combat. Rapid visual searching, often with a relatively static avatar perspective, would be replaced with a less mobile focal point, with visual control determined predominantly by avatar perspective, once a target had been identified. It seems fair to assume this is due to the functional nature of the aiming mechanism. All the participants, for the vast majority of the time, locked onto one target at a time and dealt with threats in a linear manner. The alternative behaviour usually occurred when hostile targets arrived unseen and the player was alerted to their presence by audio and visual cue, such as the sound of gunshots or bullet holes appearing in the walls around them. It would be interesting to correlate pupil diameter and heart rate to these moments, as it was at these points that the relationship between gaze and avatar perspective seemed to break down, and both would move rapidly and independently until the source of the threat was identified.

Finally, some attention was paid to whether the moving targets in the combat sequences were observed in any detail, whether any semantic information about them was picked up. All participants correctly identified simple details in the agents' appearance, such as colour of uniform and type of headgear. Whether participants would correctly identify individual traits or more complex visual information may be worth examining in more detail, as a means of further investigating the relationship between the relative importance of object characteristics and perceived agency. Participants were also asked how many agents they had fought during the ten minute sequence, with varying degrees of accuracy. One participant estimated only half the actual number. It would be interesting to see how accurate players are at counting agents or other devices within a space, which may help ascertain the role of peripheral perception in play, as well as a corroborative measure of presence.

5 Goal, Agency and Narrative

It is possible to cautiously infer some patterns from the limited results detailed above. In particular, the question of how, and indeed whether, players respond to narrative devices can be addressed. It would appear that although players perceive more than just moving objects, and both subconsciously and specifically use agents to structure their perceptual exploration, the narrative aspects of the game are, at the least, pushed to the background. The suggestion that media- and genre-specific schema are formed and used as players gain experience and confidence supports this idea. At one level, the agent-to-agent navigation demonstrated by experienced players may be explained as constant threat assessment, but this does not explain why objects in exploratory sequences are not treated to similar surveying tactics as the ones observed in action sequences. The fact that non-narrative devices are the focus of visual attention more than any passive narrative devices in such situations seems to suggest that achieving micro-goals far outweighs any importance attached to even short term narrative. How an object may be dangerous or useful seems to overshadow its meaning. Dynamic objects thus require closer observation. There also seems to be a prioritisation of dynamic objects that support the intentional stance, and easily lend themselves to a projection of agency; in other words, humanoid avatars. Although FPS games are replete with non-human and often non-humanoid dynamic objects, the flying scanners in Half Life 2, in both sequences, received less attention. In the combat sequence, this is easily explained, as they represent far less of a threat, but at least in the early stages of the exploratory sequence, they intrude upon the player to a far greater degree than the human avatars, who require provocation to actually act. It could perhaps be argued that human avatars are looked to as objects expected to deliver a high degree of narrative, but this would seem to initiate a greater degree of active interaction than was carried out.

Whether the hierarchy is based upon possible danger (i.e. a mechanical, ludic assessment of the environment), or a natural tendency to pay closer attention to human like forms may be tested further by placing players in situations where human avatars are dynamic but clearly unthreatening, but non-human avatars represent the greatest threat.

Regardless of this, what is suggested strongly by the results is that mid- and long-term narrative seem to matter little to players during the act of play. The lack of awareness of passive narrative objects was striking; especially in a game that is generally acknowledged to contain a strong narrative. Rather than steering play, the results may indicate that narrative has little to do with it, and is imposed post-experience. In other words, when we speak of narrative influencing play, it essentially translates to the narrative being used to position action in context for memory, but not actually being part of the play experience. There is no game narrative in the present tense, no enacted narrative in Jenkin's terms; it only exists prior to and after play.

If play is structured more by projection of, and interaction with, in-game agency, in pursuit of micro-goals that are more or less invisibly underpinned by narrative structures, this has implications for the relative importance attached to environment design and artificial intelligence within game worlds. If players consistently prioritise agents as sources of visual focus and information, the complexity and depth of passive or background devices may be to some extent superfluous. This would certainly find support in the argument that gameplay will always win over graphics, though it may well be that visual depth and complexity matter a great deal, but only in the periphery. Schneider et al (2004) found that highly narrative games (Half Life and Outlaws) evoked greater empathy and presence than non-narrative games (Doom 3 and Quake 3). However, skin conductance was the only objective measure used and, furthermore, the only measure taken during the game experience. Other data was subjective and post-test. Schneider's results should then be no surprise, but do not necessarily mean that highly narrative games are actually engaged with on a narrative level during play. Alternatively, the strong narrative could make it easier to recall specific moments within play by acting as a more robust framework for retrieval, thus yielding more specific and potentially more vivid recollections of the experience. Additionally, each game was played for only eight minutes, a short space of time for any sense of long-term narrative to develop and certainly short enough for Doom 3's limited narrative to remain unexhausted. Indeed, in a straight comparison between Doom 3 and Half Life 2's opening sequences, it could be argued there is little difference in terms of narrative strength. The former is certainly more clichéd in both its science fiction setting and testosterone levels but actually invests more depth into most of the encountered avatars. It may well be that it is the very strength of the narrative, in terms of genre conventions and cult appeal, that reduced subjects' empathy with the game.

6 Conclusion

The data from eight participants is not enough to draw firm conclusions, but the questions raised by this feasibility study confirm that investigating a player's perceptual behaviour is a valuable exercise. The role of narrative devices and structures during real time play appears to operate at least in the background, and may even be more or less negligible. Far more focus is given to agents, with dynamic humanoid avatars receiving most attention, regardless of their narrative function, or lack of it.

Novice players appear to read the game environment in a different way to experienced players. This is probably due to unfamiliarity with the interface, but the results are also highly suggestive of media- and genre-specific schema developing to structure perception and interaction. Exploration and action sequences appear to trigger different modes of perceptual behaviour, with the latter actually appearing to yield more investigation of the surrounding environment, particularly non-narrative objects.

Finally, it is recognised that even though narrative may not play such a large part in the action of play as may be assumed, and may not affect player experience in real-time, it may nevertheless serve three very important functions. Firstly, it allows the information and affordances embedded in the environment to be structured in a meaningful way during the design process. Secondly, background narrative schema that pre-exist play, but may be the result of prior game experience, allows the projection of causality and, crucially, agency, that enable successful ludic experiences. Thirdly, strong game narratives may assist the structuring and management of memories of the play experience, by supporting actions with a robust, temporal and contextual framework. Thus, games with strong narratives may feel like they are more immersive, emotionally involving experiences whilst the in-game experience, in fact, remains indistinguishable from games with weaker narratives. It is also worth bearing in mind the crucial distinction between weak and strong narratives, and complex and simple narratives. A weak narrative, such as Doom 3’s may be delivered and maintained highly effectively, whilst a more complex narrative, such as Deus Ex: Invisible War is far weaker in comparison.

A full study will now be carried out investigating two of the three observations detailed here. The relationship between different narrative devices will be further explored, investigating whether different perceptual schema mean more of less narrative information is picked up from the environment. It would also be interesting and valuable to survey observational behaviour amongst a range of FPS games with varying degrees of narrative complexity and strength to see if more narrative games do, in fact, provoke, maintain and reward alternative perceptual schema.

References

Frederic Bartlett. Remembering: A study in experimental and social psychology. CambridgeUniversity Press, 1932.

Jeffrey E. Brand and Scott J. Knight. The Narrative and Ludic Nexus. in Computer Games: Diverse Worlds II. *Proceedings of DiGRA 2005 Conference: Changing Views – Worlds in Play*. 2005

Daniel C. Dennett. Consciousness Explained. Penguin, 1991.

Jos de Mul. The Game of Life: Narrative and Ludic Identity Formation in Computer Games. *In: Joost Raessens, Jeffrey Goldstein (eds). Handbook of Computer Game Studies*, MIT Press:251-266, 2005.

Id Software (developer). Doom 3. Activision (publisher). 2004

Ion Storm (developer). Deus Ex. Eidos Interactive (publisher). 2000

Henry Jenkins. Game Design as Narrative Architecture. *In: Noah Wardrip-Fruin, Pat Harrigan (eds). First Person: New Media as Story, Performance and Game*. MIT Press:118-130. 2003.

Jesper Juul. Games Telling Stories? *In: Joost Raessens, Jeffrey Goldstein (eds). Handbook of Computer Game Studies*, MIT Press:219-226, 2005.

Ulric Neisser. Cognition and Reality: Principles and Implications of Cognitive Psychology, W.H.Freeman and Company, 1976

Dan Pinchbeck. Ritual Co-location: Play, Consciousness and Reality in Artificial Environments. *In Proceedings of Connectivity: The Tenth Biennial Symposium on Arts and Technology*, 2006 (in press)

Marie-Laure Ryan. Narrative as Virtual Reality: Immersion and Interactivity in Literature and Electronic Media. Parallax. 2001.

Theodore R. Sarbin (ed.). Narrative Psychology: The Storied Nature of Human Conduct. Praeger, 1986.

Roger C. Schank. Tell Me a Story: A new look at real and artificial memory. Charles Scribner's Sons, 1990.

Edward Schneider, Annie Lang, Mija Shin, Samuel D. Bradley. Death with a Story: How story impacts emotional, motivational and physiological responses to first person shooter video games. *Human Computer Research,* 30(3):361-375, 2004.

Endel Tulving. Episodic and semantic memory. *In Tulving, Endel and Donaldson, Wayne (eds.), Organisation of Memory*. Academic Press, 1972.

Valve (developer). Half Life 2. Electronic Arts (publisher). 2004

Christine Youngblut. Experience of Presence in Virtual Environments. *IDA Document D-2960[Log H 03001849], 71 + 71 (Appendix)*. 2003.

Requirements of non-verbal communication in believable synthetic agents

John Shearer
Informatics Research Institute
University of Newcastle upon Tyne
Newcastle upon Tyne
NE1 7RU
United Kingdom
`John.Shearer@ncl.ac.uk`

Patrick Olivier
Informatics Research Institute
University of Newcastle upon Tyne
Newcastle upon Tyne
NE1 7RU
United Kingdom
`P.L.Olivier@ncl.ac.uk`

Philip Heslop
Informatics Research Institute
University of Newcastle upon Tyne
Newcastle upon Tyne
NE1 7RU
United Kingdom
`Philip.Heslop@ncl.ac.uk`

Marco De Boni
Unilever Corporate Research
Unilever R&D Colworth
Sharnbrook, Bedford
MK44 1LQ
United Kingdom
`Marco.De-Boni@unilever.com`

Abstract

Realism for synthetic characters, both in computer games and conversational agent mediated applications, requires both visual and behavioural fidelity. One significant area of synthetic character behaviour, that has to date received little attention, is non-verbal communication. In identifying the scope and participants of non-verbal communication in computer games we first review the range of spatial and task scenarios that are relevant. We then select four principal categories of non-verbal behaviour: gesture; gaze; proxemics; self-adaptors; and both summarise their role in communication and propose their incorporation in the design of non-player characters. In conclusion we review the question of how to capture the non-verbal behaviour of players or provide them with interaction techniques that might facilitate non-verbal communication with players and non-player characters alike.

1 Introduction

Perhaps the most significant current challenge for synthetic agents in games is the integration of narrative and *intelligent* character behaviour with a view to further enhancing user engagement. The development of non-player characters has primarily focussed on high level coordination of non-player character behaviours (typically through the characterisation of state space) and little attention has been paid to low level, non-physical, interactions between characters (both non-player characters and player characters) such as different forms of verbal and non-verbal communication.

Realistic interaction between characters is a challenging problem. In simple terms it requires non-player characters to have sophistication in their beliefs, desires, and intentions equivalent to the expectations that a player might have of a real equivalent character (within the narrow domain of the game). Although, one should keep in mind the possibility of an ‘uncanny valley’ (Mori, 1970) – that as characters become more humanlike, with a positive emotional response from a human, there becomes a point where the response suddenly becomes strongly negative – the character is almost human, but is a very strange human – something is perceived as being not quite right. This is clearly contrary to the desired effect.

Crucially, non-player characters must also have beliefs as to the beliefs, desire and intentions of other characters, and therefore require some ability to monitor the activity of other characters for intentional behaviour. This then allows for communication (the intentional influence of other’s beliefs).

Of course, this requires a non-player character to distinguish between behaviour that is, and is not, intentionally influencing another’s beliefs, as many behaviours are interpreted differently depending on the context. For example, opening a door can be an action (enabling continuation of a route) and communication (showing another person respect). Humans (and many higher mammals) are extremely good at separating basic actions from actions intended to communicate (as a conscious act or otherwise), and, in the course of social interactions, people constantly analyse each other's behaviour (e.g. speaking, moving, gesturing, touching) to understand their (the other's) beliefs, desires and intentions. It has been shown that people also do the same for artificial entities (Reeves and Nass, 1996). That is, people have a tendency to attribute intention to all things they interact with, even if those objects have no possible mechanism from which intention might derive (according to our present scientific understanding of the world), for example, by shouting at a household object if it is not working correctly. Inevitably, players ascribe intention (often very complex intention) to non-player characters even when, their behaviour, and/or the underlying control architecture, is very limited.

Non-player characters in present-day games often incorporate a restricted notion of desire and intention, but have no significant ability to perceive the desires and intentions of other characters (and thereby infer their beliefs and other aspects of internal state). Thus, intention is usually exhibited through direct action – in a first person shooter non-player characters display their intention to frag (kill) a character simply by shooting at that character. Indeed, in games where the role of non-player characters is not so clearly defined, players quickly habituate to repetitive behaviour as this "supporting cast" mills around according to simple rules or scripts.

A discussion of the development of interactions in all areas of games is beyond the scope of our discussion. Instead, we address communication between characters, specifically non-verbal communication through gesture, facial expression, eye gaze and other bodily movement. That is, we address non-verbal interactions between characters that are not direct actions (i.e. not actions such as attacking). Most games portray the intentionality and personality of characters with cut-scenes – scenes inserted, pausing game-play, at various points of the game – with fully animated characters using non-verbal and verbal behaviour in an attempt to provide an engaging narrative for a game. In contrast, during actual game-play a character's behaviour is much simpler, and very little intentionality is usually displayed.

The advent of high performance 3D graphics hardware and software, and high quality audio has given rise to the use of geometrically detailed, skeletally animated characters, that exhibit some non-verbal behaviour. For example, Half-Life 2 (Valve, 2004) makes extensive use of non-verbal behaviour in non-player characters, to the extent that player engagement and narrative development can occur effectively without the use of cut-scenes. Non-player characters attempt to talk with the player; if engaged in talking to a player character they will: move around to maintain eye contact during the conversation; respond to the players actions; and show expressions and gestures appropriate to the game-play, thus attempting to engage the player with the narrative.

While the characters within Half-Life 2 appear to have desires, personality and intentions, their actual behaviour is, in fact, highly scripted. Most non-player characters have only a relatively small set of simple behaviours and a number of complex scripts that only occur once or twice. These complex behaviours replace the traditional cut-scene, but without the need to break game-play. However, the integrity of the non-verbal behaviour relies on the skill of the artists in coordinating the non-player character's action with the game-play and the non-player character's speech. This can clearly be seen when observing non-player characters' interactions with each other. Aside from pre-scripted cut scenes, non-player characters do not interact with each other, other than to avoid collision. Even then, they still walk on paths that lead to collision, but evade each other at the last moment (sometimes accompanied with a sound effect along the lines of 'watch where you are going'). Indeed, following an individual non-player character through the environment reveals that they are predominantly on pre-fixed paths that simply enter the environment at one point and walk off at another. Some of the inhabitants even walk in a constant looping path around the environment.

Similar behaviour is exhibited in other games, such as Grand Theft Auto 3 (Rockstar Games, 2001). Although more densely populated, the non-player characters exhibit similarly aimless behaviour, reacting only to certain events (such as explosions or being attacked) but otherwise wandering through the environment with no apparent destination.

There are games in which the behaviour of non-player characters is more sophisticated, such as 'The Sims' (Maxis, 2000), where the characters fall half-way between being user controlled and AI controlled. The characters converse with each other and utilise objects in their environment. Though, even in this case the characters exhibit only the desire to satiate certain undesirable internal states, such as hunger and tiredness, by using the objects and characters around them. They have no longer term and more complex goals, and the conversations between characters do not actually convey any specific information but rather the act of conversation itself satisfies some current short-term need; the characters can obtain the same effect by using an object in the environment, such as reading a book or listening to the radio.

In practice the scope of application of non-verbal behaviour extends far beyond the replacement of the cut-scene and in the next section we consider the applicability to games of the relevant spatial and task contexts over which non-verbal communication is known to be utilised in humans.

2 Spatial and task context

Communication can be considered to occur in four different task contexts: cooperation; coaction; competition; conversation (Knapp and Daly, 2002, p177). In other words, communication occurs in order for some number of parties to: perform a task together (cooperation), to exist in the same vicinity (coaction), to perform a task at the expense of another (competition), and to entertain or pass on information (conversation). Communication varies across these different contexts and also with the physical proximity of the communicating parties. As already described, non-verbal communication provides information as to the beliefs, desires, and intentions of a character, or alternatively it can be considered as providing indicators as to another character's: cognitive state; emotional state; physical state; intentional state; attentional state; perceptual state; interactional state; social state.

Figure 1a: Cooperation in intimate space.

Figure 1b: Coaction in social space.

Figure 1c: Conversation in personal space.

Figure 1d: Competition in personal space.

Figure 1e: Cooperation at a distance.

Figure 1f: Conversation in social space.

Figure 1: Examples from Half-Life 2 of spatial and task contexts for non-verbal communication

A set of distinct but common communication situations (spatial and task contexts) for computer games is illustrated in Figure 1 using screenshots from Half-Life 2. Figure 2 maps out the range of spatial and task context for these examples. Synthesizing non-verbal communicative behaviour for conversational partners at close physical proximity is particularly difficult, due to the full movement of a character (both body and face) being visible in detail to the player. Furthermore, people are highly attuned to interactions in intimate, personal, and social spaces and are sensitive to many subtle cues and nuances in non-verbal behaviour. Thus, at close proximity players are very aware of errors, un-realistic, or unnatural behaviour in non-player character. At further distances less detail of a character's movement is apparent. Moreover, there is a significant transition in non-verbal communication from situations where intimate verbal communication is possible to those where it is not. The sensitivity of non-verbal behaviour to proximity is due to a number of factors, including the more public nature of non-verbal gesture in open spaces, and the requirement on particular physical behaviour to carry the full communicational load (e.g. subtleties in gaze and facial expression are not visible at a distance).

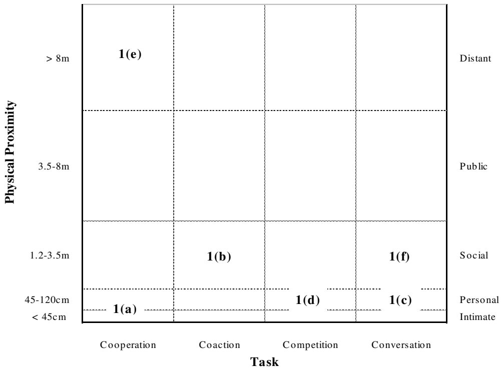


Figure 2: Task and spatial contexts.

Figure 1a shows an example of cooperation in intimate space. The male character demonstrates his attentional state – that he is attending to the female character – with his body orientation, face orientation, and gaze direction. Of course, people are rarely static, but different non-verbal channels (e.g. face orientation, body orientation, gaze direction, body position) are closely coordinated in demonstrating attention. Thus, the male character could look away but still communicates his attention sufficiently with the rest of his body. It should be noted that between unfamiliar participants strong or constant facing or looking at a person is widely considered an aggressive signal, or is considered rude, or at least off-putting, in an interaction (Knapp and Daly, 2002, p380). Figure 1a also illustrates non-verbal communication using facial expressions and kinaesthetic (touching) behaviour.

Figure 1e illustrates a situation at the other end of the spatial scale, cooperation at a distance between the player and a non-player character (in fact, navigation and negotiation, a subset of cooperation). The non-player character shown and the player will collide if they do not arrive at an agreement as to how to pass one another and communicate this – the characters must cooperate through the use of non-verbal communication to resolve a potential conflict. In the real world, people in this situation use a range of subtle non-verbal mechanisms such as gaze and body turning to initiate and mutually negotiate space. Non-player characters in Half-Life 2 will avoid the player, but will not exhibit non-verbal communication in doing so and simply move around the players as they approach. It would clearly be more realistic for the non-player characters to use non-verbal communication as humans do to negotiate past the player. Without this level of non-verbal behaviour it is difficult for the player to decide which way to move out of the way (indeed they do not need to) and it might be argued that this absence of social convention (and the ability to break them) both undermines the engagement of players with the game and limits their expressivity. For example, walking into another character and the social conflict that arises may serve to further or undermine a player's ends.

Between the proximal and distant spatial scales are social spaces and figure 1f is an example of conversation in a social space. Here non-verbal behaviour facilitates a number of aspects of the interaction (and the dialog in particular) including the mediation of conversation flow, such as whose turn it is to speak (interactional state). Turn-taking mediation is a complex coordination of behaviours, but in simple terms speakers provide turn-swap opportunities (such as, a slightly prolonged pause, or a look up into the eyes), at which point other listeners *can*, if they choose, take a turn. If not, then the speaker will continue. Additionally, the non-listener can indicate that they would like to speak, with signals such as: increased eye contact; leaning forward; standing taller (Duncan and Fiske, 1977). Turn-taking mediation is not required in Half-Life 2 because the game developers have not allowed the player to speak, but it is potentially a very important component of games that hope to include natural language interactions (particularly spoken interaction) between player and non-player characters.

Finally, figures 1b, 1c and 1d illustrate the remaining task contexts: coaction, conversation and cooperation, and competition. Characters sharing the same approximate area of space engage in coaction behaviour, corresponding to mutual monitoring – this can be interpreted as communication by virtue of the fact that watching a character implies that you might react to it – that is, there is an implied reason (intention) for watching. Coaction can be considered the default task context, which develops into the other contexts. Competition contexts give rise to distinctly different forms of non-verbal communication from other contexts, but these still serve to communicate internal states. In figure 1d the raised baton serves to communicate "you have crossed a line – back off or I will hit you".

3 Non-verbal communication

Non-verbal communication in non-player characters should be based on the non-verbal communication that people actually use and exhibit in human-human interactions. Hence, the forms of non-verbal communication that are particularly relevant to the synthesis of non-verbal communication in virtual characters are: proxemics (the use and arrangement of the self in the physical world); gaze (where the eyes are looking and pupil dilation); gesture (movement of hands and head); self-adaptors (movements that serve to alter the self); passive communication (communication that occurs without a specific action).

It should be noted that there is a very significant variation among individuals in the precise nature of their non-verbal communication. Non-verbal communication is influenced by age; gender; social class/hierarchy position; culture; presence of others; illness; physical ability, as well as the spatial and task context. This presents problems for both the study and the synthesis of non-verbal behaviour. Fidelity in non-verbal communication requires that each distinct non-player character employs its own characteristic behaviour in this respect. In the following sections we characterise non-verbal communication in people, generally from a Western/Anglo-Saxon cultural perspective, with a view to identifying which aspects should be addressed in the design of engaging virtual characters for computer games.

3.1 Proxemics

Proxemic communication is defined as the closeness and arrangement of the self in physical world compared to others (Hall, 1966). This is the use of personal space and territory, and although it is difficult to get an accurate sense of distance in games (people rarely feel their personal space is invaded in games), the use of space does have an impact on player behaviour. Moreover, as games become more immersive (due to improved graphics, larger screens, head-mounted displays, immersive environments, and surround sound) and dialog with non-player characters becomes a common occurrence, the impact of proxemics will increase. As discussed, non-verbal communication varies significantly with distance (and spatial arrangement), so its consideration is therefore important in the synthesis of non-player character behaviour.

Proxemics also exhibit some of the largest (but consistent) cultural variations, especially in conversation. In order of increasing distance (or decreasing proximity) the spaces for UK/US culture

(Hall, 1966) are: *intimate space* for embracing, touching or whispering (15-45 cm, 6-18 inches); personal space for interactions among good friends (45-120 cm, 1.5-4 feet); social space for interactions among acquaintances (1.2-3.5 metres, 4-12 feet); public space used for public speaking (over 3.5 metres, 12 feet). Additionally, we can define *distant space* for interaction at over 8 metres (25 feet), where people still interact but do not conduct conversations.

Proxemic behaviour stems from the idea of territory. Territories vary dynamically and are dependent upon many different factors, but the categories above are likely to be sufficient for basic synthesis. The requirement for synthesis is that non-player characters use proxemics as people do – to be socially correct use: social space for interacting with relatively unknown people; personal space for people known well; intimate space with people known very well. Violation of these rules is socially incorrect. Invasion of personal space can be intimidating, flirtatious, or could be the result of interaction between participants from different cultural backgrounds. Whilst the requirements of a task, such as needing to be close to attend to a wound, may override the usual norms, in such circumstances the progression through the spaces is still mediated by non-verbal behaviour.

The synthesis of complex proxemic behaviour requires further parameterisation with respect to types of territory: primary; secondary; public (Altman, 1975). Primary territories are the exclusive domain of the owner (such as a home), secondary territories are those felt to be partly owned (such as the local pub), and public territories those available to almost anyone for temporary ownership (such as a park bench). The idea of temporary ownership, is important as the temporary ownership means a person behaves very differently towards that object or space while they 'own' it (see Knapp and Daly, 2002).

3.2 Gaze and eye-based communication

The use of the eyes is an important component of human-human communication, and it involves far more than just what a person is looking at. Peoples' eyes are constantly moving from place to place (not smoothly, but jumping from one locus of attention to another). As Knapp and Daly note, "we associate various eye movements with a wide range of human expressions: downward glances are associated with modesty; wide eyes with frankness, wonder, naïveté, or terror" (2002, p370). Where the eyes are looking (or gaze) is the primary (and most obvious) form of communication by gaze, but people are also sensitive to, and react to, pupil dilation/constriction. There is also a close relationship between gaze behaviour and facial expression (not considered here).

Kendon (1967) identifies four functions of gaze behaviour (in addition to looking at specific items for information gathering), and Knapp (2002) builds on this, classifying five functions of gaze as:

1. regulating the flow of communication;
2. monitoring feedback;
3. reflecting cognitive activity;
4. expressing emotions;
5. communicating the nature of an interpersonal relationship [added by Knapp]

Within the context of a non-player character, the five categories pose challenges for synthesis that are considerably more challenging than simply having the character look at where it is interested. The regulation of communication flow, gazing briefly at another person (specifically at the face) establishes an obligation to interact; further and longer gazing shows a desire to increase the level of interaction; while decreased and shorter gazing desires a decrease in the level of interaction. During an interaction eye glances serve as turn-taking signals and also highlight grammatical breaks, conceptual unit breaks, and the ends of utterances (a sequence of speech separated from another by a marked gap). These glances also allow feedback on the interaction by monitoring the reactions of the other person.

When under increased cognitive load (trying to process difficult or complex ideas) both listeners and speakers tend to look away. It is thought that this averted gaze reflects a shift in attention from the external to the internal. Additionally, there is evidence that the eye gaze direction under this condition changes with different forms of cognitive load, and that it is linked to the active hemisphere of the brain (Ehrlichman and Weinberger, 1978; Weisz and Adam, 1993; Wilbur and Roberts-Wilbur, 1985).

The eyes are also a site for the display of emotions: surprise; fear; disgust; anger; happiness; sadness, in addition to blends of these and more complex emotions. Interestingly, in certain contexts emotions displayed with the eyes will not always match the facial emotional expressions (e.g. during emotional masking) and can be very transitory. However, people are adept at detecting emotional state from the eyes, and there is evidence that different emotions are in fact detected from different areas around the face (Ekman et al., 1971; Ekman and Friesen, 1975), but that it is the facial area *around* the eyes that displays the emotion, not the eyes themselves.

Finally, eye gaze can communicate the nature of an interpersonal relationship. Gazing and mutual gazing is found most in conversations with a high-status addressee, lower with a very high status addressee, and minimal with a very low-status addressee (Hearn, 1957; Efran, 1968). And, of course, lovers have extended periods of mutual gazing (this also occurs with mothers and babies).

3.3 Gesture

Gesture is body movement that serves to communicate, mainly involving movement of the hands and the head. There are many forms of gesture, but of particular relevance for non-verbal communication synthesis in non-player characters are emblematic gesture (gesture with specific meanings that occur without speech) and spontaneous gesture (hand and head movements that occur with speech). Emblematic gestures are well defined in both their form and meaning and are therefore readily synthesised with standard skeletal animation and scheduling frameworks. For example, the 'come here' gesture is performed using the moving of a finger, fingers, hand, hands, arm, or arms towards the body from the direction of the addressee (often in a repetitive form). Synthesis of spontaneous gesture for non-player characters is significantly more challenging.

Spontaneous gesture is performed by people while speaking, in synchrony with the speech and is generally made with the head or hands. When needs be people are apt to use any available body part, or even the whole body (e.g. pointing with a foot when one's hands are full). Spontaneous gesture continues to be studied in depth by the psychology and psycholinguistics communities, and though most studies are descriptive in nature, recent research on growth points is leading to theories of how gesture relates to other cognitive processes (McNeill Lab, University of Chicago, 2006; McNeill, 2005).

It has been found that the gesture stroke (the semantic, or meaningful part) of the gesture commonly coincides with the *peak phonological stress* – the most emphasised phoneme – of the speech stream. Gesture is tied closely to the underlying speech and both speech and gesture are widely believed to be generated from a single underlying conception (McNeill, 1992). Indeed, spontaneous gesture can be complementary, supplementary, or contrastive to the speech. In other words, gesture can re-iterate or emphasise the speech, add information to the speech, or communicate something contradictory (or slightly different) from the associated speech. In contrast to speech, gesture has few constraints on how it is constructed. As McNeill notes (1992, p1) "the important thing about gestures is that they are *not* [original emphasis] fixed. They are free and reveal the idiosyncratic imagery of thought." McNeill identifies five categories of spontaneous gesture:

1. emphatic;
2. deictic;
3. cohesive;
4. metaphoric;
5. iconic.

Emphatic gesture (also known as beat gesture, or baton gesture) provides emphasis to: parts of speech; phrases; words; phonemes. Emphatic gesture consists of just two movement phases (up/down, in/out, left/right) with the transition from one phase to the other (such as, up to down) being the point of emphasis. These have little variation in form other than the scale and speed of the phase transition, with larger, faster transitions meaning more emphasis (within an individual). Emphatic gesture can, and frequently does, utilise all body parts, especially the head and hands, but additional movement of more of the body provides further emphasis. This form of spontaneous gesture is distinct from other forms in that it can overlay any other gesture as it indexes a part of speech rather than providing semantic content (though it is also commonly used independently).

Deictic gestures are simply pointing actions that refer to an object or objects, generally using fingers, hands, or head. This is complicated by the fact that deictic gesture can, in addition to referencing concrete objects, reference more abstract objects, such as where an object was previously, the physical space referred to previously with the introduction of an idea or object, or almost any abstract space, including time. For example when describing a cartoon involving two characters people will often reference one specific area of their gesture space for one character and another separate area for the other (McNeill, 1992). It should be noted that deictic gestures can take the form of whole body movement towards a space as well as the orientation of body parts.

Cohesive gestures serve to connect related parts of discourse that are temporally separated. For example, when listing items people often provide an emphatic gesture on each item. The emphatic gesture clearly marks each item, while the repetition of the same gesture form connects them together to say 'here's *one*, and *another*, and *another*, and *another* ...'. Cohesive gesture usually consists of the repetition of a specific gesture and so they require the use of other gesture forms (which could simply be emphatic gesture). Finally, metaphoric and iconic gestures are in essence pictorial. An iconic gesture is a pictorial animation of a concrete entity or action, while metaphoric gesture animates an abstract concept as if it were a physical object. For example, a speaker saying 'and he bends it way back' while illustrating the action of bending a stick is performing an iconic gesture (McNeill, 1992, p12). In contrast metaphoric gesture occurs in situations such as when a speaker says 'I had this great idea' and marks the 'great idea' with a cupped hand gesture (i.e. information container metaphor). Metaphoric gesture takes abstract ideas and grounds them in real entities. In practice only a few types of object are portrayed, mostly containing or enclosed objects, though it is often not clear what object is being portrayed – it is just *some* object.

Iconic gestures are probably the most sophisticated and developed class of gesture; each gesture attempts to portray some aspect(s) of a situation, event, or object in the physical world and therefore absorb much of the complexity of the physical world. Typically the gesture portrays the most important and semantically salient features in the context of the interaction, and therefore a gesture about a specific object, such as a teacup, can vary significantly depending on the context. For example, a person talking about a teacup in context of drinking tea may perform a gesture of lifting a teacup by its handle, while if the context was about how much tea was in the cup, the gesture would be distinctly different – perhaps portraying the size of the teacup as opposed to how it is lifted. Iconic gesture is the form of gesture that most allows for the addition of extra information in the gestural channel and almost anything is permitted. For objects or ideas that are common in interaction the gesture forms portraying them can often become stylised, and eventually can become symbols, akin to emblems.

In addition to gesture accompanying speech, gesture can be used in a similar way to eye gaze and many vocal signals to regulate communication flow and the rhythm of interaction (Knapp and Daly, 2002, p252-3). Head nods are the most frequent form of these gestures, but hand and body gestures can also serve for flow regulation. Unsurprisingly, flow regulation gesture frequently coincides with flow regulation signals in other channels.

3.4 Self-adaptors

Self-adaptors are movements that serve to change the self, such as scratching an arm; as such they are typically not intended to serve a communicative function – their purpose is to adjust something about the self, but people read meaning into self-adaptors and so it can be considered a channel for communication. Self-adaptors can take on many forms and are distinctly idiosyncratic. The meanings assigned to them by conversational participants and bystanders vary considerably. For example, flicking hair out of the face, while a practical movement, is often interpreted as a flirtatious behaviour.

3.5 Passive Communication

Finally, another consideration to be taken into account is passive communication – communication that occurs without a specific action on the part of the communicator, and usually takes place at the co-action level of communication, and at a greater distance than conversational communication.

The communicator expresses information about themselves simply by their physical appearance, demeanour and observed behaviour. These factors can be considered by the user (or indeed by other non-player characters) before initiating a more direct method of communication. For example, a large aggressive character might be approached with more caution than a small timid looking character. Similarly, witnessing a character commit some violent act would instil more caution in the user than a character who has been doing something less impactful.

Passive communication is often unintentional at the time of communication itself, but can be premeditated, for example wearing a certain type of clothing might convey a specific message.

4 Facilitating interaction

The discussion of non-verbal communication above concentrates on the different forms of non-verbal communication in an attempt to establish requirements for next generation virtual characters. Mechanisms for the synthesis of such behaviour are inevitably beyond the scope of this analysis but there have been a number of recent attempts to synthesise spontaneous gesture (Kopp and Wachsmuth, 2004; Olivier, 2003; Cassell, Vilhjálmsson, and Bickmore, 2001) and gaze behaviour. However, whilst we can imagine a situation whereby non-player characters have sophisticated cognitive models and the ability to both synthesise and interpret non-verbal behaviour, there is no mechanism for the player to communicate non-verbally with either the non-player characters or other players.

Present-day technology allows collection of full data on all aspects of human physical behaviour that may be a channel for non-verbal communication. This includes body position, body movement, hand shape, eye gaze direction, pupil dilation, facial expression, vocal behaviour, voice, and a variety of other biometrics. Unfortunately, such data can only be collected accurately using specialised (expensive) invasive equipment. With that commercial consideration, in addition to joystick (or controller), keyboard and mouse inputs it is only reasonable to expect relatively basic additional input devices for computer games in the near future. These devices include webcams and microphones, but also less standard input methods, such as ‘Dance mats’, light guns, and low point motion capture devices, such as the Gametrack 3D motion tracker.

Within current commercial and technological constraints, three alternatives for the control of player character non-verbal behaviour generation can be identified:

1. simulation – non-verbal communication of human players is simulated as for non-player characters, and is independent of the actual non-verbal communication of the human player.

2. augmentation – as for simulation, but specific controls are given to allow a player to explicitly alter the simulation, such as a slider to indicate how happy the player is, or a button to increase the level of interest in an interaction.
3. tracking and mapping – the human player is tracked using equipment such as webcams and the coarse features that can be identified are either mapped to explicit controls (as in the case of augmentation) or directly to the character animation.

The challenge for augmentation is to design an interface that is intuitive, non-obtrusive, and useful all at the same time. While the challenges in player non-verbal communication data collection are daunting, the challenges of understanding or recognising behaviours or meaning from that data are even more so. Gesture recognition, for example, is in its infancy and mostly addresses the use of gesture as an explicit interaction technique and little research has been conducted into the recognition or understanding of spontaneous gesture.

References

I. Altman. *The environment and social behaviour*. Brooks/Cole, Monterey, CA, 1975.

J. Cassell, H. Vilhjálmsson, T. Bickmore. BEAT: the behavior expression animation toolkit. *Proceedings of SIGGRAPH '01*:477-486, Los Angeles, CA, 2001.

S. Duncan and D.W. Fiske. *Face-to-face interaction: research, methods, and theory*. L. Erlbaum Associates, Hillsdale, N.J., 1977.

J.S. Efran. Looking for approval: Effects on visual behaviour of approbation from person differing in importance. *Journal of Personality and Social Psychology*, 10:21-25, 1968.

H. Ehrlichman and A Weinberger. Lateral eye movements and hemispheric asymmetry: A critical review. *Psychological Bulletin*, 85:1080-1101, 1978.

P. Ekman and W.V. Friesen. *Unmasking the face*. Prentice-Hall, Englewood Cliffs, NJ, 1975.

P. Ekman, W.V. Friesen and S.S. Tomkins. Facial affect scoring technique: A first validity study. *Semiotica*, 3:337-58, 1971.

S. Kopp and I. Wachsmuth. Synthesizing Multimodal Utterances for Conversational Agents. *The Journal of Computer Animation and Virtual Worlds*, 15(1):39-52, 2004.

E.T. Hall. *The hidden dimension*. Doubleday, Garden City, N.Y., 1966.

G. Hearn. Leadership and the spatial factor in small groups. *Journal of Abnormal and Social Psychology*, 54:269-72, 1957.

A. Kendon. Some functions of gaze-direction in social interaction. *Acta Psychologica*, 26:2622-63, 1967.

M.L. Knapp and J.A. Daly. *Handbook of interpersonal communication*. SAGE Publications, Thousand Oaks, CA, 2002.

Maxis, *The Sims*, Electronic Arts 2000.

D. McNeill. *Hand and mind: what gestures reveal about thought*. University of Chicago Press, 1992.

D. McNeill. *Gesture and thought*. University of Chicago Press, Chicago, IL, 2005.

McNeill Lab. *Center for Gesture and Speech Research*. http://mcneilllab.uchicago.edu/ (from February 2006).

M. Masahiro. *Bukimi no tani [The Uncanny Valley]*, Energy(7):33-35, 1970.

P. Olivier. Gesture Synthesis in a Real-World ECA, *Lecture Notes in Computer Science*, 3069:319-322, Springer, 2004.

B. Reeves and C.I. Nass. *The media equation: how people treat computers, television, and new media like real people and places*. Cambridge University Press, New York, 1996.

Rockstar Games. *Grand Theft Auto 3*. Edinburgh, Scotland, 2002

Valve Corporation. *Half-Life 2*. Valve Corporation. Bellevue, Washington, 2004

J. Weisz and G. Adam. Hemispheric preference and lateral eye movements evoked by bilateral visual stimuli. *Neuropsychologia*, 31(12):1299-1306, 1993.

M.P. Wilbur and J. Roberts-Wilbur. Lateral eye-movement responses to visual stimuli. *Perceptual and motor skills*, 61:167-177, 1985.

Social Norms Models in Thespian: Using Decision Theoretical Framework for Interactive Dramas

Mei Si*
*Information Sciences Institute
Univ. of Southern California
Los Angeles, CA 90292
meisi@isi.edu

Stacy Marsella*
†Information Sciences Institute
Univ. of Southern California
Los Angeles, CA 90292
marsella@isi.edu

David Pynadath*
‡Information Sciences Institute
Univ. of Southern California
Los Angeles, CA 90292
pynadath@isi.edu

Abstract

Social norms are shared rules of behavior that facilitate social interaction. Although norms are commonly followed, other factors such as personality traits or more pressing goals may nevertheless lead to behavior that violates norms. To facilitate *life-like* social interaction with users in interactive dramas, the virtual characters ideally should follow similar social norms as that in human-human interaction. In many existing interactive dramas, the effects of social norms are often crafted by the author to integrate them with goals and personalities of characters. In this paper, we present a principled way to model social norms in a decision theoretic framework, Thespian. In Thespian, characters have explicit goals of following social norms in addition to their other goals. Characters can reason about the effect of following or violating social norms the same way as achieving or sacrificing their other goals. They can therefore reason about conflicts between social norm goals and other goals. Different characters can weight social norm goals with respect to other goals in different ways. We discuss the model of social norms in Thespian. We also present preliminary experiments on testing the efficiency and necessity of Thespian's social norm model for virtual characters.

1 Introduction

The design of interactive stories faces several key challenges. As in any story, narrative structure must be integrated with portrayal of characters. For an interactive story, this integration must in addition incorporate user interaction into how the story unfolds and how characters act. Characters in particular should respond "in character", consistent with the characters' motivations in the story. Since interactive stories are typically designed to provide users life-like social interactions, character motivations should also be life-like. Further, the design of interactive stories seeks to encourage the user's interaction. However, the more open-ended the user interactions, the more difficult it may become to craft an experience that balances story, character and interaction.

Due to such challenges, interactive stories can be a challenge to create. Indeed, one of the research challenges for interactive stories is to find ways to transform the authoring process from an onerous burden, of detailing all the ways the story may unfold, to a more creative exercise (Gebhard et al., 2003; Mateas and Stern, 2003; et al., 2004a; Si et al., 2005a).

In previous work on the Thespian system, Si et al. (2005b) have explored how multi-agent systems that have decision-theoretic agents can be used to address some of these challenges. Si et al. (2005b) argues that decision-theoretic goal driven agents are well suited for building characters in interactive dramas that are responsive to the user interactions while maintaining their motivational consistencies. Such agents can balance multiple competing goals, such as responding sociably to the user and not disclosing sensitive information while being asked a sensitive question. Si et al. (2005a) discusses how decision-theoretic agents can be *trained* to perform their roles according to linear story scripts provided by authors through a semi-automatic fitting process. This process can dramatically reduce authoring effort and ideally transform it into a more creative exercise of writing stories, a process that authors are more familiar with.

In this paper, the emphasis is on how Thespian creates characters with life-like motivations; in particular, how social norms are represented in Thespian.

Social norms play an important role in governing social interactions. Social norms serve as a guide for people's behaviors, and as the basis for their beliefs and expectations about others. Without commonly believed social norms, conversations can break down easily. For example, if there were no norms governing question answering, people would not be able to exchange information. To facilitate *life-like* social interaction with the user in interactive dramas, the virtual characters ideally should follow similar social norms as that in human-human interaction.

In many existing approaches to interactive dramas, norm-following behavior is embedded in the system design. Agent-based approaches often embed social norms in characters' behavioral rules. For example, the "action rules" for deciding action tendencies in FearNot (et al., 2004a) exclude inappropriate actions from receiving positive tendencies. In MRE (et al., 2001), the rules for dialog management will only generate action options relevant to the character's current status. In Façade (Mateas and Stern, 2003), the story is organized around dramatic beats, realized as brief patterns of interactions between characters; norms are encoded in individual beats and the process of beat selection. And in Cavazza et al. (2002)'s storytelling system, the characters' roles are represented in a consistent fashion as Hierarchical Task Networks (HTN).

Many variations in people's behaviors can be viewed to be caused by different norms being violated. For example, a person who likes lecturing others ignores their right to talk. And a person who does not answer questions ignores the norm to provide relevant information when being enquired. In fact, although norms are commonly followed, other factors such as more pressing goals may nevertheless lead to behavior that violates norms. In addition, people of different personalities may view various norms with different importance. The effects of social norms are constantly regulated by these factors. However, in the above interactive drama systems, the norms are largely implicit, the authors need to craft characters' behaviors to reflect such individual differences.

In this paper, we present a principled way to model social norms in a decision theoretic framework. In Thespian, characters have explicit goals of following social norms in addition to their other goals. We identified basic social norms for conducting meaningful communications, regulating turn-taking and conversation flow, and implemented them inside Thespian as goals and dynamics functions, that define how actions affect agents' states including their goals. The same dynamics functions are shared among all characters. Different characters can weight their social norm goals with respect to other goals in different ways. Thus, we allow characters to reason about the effect of following or violating social norms the same way as reaching or sacrificing their other goals.

Figure 1: A screen-shot from the Tactical Language Training System

2 Example Domain

Our social norm model is built within the Thespian framework that was used to realize the Mission Environment (Figure 1) of the Tactical Language Training System (TLTS) (et al., 2004b). TLTS is designed to teach the user a foreign language and cultural awareness. The user takes on the role of a male army Sergeant (Sergeant Smith) who is assigned to conduct a civil affairs mission in a foreign (e.g. Pashto, Iraqi) town. TLTS uses a 3D virtual world built on top of the Unreal Engine. The human user navigates in the virtual world and interacts with virtual characters using spoken language and gestures. An automated speech recognizer identifies the utterance and the mission manager converts them into a dialogue act representation that Thespian takes as input. Output from Thespian consists of similar dialogue acts that instruct virtual characters to speak and behave.

The story in TLTS consists of multiple scenes. We will use a scene from the Pashto version to illustrate the working of Thespian's social norm model. The story begins as the user arrives outside of a Pashto village. Some children are playing nearby and come over to talk to the user as the vehicle arrives. The user's aim in the scene is to establish initial rapport with people in the village through talking to their children in a friendly manner. The children possess different personalities. Some are very shy and some are very curious about the American soldier.

3 Thespian

Thespian is a POMDP (Partially Observable Markov Decision Processes)(Smallwood and Sondik, 1973) based multi-agent framework for authoring and simulating interactive dramas. Thespian is built upon PsychSim (Marsella et al., 2004; Pynadath and Marsella, 2005), a multi-agent system for social simulation. In Thespian, each character in a drama is controlled by a POMDP based agent. A human user can substitute any of the characters and interact with other characters. All characters in Thespian communicate with each other through dialogue acts. In their conversation, once a character gets the turn, it can speak multiple dialog acts, and nobody can interrupt. In its turn, the character decides one dialog act at a time. It indicates giving up the turn by selecting a special dialog act of "do nothing". In this section we give an overview of the major components of Thespian architecture.

3.1 Agent State

A character's state is defined by a set of state features, such as the name and age of the character, and the affinity between two characters. In Thespian, values of state features are represented as a range of real numbers within [-1, 1]. The size of the range indicates the character's confidence level about this value. If a value equals to [-1, 1], it means the character does not know what this value should be. On the other hand [.1, .1] indicates the agent is 100% confident of the value being exactly .1.

Goals are expressed as a reward function over the various state features an agent seeks to maximize or minimize. For example, Sergeant Smith has a goal of maximizing his *affinity with the children* with initial value set to [.0, .0]; this goal is completely satisfied once the value reaches [1.0, 1.0].

3.2 Dynamics Functions

Dynamics functions define how a character's actions can affect its state, the state of other agents, and the environment. These dynamics functions influence the agent's reasoning about the next action to be taken and hence their behavior.

Rules governing social norms of the character are modeled as a set of dynamics functions that describe each action's effects on social norm related state features, such as *impose_obligation_norm* and *satisfy_obligation_norm*, and various obligations an agent can have.

3.3 Agents' Beliefs and Action Selection

In Thespian, agents have a "theory of mind" which enables them to form mental models about other characters including the user. These mental models also allow a Thespian agent to reason about the effects of its behavior on its relationships with other characters. Each agent has its own goals, which include social norm related goals, such as maximizing *impose_obligation_norm* and *satisfy_obligation_norm*, other personality related goals, such as maximizing *self esteem*, and task related goals, such as *finding out direction to a person's home*. All agents use bounded lookahead (e.g. "which action will best achieve my goals in the near future, such as the next three steps") to choose their next actions.

3.4 Fitting Procedure

Given a particular environment with state features and dynamics functions, the goals of an agent decide its behavior. Variations in the relative importance of goal items will result in different agent behaviors. Thespian has an automatic fitting procedure that can translate a character's desired behavior from sequences of dialog acts into the goals needed for the agent to autonomously select those behaviors (Pynadath and Marsella, 2004; Si et al., 2005a). This mechanism compiles the agent's policy of behavior into an invertible piecewise linear function of its goals. We can thus translate desirable behaviors (e.g. scripts) into constraints on goal weights, supporting the automatic configuration of characters.

3.5 Dialogue act Definition

In Thespian, dialogue acts are defined as a tuple with five fields, type, speaker, addressee, proposition, and attitude. Algoritm 1 shows the definition of dialogue acts in BNF format.

The *speaker* is the character that performs the dialog act. The *addressee* is the character or characters who are addressed in the dialog act. The *type* specifies the type of the dialogue act. The *proposition* specifies the belief being mentioned in the dialogue act. The propositional representation contains a recursive structure that directly corresponds to the underlying recursive belief structure in Thespian. For example, the proposition "*entity*: town , *attribute*: name, *value*: [-1, 1]" means the speaker has no knowledge about the name of the town; while the proposition "*entity*: town , *attribute*: leader, *attribute*: name, *value*: [-1, 1]" means the speaker has no knowledge about who is the leader of the town.

Algorithm 1 BNF of Dialog Act

```
⟨dialog act⟩→ ⟨type⟩⟨speaker⟩⟨addressee⟩⟨proposition⟩
        ⟨attitude⟩
⟨speaker⟩→ ⟨character⟩
⟨addressee⟩→ ⟨character⟩+
⟨type⟩→ initiating greeting | respond to greeting | initiating bye | respond to bye | thanks | you are
        welcome | enquiry | inform | request | accept |
        reject | convey information
⟨proposition⟩→ ⟨attribute⟩ ⟨entity⟩ ⟨value⟩
⟨entity⟩→ ⟨attribute⟩ ⟨entity⟩ | ⟨entity⟩
⟨entity⟩→ ⟨character⟩ | ⟨object⟩
⟨attitude⟩→ [formality] [politeness] [strength] [positive face redress] [negative face redress]
```

The *proposition* field does not apply to all dialogue acts, as some of them, such as *initiating greeting*, do not require one. Finally, the *attitude* field holds factors that describe how the speaker performs this dialog act.

4 Social Norms

The main reason we want to build our virtual characters with ability to understand and follow social norms is to facilitate their interactions with human users. People would find it difficult to interact with the characters if they could not interpret and predict the characters' behaviors completely.

Social norms describe general expectations in social interactions. What people expect in communication varies in different language settings. We model those social norms relevant to face to face communication, what Clark (1996) calls the *Personal* setting, because it is most commonly used (Clark, 1996) and most relevant to our stories.

4.1 Updating Obligations

Adjacency pairs (Schegloff and Sacks, 1973), such as greetings and greetings, question and answer, assertions and assent, are very common in conversations. We model obligations involved in adjacency pairs. For each adjacency pair and each of its relevant propositions in the story, a separate state feature is used to represent the obligation. For example, the user may have obligation to greet back to Xaled, or reply his name to Xaled. If a character performs the first part of an adjacency pair, it imposes an obligation on the addressee of its action to perform the second part. By performing the action desired by the first character, the second speaker can satisfy the obligation imposed on it.

In addition, after imposing an obligation, the speaker needs to stop talking to give the addressee a turn to respond. Therefore, in Thespian, when a character imposes an obligation on another character, it at the same time imposes an obligation on itself to wait for responses. This obligation will be satisfied after getting a response from other characters.

Next we will present a set of dynamics functions that define how social norms related goals are updated in Thespian. Characters have goals on maximizing all of the goal features. These dynamics functions enable virtual characters to behave life-like in terms of three aspects: conducting meaningful communications, emerging natural turn taking patterns, and having appropriate conversation flow.

4.2 Social Norms for Conducting Meaningful Conversations

People impose obligations onto each other to trigger desired responses. *satisfy_obligation_norm* enables characters to interact in the same way. Also for the communication to be succinct, we included the *no_repeat_norm*.

update satisfy_obligation_norm

```
if $self == dialogact.speaker$ then
    if $dialogact$ satisfies an existing obligation then
        return original_value+0.1
    else if $dialogact$ attempts to satisfy a non-existing obligation then
        return original_value-0.5
return original_value
```

update no_repeat_norm

```
if $self == dialogact.speaker$ then
    if $dialogact$ has already happened then
        return original_value-0.1
return original_value
```

4.3 Social Norms for Turn Taking

In addition to motivating characters to choose the right content to talk about, we want the conversation to exhibit natural turn taking behaviors. Sacks et al.

(1974) summarized three basic rules on turn-taking behaviors in multiparty conversations. In Thespian, we simulate Sacks' first rule by enforcing adjacency pairs. The *impose_obligation_norm* prevents characters from imposing new obligations and perform obligation irrelevant actions when somebody in the conversation still has unsatisfied obligations. Hence, only the characters that have unsatisfied obligations will not get punished for taking the turn to act. If the dialog act performed in the current turn is aimed at satisfying an existing obligation, we count it as a case of the current speaker not selecting the next one.

Sacks' Rules on Turn Taking

1.If a party is addressed in the last turn, this party and no one else must speak next.

2.If the current speaker does not select the next speaker, any other speaker may take the next turn.

3.If no one else takes the next turn, the current speaker may take the next turn.

---

update impose_obligation_norm

```
if $self == dialogact.speaker$ then
    if $dialogact$ does not satisfy an existing obligation then
        for $character$ in $conversation$ do
            if $character$ has unsatisfied obligations then
                return original_value-0.1
return original_value
```

In addition, face to face conversation is different from lecturing; *keep_turn_norm* prevents any character from dominating the conversation. If a character keeps talking after reaching the maximum number (currently set to 2) of dialog acts it can perform within a conversational turn, its degree of achieving this goal decreases. The counter of dialog acts will reset to zero only after another character starts speaking. In the case when the turn is free to be taken by anybody, *keep_turn_norm* prevents the previous speaker from taking the turn again. This is consistent with what is described in Sacks' second and third rules.

update keep_turn_norm

```
if $self == dialogact.speaker$ then
    if $self.sentences\_in\_current\_turn > 2$ then
        return original_value-0.1
return original_value
```

4.4 Social Norms for Conversation Flow

Furthermore, we want conversations to exhibit the right structure. Conversations normally have an opening section, body and closing section (Clark, 1996). In Thespian, we use a state feature *conversation* to keep track of what a character thinks the current status of the conversation is. Initially the value for *conversation* is "not opened". Once a character starts talking to another, the value changes to "opened". After the conversation finishes (judged by characters walking away from each other, or no eye contact for a long time), the value of *conversation* is changed back to "not opened". We use *conversation_flow_norm* to enforce an appropriate conversation flow. The character that opens the conversation should open with proper greeting; and if a character ends a conversation, it needs to have said *bye* to other characters. Else, the value of this goal feature will get reduced.

update conversation_flow_norm

```
if $self == dialogact.speaker$ then
    if $self.conversation$ == 'not opened' then
        if $dialogact.type$ != 'initiate greeting' then
            return original_value-0.1
    else if $dialogact.type$ == 'end conversation' then
        if characters have not said bye to each other then
            return original_value-0.1
return original_value
```

4.5 Affinity

Finally, we want to consider the effect of affinity. In order to take place, most social interactions require the affinity between the two characters involved to be within a certain range. Some social interactions require closer affinities than others. For example, greeting, saying "thanks", saying "bye", and asking about time can happen between almost any two characters. While asking potentially sensitive questions, e.g. who is the leader of the town, usually requires closer affinity.

To enable characters to anticipate that their obligation imposing actions may not get desired responses, we augmented *impose_obligation_norm* with *affinity*. If satisfying an existing obligation requires closer affinity between the two characters than what it is currently, ignoring this unsatisfied obligation will result in much less punishment than if the affinity between the two characters is appropriate. The augmented rule will allow characters to ignore unreasonable requests, such as an enquiry of personal information from a stranger.

Since affinity is an important factor in social norms, we briefly describe how affinity is updated.

Affinity is affected by whether the characters act following social norms. The logic behind this rule is people feel closer to each other after having successful social interactions; and people will not feel close to a person whose behavior is unpredictable.

Affinity is also affected by the attitude accomplished with a dialog act. Currently, we use a simple model that only takes one rule into account. If the dialog act is performed in an impolite manner, it will decrease the affinity between the speaker and the addressee.

Finally, the main effect of many types of dialogue acts is to change affinity. For example, the following dialogue acts while not violating social norms can always increase affinity between two characters: compliments, small talk such as asking "how are you", "how is your family", and giving offers. And some other dialogue acts, such as accusing, once performed will usually reduce the affinity between two characters.

5 Examples

There are four main characters in the story, three children and Sergeant Smith. The children's names are Hamed, Xaled, and Kamela. The possible actions for the characters are greeting each other, asking each other various questions, answering questions, saying good-bye to each other, and introducing information about oneself to others. The last action can increase the affinity between Sergeant Smith and the children and doesn't impose any obligations for replying.

Each of these four characters has the goals of following social norms, and several other goals including collecting information from each other. Sergeant Smith wants to have a close affinity with the children, and wants to know from the children their names, the name of the adults close by, etc. The children on the other hand are curious about what Sergeant Smith's nationality is, and how much Pashto he understands, etc. These goals on information collection can be fully achieved once the character gets the piece of information it is interested in. In addition, the children need their affinity with Sergeant Smith to be close enough to feel comfortable telling their parents' names, but can answer other questions without considering affinity. In the following examples, to demonstrate the effect of varying goal weights on agents' behaviors, Sergeant Smith is controlled by an agent. The same agent is used to provide the user hints for what to do next in the simulation.

Example 1:

1. Sergeant Smith to Kids: Hello!
2. Xaled to Sergeant Smith: Hello!
3. Hamed to Sergeant Smith: Hello!
4. Kamela to Sergeant Smith: Hello!
5. Sergeant Smith to Xaled: What is your name?
6. Xaled to Sergeant Smith: My name is Xaled.
7. Xaled to Sergeant Smith: What is your name?
8. Sergeant Smith to Xaled: My name is Mike.
9. Sergeant Smith to Xaled: This is my aide Abasin.
10. Xaled to Sergeant Smith: Are you an American?
11. Sergeant Smith to Xaled: Yes, I am an American.
12. Sergeant Smith to Xaled: I come from Texas.

...

Example 1 is a sample dialog in which obeying social norms dominates all other goals for all the characters. In line 1 of example 1, Sergeant Smith chooses to greet the children first because performing any other action will result in opening the conversation inappropriately (*conversation_flow_norm*). Then Sergeant Smith chooses to give up the turn, because of *impose_obligation_norm*, the action he just performed has imposed obligations for the children to reply, as well as an obligation for him to wait for replies.

Each child greets back in his/her turn because of *satisfy_obligation_norm*. Xaled and Hamed do not impose new obligations onto Sergeant Smith after greeting because they know Kamela has not greeted back yet (*impose_obligation_norm*).

In line 6, Xaled satisfied his obligation and knows that nobody in the conversation has unsatisfied obligations. Xaled is then free to impose an obligation on Sergeant Smith to satisfy his goal of curiosity.

Lines 6-7, 8-9, and 11-12 demonstrate the effect of *keep_turn_norm*. Especially in the last two examples, even though introducing himself more will further increase affinity, Sergeant Smith chooses to follow social norms by not holding the turn too long. Also, because of *no_repeat_norm* Sergeant Smith tries to introduce different information each time.

Lines 8-12 also show the effect of affinity. Sergeant Smith does not ask the names of the children's parents, but chooses to increase affinity first.

We can create interesting character personalities easily by varying the pattern of norm weights. We can create a rude character by letting it not respect *conversation_flow_norm*; a character who does not like to respond to others by letting it not respect *satisfy_obligation_norm*. As we will see, the Sergeant Smith character in example 3 can be viewed as a combination of these two types. In addition, we can create a talkative character by giving a low weight on *keep_turn_norm*, a character who likes to interrupt other people's conversation by lowering the weight of *impose_obligation_norm*, and a character who keeps on repeating sentences he believes are important by lowering the weight of *no_repeat_norm*. Combining the possible weights of all norms gives us a big space for creating interesting characters.

Next, we want to show how the goals of following social norms interact with other goals to decide a character's behavior. Since the agents are decision theoretic we can get this effect easily. In example 2, we make Sergeant Smith's information gathering goals his most important goals. As a consequence, Sergeant Smith does not respect having proper conversation flow and ignores obligation imposed on him to answer question. All his actions are aimed at gathering the information he wants.

Example 2 :

1. Sergeant Smith to Xaled: What is your name?
2. Xaled to Sergeant Smith: My name is Xaled.
3. Xaled to Sergeant Smith: What is your name?
4. Sergeant Smith to Xaled: What is the name of this town?

...

Finally consider an extreme case in which none of the characters respect social norms; however, they each believe others all follow norms (This belief is important for motivating them to communicate, as we will show later). The important goals for them are to get the information they are interested in.

Example 3 :

1. Sergeant Smith to Xaled: What is your name?
2. Xaled to Sergeant Smith: What is your name?
3. Hamed to Sergeant Smith: Who is that man?
4. Kamela to Sergeant Smith: Do you have children?
5. Sergeant Smith to Xaled: What is your name?

...

The results are shown in example 3. The characters are not able to conduct a meaningful interaction. Since none of them answer other's questions, they will keep on asking for the information they are interested in knowing. What would happen if the characters did not even expect others to follow social norms? In this case, the conversation would totally break down. The characters would choose an action that can bring them maximum immediate benefit. But, in this story all of their non social norms goals require getting responses from others to get benefited, hence the characters will just choose actions randomly.

6 Discussion

The examples we presented in Section 5 have demonstrated that our social norm model is effective and necessary for enabling characters to have meaningful conversations.

As the norms we included are the most basic ones, we will be working on enriching our model. As part of future work, we want to extend our model to better support subgroups in conversations. The current model we built supports multiparty conversation, but mainly as an extension to two-party conversation, in the sense that characters only have goals on satisfying their own obligations. In the future, we want to support models of situations that characters have shared obligations, e.g. characters can answer questions for their friends, and a character can impose obligations onto a group of characters. In the latter case, each addressee has obligation to respond, however the obligation is different with when the character is the only addressee.

On the other hand, we are also interested to study how the norms (or dynamics functions in general) with different degrees of details affect user experiences in the interactive drama, both in terms of believability of the characters and immersive nature of the interaction.

7 Conclusion

Understanding social norms is the basis for people to interact with each other. People generate their own behaviors, form expectations about others, and interpret other people's behaviors, all based on social norms. To facilitate life-like social interaction with the user, the virtual characters ideally should be built with social norms similar to the ones that govern human-human interaction.

We summarized a set of basic social norm rules for face to face communication. These rules are implemented inside Thespian as goals and dynamics functions for decision-theoretic goal driven agents. We have demonstrated that Thespian's social norm model is effective in enabling characters to perform appropriately in social interactions. In addition, our experiments show that these norms are necessary for a meaningful conversation to get conducted. Without these norms, characters would not be able to reach their actual goals (other than their social norm goals).

The benefit of building our model within Thespian's framework is two-fold. First, because of the underlying POMDP model each character has, we can easily create the effect of social norms interacting with a character's personality traits and other goals in deciding the character's behavior. Secondly, we are able to support easy authoring of characters. Since the set of dynamics functions we defined are independent of a particular story or character, this same set of social norms can be applied to any character. Moreover, our social norm model is compatible with Thespian's automatic fitting procedure, which enables characters to learn to behave according to dialog act sequences specified by authors via automated tuning of goal parameters.

Our future work involves enriching our model, particularly for supporting subgroups in multiparty conversations, and studying how the levels of complexity embedded in the norms affect users' experiences in the interaction.

Acknowledgments

We thank our colleagues on the Tactical Language Training Project, especially Lewis Johnson and Hannes Vilhjálmsson for their support and thoughtful discussions. This project is part of the DARWARS initiative sponsored by the US Defense Advanced Research Projects Agency (DARPA).

References

Marc Cavazza, Fred Charles, and Steven J. Mead. Emergent situations in interactive storytelling. In *Proceedings of ACM Symposium on Applied Computing (ACM-SAC)*, March 2002.

Herbert Clark, editor. *Using Language*. Cambridge University Press, New York, NY, 1996.

Ana Paiva et al. Caring for agents and agents that care: Building empathic relations with. In *AAMAS*, pages 194–201, 2004a.

W. Lewis Johnson et al. Tactical Language Training System: An interim report. In *Proc. of the Internat'l Conf. on Intelligent Tutoring Sys.*, pages 336–345, 2004b.

William Swartout et al. Toward the holodeck: Integrating graphics, sound, character and story. In *Agents*, pages 409–416, 2001.

Patrick Gebhard, Michael Kipp, Martin Klesen, and Thomas Rist. Authoring scenes for adaptive, interactive performances. In *AAMAS*, pages 725–732, 2003.

Stacy Marsella, David V. Pynadath, and Steven J. Read. PsychSim: Agent-based modeling of social interactions and influence. In *Proc. of the Internat'l Conf. on Cognitive Modeling*, pages 243–248, 2004.

Michael Mateas and Andrew Stern. Integrating plot, character and natural language processing in the interactive drama façade. In *Proc. of the Internat'l Conf. on Tech. for Interactive Digital Storytelling and Entertainment*, 2003.

David V. Pynadath and Stacy Marsella. Fitting and compilation of multiagent models through piecewise linear functions. In *AAMAS*, pages 1197–1204, 2004.

David V. Pynadath and Stacy C. Marsella. Psychsim: Modeling theory of mind with decision-theoretic agents. In *IJCAI*, pages 1181–1186, 2005.

Harvey Sacks, Emmanuel A. Schegloff, and Gail Jefferson. A simplest systematics for the organization of turn-taking for conversation. *Language*, 50: 696–735, 1974.

Emanuel A. Schegloff and Harvey Sacks. Opening up closings. *Semiotica*, 7:289–327, 1973.

Mei Si, Stacy C. Marsella, and David V. Pynadath. Thespian: Using multi-agent fitting to craft interactive drama. In *AAMAS*, pages 21–28, 2005a.

Mei Si, Stacy C. Marsella, and David V. Pynadath. Thespian: An architecture for interactive pedagogical drama. In *AIED*, 2005b.

Richard D. Smallwood and Edward J. Sondik. The optimal control of partially observable Markov processes over a finite horizon. *Operations Research*, 21:1071–1088, 1973.

Affect Detection from Open-Ended Improvisational Text

Li Zhang, John A. Barnden, Robert J. Hendley, and Alan M. Wallington
School of Computer Science, University of Birmingham
Birmingham, B15 2TT, UK
`L.Zhang@cs.bham.ac.uk`

Abstract

We report progress on adding affect-detection to a program for virtual dramatic improvisation, monitored by a human director. We have developed an affect-detection module to control an automated virtual actor and to contribute to the automation of directorial functions. The work also involves basic research into how affect is conveyed through metaphor. The relevance of the project to the symposium is mainly in the application of AI to the creation of emotionally believable synthetic agents for interactive narrative environments, and in the study of the language used in improvisatory story-construction.

1 Introduction

The benefits of using improvised drama or role-play in education, training, conflict resolution and counselling are widely recognised, and researchers have recently explored the possibility of e-drama, in which virtual characters (avatars) on computer displays interact under the control of human users (e.g. Machado et al., 2000). This paper focuses on our experience[1] in adding types of intelligent processing to an existing e-drama system, *edrama*, created by one of our industrial partners (Hi8us) and used in schools for creative writing and teaching in various subjects. The intelligent processing is focussed on the detection and assessment of emotional and other "affective" aspects of language utterances created freely by users.

Although the symposium is not focused on improvisatory drama systems as such, an important desirable aspect of various types of believable synthetic agent, in games or other environments, is the ability to be sensitive to the affect apparently expressed in natural-language expressions uttered by other participants, in order to be able to respond in an affectively appropriate way.

In the original *edrama* system mentioned above, "actors" (human users) control virtual characters on a virtual stage, with "speeches" displayed as text bubbles typed by the actor operating the character. One director and up to five actors are involved in an e-drama session. A graphical interface shows the characters and virtual stage on the director's and each actor's terminal. Actors can choose the clothes and bodily appearance for their own characters. So far, the characters' visual forms have been static cartoon figures, with real-life photographic images used as backdrops. However, we are now bringing in animated gesturing avatars and 3D computer-generated settings using technology from another industrial partner (BT), although these matters are outside the scope of the present paper.

Actors and a human director all work at separate terminals, through software clients connecting with the server. The clients communicate by XML stream messages via the server, over the Internet using standard browsers. The terminals are generally at a site or sites remote from the server site, and may be remote from each other.

The actors are given a loose scenario around which to improvise, but are at liberty to be creative. (There are no canned speeches and no firmly required plot elements.) For example, one scenario we have used is a school-bullying one involving a schoolgirl Lisa, who is being bullied by her classmate Mayid. Lisa is a shy child and afraid of Mayid. Other characters are Lisa's mother, a friend, and a schoolteacher. Actors are expected to improvise interesting interchanges within these parameters. It is expected that normally the Mayid character will express hostility to Lisa and that she will express fear, but actors can be creative, so that

---

[1] The project is supported by grant RES-328-25-0009 under the ESRC/EPSRC/DTI "PACCIT" programme, and its metaphor aspects also by EPSRC grant EP/C538943/1. We thank our partners—Hi8us Midlands Ltd, Maverick Television Ltd and BT—and colleagues W.H. Edmondson, S.R. Glasbey, M.G. Lee and Z. Wen.

for example the Mayid actor might play him as repenting of his bullying.

The human director has a number of roles. S/he must constantly monitor the unfolding drama and the actors' interactions, or lack of them, to intervene if they are not keeping to the general spirit of the scenario. For example, a director may intervene when the emotions expressed or discussed are not as expected (or are not leading consistently in a new interesting direction). The director may also intervene if, for example, one character is not getting involved, or is dominating the improvisation.

Intervention can take a number of forms. The director can send suggestions to actors. However, another important means of intervention is for the director to introduce and control a 'bit-part' character. This character may not have a major role in the drama, but might, for example, try to interact with a character who is not participating much in the drama or who is being ignored by the other characters. Alternatively, it might make comments intended to 'stir up' the emotions of those involved, or, by intervening, diffuse an inappropriate exchange developing between two characters.

Clearly, all this imposes a heavy burden on the director. Playing the role of the bit-part character makes it difficult to fully monitor the improvisation and send appropriate messages to actors. The difficulty is particularly acute if the directors are novices, such as teachers trying to use e-drama within their lessons.

One major research aim is accordingly to automate some directorial functions, either to take some of the burden away from a human director or to provide a fully automated and therefore necessarily very restricted director. With a fully-automated director, even if highly restricted in what it can do, little or no human supervision might be required for at least minimally adequate improvisations and so these directorial functions could make a useful addition to role-playing games.

However, our main current work is on assisting a human director by providing fully-automated control of a bit-part character, although we are also working on automating limited types of director-to-actor message-sending to allow the human director to concentrate on the more difficult aspects of the task. For this reason, we have created a simple automated actor, EmEliza, which is under the control of an affect-detection module. The module tries to identify affect in characters' speeches, allowing the EmEliza character to make responses that it is hoped will stimulate the improvisation. Within affect we include: basic and complex *emotions* such as anger and embarrassment respectively; *meta-emotions* such as desiring to overcome anxiety; *moods* such as hostility; and *value judgments* of goodness, importance, etc.

The name EmEliza, while appropriate at an initial stage of its development because of the program's Eliza-like qualities, is now outmoded as we have begun to include more sophisticated syntactic and semantic processing, as sketched below.

Although merely detecting affect is limited compared to extracting the full meaning of characters' utterances, we have found that in many cases this is sufficient for the purposes of stimulating the improvisation. Also, even limited types of affect detection can be useful. We do not purport to be able to make EmEliza detect all types of affect under all ways affect can be expressed or implied, or to do it with a high degree of reliability. The spirit of the project is to see how far we can get with practical processing techniques, while at the same time investigating theoretically the nature of, and potential computational ways of dealing with, forms of affective expression that are too difficult to handle in a usable implemented system.

Much research has been done on creating affective virtual characters in interactive systems. Indeed, Picard's work (2000) makes great contributions to building affective virtual characters. Also, emotion theories, particularly that of Ortony, et al. (1988) (OCC), have been used widely therein. Prendinger and Ishizuka (2001) used the OCC model in part to reason about emotions and to produce believable emotional expression. Wiltschko's *eDrama Front Desk* (2003) is designed as an online emotional natural language dialogue simulator with a virtual reception interface for pedagogical purposes. Mehdi et al. (2004) combined the widely accepted five-factor model of personality (McCrae et al., 1992), mood and OCC in their approach for the generation of emotional behaviour for a fireman training application. Gratch and Marsella (2004) presented an integrated model of appraisal and coping, in order to reason about emotions and to provide emotional responses, facial expressions and potential social intelligence for virtual agents. Egges et al. (2003) have provided virtual characters with conversational emotional responsiveness. Elliott et al. (1997) demonstrated tutoring systems that reason about users' emotions. There is much other work in a similar vein.

There has been only a limited amount of work directly comparable to our own, especially given our concentration on improvisation and open-ended language. There has been relevant work on general linguistic clues that could be used in practice for affect detection (e.g. Craggs & Wood (2004)), whilst Façade (Mateas, 2002) included shallow natural language processing for characters' open-ended utterances, but the detection of major emotions, rudeness and value judgements is not mentioned. Zhe and Boucouvalas (2002) demonstrated an emotion extraction module embedded in an Internet chatting environment. It uses a part-of-speech tagger and a syntactic chunker to detect the emotional words and to analyse emotion intensity for the first person (e.g. 'I' or 'we'). Unfortunately the emotion detection focuses only on emotional adjectives, and does not address deep issues such as figurative expression of emotion. Also, the concentration purely on first-person emotions is narrow.

Our work is distinctive in several respects. Our interest is not just in (a) the first-person, positive expression of affect: the affective states or attitudes that a virtual character X implies that it itself has (or had or will have, etc.), but also in (b) affect that the character X implies it lacks, (c) affect that X implies that other characters have or lack, and (d) questions, commands, injunctions, etc. concerning affect. We aim also for the software to cope partially with the important case of communication of affect via metaphor (Fussell & Moss, 1998; Kövecses, 1998), and to push forward the theoretical study of such language, as part of our research on metaphor generally (see, e.g., Barnden et al. (2004)).

Our project does not involve using or developing deep, scientific models of how emotional states, etc., function in cognition. Instead, the deep questions investigated are on linguistic matters such as the metaphorical expression of affect and how ordinary people understand and talk about affect in ordinary life. What is of prime importance is their *common-sense* views of how affect works, irrespective of how scientifically accurate those views are. Metaphor is strongly involved in such views.

It should also be appreciated that this paper does not address the emotional, etc. states of the *actors* (or director, or any audience). Our focus is on the affect that the actors make their characters express or mention. While an actor may work him/herself up into, or be put into, a state similar to or affected by those in his/her own characters' speeches or those of other characters, such interesting effects, which go to the heart of the dramatic experience, are beyond the scope of this paper, and so is the possibility of using information one might be able to get about actors' own affective states as a hint about the affective states of their characters or vice-versa.

2 Our Current Affect Detection

Various characterizations of emotion are used in emotion theories. The OCC model uses emotion labels (anger, etc.) and intensity, while Watson and Tellegen (1985) use positivity and negativity of affect as the major dimensions. Currently, we use an evaluation dimension (negative-positive), affect labels, and intensity. Affect labels plus intensity are used when strong text clues signalling affect are detected, while the evaluation dimension plus intensity is used for weak text clues. Moreover, our analysis reported here is based on the transcripts of previous e-drama sessions. Since even a person's interpretations of affect can be very unreliable, our approach combines various weak relevant affect indicators into a stronger and more reliable source of information for affect detection. Now we summarize our affect detection based on multiple streams of information.

2.1 Pre-processing Modules

The language in the speeches created in e-drama sessions severely challenges existing language-analysis tools if accurate semantic information is sought even for the purposes of restricted affect-detection. The language includes misspellings, ungrammaticality, abbreviations (often as in text messaging), slang, use of upper case and special punctuation (such as repeated exclamation marks) for affective emphasis, repetition of letters or words also for affective emphasis, and open-ended interjective and onomatopoeic elements such as "hm", "ow" and "grrrr". In the examples we have studied, which so far involve teenage children improvising around topics such as school bullying, the genre is similar to Internet chat.

To deal with the misspellings, abbreviations, letter repetitions, interjections and onomatopoeia, several types of pre-processing occur before actual detection of affect.

A lookup table has been created containing abbreviations for Internet chat rooms and abbreviations that we have found by analyzing previous e-drama sessions (e.g. 'im (I am)' and 'c u (see you)'). Multiword phrases and original words, which translate the abbreviations, are listed correspondingly in the lookup table. The abbreviation module can handle most of the abbreviation cases in users' input. Especially we also deal with abbreviations such as numbers embedded within words (e.g., "l8r" for "later") using the lookup table. Unfortunately certain abbreviations can be ambiguous. E.g., '2' may stand for 'to', 'too' or 'two' (although the last is rare in our genre), as exemplified by "I'm 2 hungry 2 walk". A lookup table on its own cannot solve such context-sensitive ambiguity. In order to solve this problem, part-of-speech information has been assigned to the surrounding words using the lexicon from Brill's tagger (1994). Then, simple context-sensitive strategies are used to find the appropriate words for the ambiguous items. These simple strategies may lead to errors in some special cases, but we have evaluated them using examples from previous e-drama transcripts and obtained an 85.7%

accuracy rate, which is adequate. We are also considering dealing with abbreviations, etc. in a more general way by including them as special lexical items in the lexicon of the robust parser we are using (see below).

Letter repetition comes in two flavours. One is repetition added to ordinary words (e.g. ‘yessss’, ‘seeeee’) and the other is repetition that expands interjections or onomatopoeic elements (e.g. ‘grrrrrrrr’, ‘agggghhhhh’). The iconic use of word length here (i.e., written word length corresponding roughly to imagined sound length) normally implies strong affective states in the characters’ input. Usefully, adding letters does not change the pronunciation a great deal. We therefore handle added letter repetitions by means of the Metaphone spelling-correction algorithm (2005), whose working strategy is based on pronunciations, together with a small dictionary that we created, containing base forms of various interjections and onomatopoeic elements together with some ordinary words that are often subject to letter-repetition in e-drama sessions. We also aim to develop a more general detector of onomatopoeic elements that does not rely on particular base forms. We stress that added letter-repetition is not simply eliminated: its occurrence is recorded, to aid affect-detection.

Finally, the Levenshtein distance algorithm (2005) with a contemporary English dictionary deals with spelling mistakes in users’ input.

Having described the necessary preprocessing, we now turn to the core detection of affect in users’ input. In an initial stage of our work, affect detection was based purely on textual pattern-matching rules that looked for simple grammatical patterns or templates partially involving lists of specific alternative words. This continues to be a core aspect of our system but we have now added robust parsing and some semantic analysis. First we describe the pattern-matching.

2.2 Pattern Matching

In the textual pattern-matching, particular keywords, phrases and fragmented sentences are found, but also certain partial sentence structures are extracted. This procedure possesses the robustness and flexibility to accept many ungrammatical fragmented sentences and to deal with the varied positions of sought-after phraseology in speeches. However, it lacks other types of generality and can be fooled when the phrases are suitably embedded as subcomponents of other grammatical structures. For example, if the input is “I doubt she’s really angry”, rules looking for anger in a simple way will fail to provide the expected results. Below we indicate our path beyond these limitations.

The transcripts analysed to inspire our initial knowledge base and pattern-matching rules were produced from earlier *edrama* improvisations based on a school bullying scenario, using school children aged from 8 to 12. We have also worked on another, distinctly different scenario – Crohn’s disease, based on a programme from one of our industrial partners, Maverick Television Ltd. One interesting feature in this scenario is meta-emotion (emotion about emotion) and cognition about emotion, because of the need for people to cope with emotions about their illnesses. The rule sets created for one scenario have a useful degree of applicability to other scenarios, though there will be a few changes in the related knowledge database according to EmEliza’s different roles in specific scenarios.

A rule-based Java framework called Jess (2004) is used to implement the pattern/template-matching rules in EmEliza. When Mayid says “Lisa, you Pizza Face! You smell”, EmEliza detects that he is insulting Lisa. Patterns such as ‘you smell’ have been used for rule implementation. The rules conjecture the character’s emotions, evaluation dimension (negative or positive), politeness (rude or polite) and what response EmEliza should make.

Multiple exclamation marks, capitalisation of whole words and added letter repetition (Werry, 1996) are frequently employed to express affective emphasis in e-drama sessions. If emotion and exclamation marks or capitalisation are detected in a character’s utterance, then the emotion intensity is deemed to be comparatively high (and emotion is suggested even in the absence of other indicators).

A reasonably good indicator that an inner state is being described is the use of ‘I’ (see also Craggs & Wood (2004)), especially in combination with the present or future tense. In the school-bullying scenario, when ‘I’ is followed by a future-tense verb the affective state ‘threatening’ is normally being expressed; and the utterance is usually the shortened version of an implied conditional, e.g., “I’ll scream [if you stay here]”. When ‘I’ is followed by a present-tense verb, other emotional states tend to be expressed, e.g. “I want my mum” (fear) and “I hate you” (dislike). Further analysis of first-person, present-tense cases is provided in section 2.4.

2.3 Processing of Imperatives

One useful pointer to affect, particularly to strong emotions and/or rude attitudes, is the use of imperative mood, especially when used without softeners such as ‘please’ or ‘would you’. There are special, common imperative phrases we deal with explicitly, such as “shut up” and “mind your own business”. They usually indicate strong negative emotions. But the phenomenon is more general.

Detecting imperatives accurately in general is by itself an example of the non-trivial problems we face. To go beyond the limitations of the text matching we have done, we have also used syntactic outputs from the Rasp parser (Briscoe & Carroll, 2002) and semantic information in the form of the semantic profiles for the 1,000 most frequently used English words (Heise, 1965) to deal with certain types of imperatives. This helps us to deal with at least some of the difficulties.

The Rasp parser recognises some types of imperatives directly. Unfortunately, the grammar of the 2002 version of the Rasp parser that we have used does not deal properly with certain imperatives (John Carroll, p.c), which means that examples like "you shut up", "Dave bring me the menu", "Matt don't be so blunt" and "please leave me alone", are not recognized as imperatives, but as normal declarative sentences. Therefore, further analysis is needed to detect imperatives, by additional processing applied to the possibly-incorrect syntactic trees produced by Rasp.

If Rasp outputs a subject, 'you', followed by certain verbs (e.g. 'shut', 'calm', etc) or certain verb phrases (e.g. 'get lost', 'go away' etc), the sentence type will be changed to imperative. (Note: in "you get out" the "you" could be a vocative rather than the subject of "get", especially as punctuation such as commas is often omitted in our genre; however these cases are not worth distinguishing and we assume that the "you" is a subject.) If a softener 'please' is followed by the base forms of a verb, then the input is taken to be imperative. If a singular proper noun is followed by a base form of the verb, then this sentence is taken to be an imperative as well (e.g. "Dave get lost"). However, when a subject is followed by a verb for which there is no difference at all between the base form and the past tense form, then ambiguity arises between imperative and declarative (e.g. "Lisa hit me").

An important special case of this ambiguity is when the object of the verb is 'me'. To solve the ambiguity, we have adopted the evaluation value of the verb from Heise's compilation of semantic differential profiles (1965). In these profiles, Heise listed values of evaluation, activation, potency, distance from neutrality, etc. for the 1,000 most frequently used English words. In the evaluation dimension, positive values imply goodness. Because normally people tend to use 'a negative verb + me' to complain about an unfair fact, if the evaluation value is negative for such a verb, then this sentence is probably not imperative but declarative (e.g "Mayid hurt me"). Otherwise, other factors implying imperative are checked in this sentence, such as exclamation marks and capitalizations. If these factors occur, then the input is probably an imperative. Otherwise, the conversation logs are checked to see if there is any question sentence directed toward this speaker recently. If there is, then the input is conjectured to be declarative.

There is another type of sentence: 'don't you + base form of verb' that we have started to address. Though such sentences are often interrogative, they can be negative versions of imperatives with a 'you' subject (e.g. "Don't you call me a dog"). Normally Rasp regards them as interrogatives. Thus, further analysis has been implemented for such a sentence structure to change the sentence type to imperative. Although currently this has limited effect, as we only infer a (negative) affective quality when the verb is "dare", we plan to add semantic processing in an attempt to glean affect more generally from "Don't you …" imperatives.

In general, the imperative-mood detection is one useful tool for extracting potential affective flavour from users' input.

Aside from imperatives, we have also worked on implementing simple types of semantic extraction of affect using affect dictionaries and electronic thesauri, such as WordNet (2005). The way we are currently using WordNet is briefly as follows.

2.4 Using WordNet

As we mentioned earlier, use of the first-person with a present-tense verb tends to express an affective state in the speaker. We have used the Rasp parser to detect such a sentence. First of all, such user's input is sent to the pattern-matching rules in order to obtain the speaker's current affective state and EmEliza's response to the user. If there is no rule fired (i.e. we don't obtain any information of the speaker's affective state and EmEliza's response from the pattern-matching rules), the subsequent further processing is applied. Then we use WordNet to track down the synonyms of the verb (possibly from different synsets) in the verb phrase of the input sentence, in order to allow a higher degree of generality than would be achieved just with the use of our pattern-matching rules. In order to find the closest synonyms to the verb in different synsets, Heise's (1965) semantic profiles of the 1,000 most frequently used English words have again been employed, especially to find the evaluation values of every synonym in different synsets and the original verb. We currently transform the graded positive and negative evaluation values in Heise's dictionary into binary 'positive' and 'negative' only. Then if any synonym has the same evaluation ('positive' or 'negative') as that of the original verb, then it will be selected as a member of the set of closest synonyms. Then, we use one closest synonym to replace the original verb in the user's input. This newly built sentence will be sent to the pattern-matching rules in order to obtain the user's affective state and EmEliza's response. Such processing (using a closest synonym to replace the original verb and sending the newly built sentence to the pattern-matching rules) continues until we obtain the speaker's affective state and appropriate response.

2.5 Responding Regimes

EmEliza normally responds to, on average, every Nth speech by another character in the e-drama session, where N is a changeable parameter (currently set to 3). However, it also responds when EmEliza's character's name is mentioned, and makes no response if it cannot detect anything useful in the utterance it is responding to. The one-in-N average is achieved by sampling a random variable every time another character says something. As a future development we plan to have N dynamically adjustable according to how confident EmEliza is about what it has discerned in the utterance at hand.

EmEliza sometimes makes a random response from several stored response candidates that are suitable for the affective quality it has discerned in an utterance it is responding to. In addition, EmEliza sometimes reflects back, in modified form, part of the user's input string as a sub-component of its response. Notice here that, because the pre-processing module reported in section 2.1 expands abbreviations and corrects misspellings, it helps to obscure the fact that part of EmEliza's response is only a reflection. For example:

<Another character> I'm ur (*your*) best m8 (*mate*).
<EmEliza character> Why are you my best mate?

On the other hand, we also, on a random basis, purposely put phenomena such as abbreviations and Internet-chat features in EmEliza's responses in order to make EmEliza express itself more like a typical school-student user. For example:

<Another character > You get lost!
<EmEliza character > Hey, coool it folks. (notice the lengthening of 'cool')
<Another character > He didn't do it.
<EmEliza character > No1 believes you, m8. (where 'no1' stands for 'no one')

It should be borne in mind that EmEliza's responses are not aimed at engaging with the detailed meaning of the utterance, but simply to stimulate the improvisation in a way that is somewhat sensitive to affect being expressed. Furthermore, in order to make the EmEliza character's responses push the improvisation forward, the character will not only ask scenario related questions to the main characters, but also introduce new topics closely related to the scenario in the improvisation. In a recent user-testing debrief session, secondary school students mentioned that the human bit-part character did not stay in character and said pointless things, while in another session one student, who played a main character, believed that the EmEliza character was the only one that stuck to scenario related topics. The directors reported that, even when a main character was silent and the director did not intervene very much, the EmEliza character led the improvisation on the right track by raising new topics other characters were concerned about.

3 Affect via Metaphor

The direct metaphorical description of emotional states is common and has been extensively studied (Fussell & Moss, 1998). Examples are "He nearly exploded" and "Joy ran through me". We say that such descriptions are "direct" because they are directly about emotional states, even though in many cases no emotional state is named. But affect is often conveyed more indirectly via metaphor, as in "His room is a cess-pit", where affect associated with a source item (cess-pit) gets carried over to the corresponding target item (the room).

In our research on metaphor (see, e.g., Barnden et al., 2004; Barnden, forthcoming) we are concerned with metaphor in general and are in particular interested in both of the types of affective metaphor in the previous paragraph. We are bringing this metaphor research to bear upon the e-drama application, and using the application as a useful source of theoretical inspiration.

Our intended approach to metaphor handling in the EmEliza affect-detection module is partly to look for stock metaphorical phraseology and straightforward variants of it, and partly to use a simple version of the more open-ended, reasoning-based techniques taken from the ATT-Meta project on metaphor processing (Barnden et al., 2004). As an example of stock phrase handling, insults in e-drama are often metaphorical, especially the case of animal insults ("you stupid cow", "you dog"). Simple pattern-matching rules are currently used in EmEliza to deal with such insulting language. However, we aim to back this method up by general processing of animal words when an animal not specifically dealt with by a rule is mentioned (with apparent metaphorical intent).

In most discourse, metaphorical phraseology tends to be of conventional form, the extreme being stock phrases such as "sit on the fence". Such phrases can be stored in a lexicon and directly recognized. However, it is quite common also to get variations of stock phraseology, of a nature that defeats any purely lexicon-based approach and raises the need for a certain amount of knowledge-based reasoning. For example, a Google™ search found the following variant of the "on the fence" stock phrase: "It will put them on a shaky fence between protecting their copyrights and technology terrorists". Such cases would benefit from the reasoning capabilities of ATT-Meta. The phenomenon is discussed in more detail in (Barnden, forthcoming). Similarly, some edrama transcript examples of metaphor have gone in an open-ended way beyond conventional wording even though based on familiar metaphorical conceptions. Such examples include (after conversion of textese, etc.): "you're looking in the mirror right now, but you probably can't see yourself with all the cracks".

We have encountered further creative uses of metaphor in e-drama. In the school-bullying scenario, Mayid has already insulted Lisa by calling her a 'pizza' (short for 'pizza-face'). In one improvisation Mayid said "I'll knock your topping off, Lisa" – a theoretically intriguing spontaneous creative elaboration of the 'pizza' metaphor.

One more standard phenomenon of theoretical and practical interest is that physical size is often metaphorically used to emphasize evaluations, as in "you are a big bully", "you're a big idiot", and "you're just a little bully". The bigness is sometimes literal as well. "Big bully" expresses strong disapproval (Sharoff, 2005) and "little bully" can express contempt, although "little" can also convey sympathy. Such examples are not only important in practice but also theoretically challenging.

4 User Testing

We conducted a two-day pilot user test with 39 secondary school students in May 2005, in order to try out and a refine a testing methodology. We concealed the fact that EmEliza was involved in some sessions in order to have a fair test of the difference that is made to users' (actors') overall experience by having one of the bit-parts played by EmEliza as opposed to a human actor. We obtained surprisingly good results judging by statistical analysis of questionnaire results and by feedback from the school students in the group debriefing sessions after the e-drama sessions. The inclusion or otherwise of EmEliza made no statistically significant difference to measures of user engagement and enjoyment, or indeed to user perceptions of the worth of the contributions made by the character "Dave" sometimes played by EmEliza, even though users did comment on some utterances of Dave's (so it was not that there was a lack of effect simply because users did not notice Dave at all). In fact, no user appeared to realize that sometimes Dave was computer-controlled though one user was getting close. We stress, however, that it is *not* and aim of our work to ensure that human actors do not realize that a character is being computer-controlled. Further, more extensive, user testing at several Birmingham secondary schools is being conducted at the time of writing this paper.

5 Conclusion and Ongoing Work

We have implemented a limited degree of affect-detection in an automated bit-part character in an e-drama application, and fielded the character successfully in pilot user-testing. Although there is a considerable distance to go in terms of the practical affect-detection that we plan to implement, the already implemented detection is able to cause reasonably appropriate contributions by the automated character.

We also intend to use the affect-detection in a module for automatically generating director messages to actors. We envisage the human director specifying certain types and levels of emotion, etc. that particular characters are expected to display, or having these specified as part of the scenario, so that the affect-detection module can look out for them. A difficulty is allowing for creative improvisation that introduces an unexpected but nevertheless valuable affective profile. We can expect to be able to do only a limited amount about this in automation, but at the very least the human director could dynamically adjust the expected affect parameters if an unexpected but desirable profile arises. Aside from affect-sensitive directorial messages, we are in the process of implementing simple facilities for automatically generating director messages when a particular character is not participating for long periods or appears to be hogging the stage.

In general, our work contributes to the issue of what types of automation should be included in the interactive narrative environments and how affect detection from language can contribute to the development of believable synthetic AI characters, which may contribute to a user's feeling of involvement in game playing. Our project also helps young people to cope with difficult issues that normally can't be presented in the classroom. Moreover, the development of affect detection provides a good test-bed for the accompanying deeper research into how affect is conveyed linguistically.

References

Barnden, J.A., Glasbey, S.R., Lee, M.G. & Wallington, A.M. 2004. Varieties and Directions of Inter-domain Influence in Metaphor. *Metaphor and Symbol*, 19(1), pp.1-30.

Barnden, J.A. Forthcoming. Metaphor, Semantic Preferences and Context-sensitivity. Invited chapter for a Festschrifft volume. Kluwer.

Brill, E. 1994. Some Advances In Rule-Based Part of Speech Tagging. In *Proceedings of AAAI*.

Briscoe, E. and J. Carroll. 2002. Robust Accurate Statistical Annotation of General Text. In *Proceedings of the 3rd International Conference on Language Resources and Evaluation*, Las Palmas, Gran Canaria. pp.1499-1504.

Craggs, R. & Wood. M. 2004. A Two Dimensional Annotation Scheme for Emotion in Dialogue. In *Proceedings of AAAI Spring Symposium: Exploring Attitude and Affect in Text*.

Egges, A., Kshirsagar, S. & Magnenat-Thalmann, N. 2003. A Model for Personality and Emotion Simulation, In *Proceedings of Knowledge-Based Intelligent Information & Engineering Systems (KES2003)*, Lecture Notes in AI. Springer-Verlag: Berlin.

Elliott, C., Rickel, J. & Lester, J. 1997. Integrating Affective Computing into Animated Tutoring Agents. In *Proceedings of IJCAI'97 Workshop on Intelligent Interface Agents*.

Fussell, S. & Moss, M. 1998. Figurative Language in Descriptions of Emotional States. In S. R. Fussell and R. J. Kreuz (Eds.), *Social and Cognitive Approaches to Interpersonal Communication*. Lawrence Erlbaum.

Gratch, J. & Marsella, S. 2004. A Domain-Independent Framework for Modeling Emotion. *Journal of Cognitive Systems Research*. Vol 5, Issue 4, pp.269-306.

Heise, D. R. 1965. Semantic Differential Profiles for 1,000 Most Frequent English Words. *Psychological Monographs*. 79, pp.1-31.

Jess, 2004. The Rule Engine for Java Platform. http://herzberg.ca.sandia.gov/jess/.

Kövecses, Z. 1998. Are There Any Emotion-Specific Metaphors? In *Speaking of Emotions: Conceptualization and Expression*. Athanasiadou, A. and Tabakowska, E. (eds.), Berlin and New York: Mouton de Gruyter, pp.127-151.

Levenshtein Distance Algorithm. 2005. http://www.merriampark.com/ld.htm.

Machado, I., Prada, R. and Paiva, A. 2000. Bringing Drama into a Virtual Stage, In *Proceedings of ACM Conference on Collaborative Virtual Environments*, ACM Press.

Mateas, M. 2002. Ph.D. Thesis. Interactive Drama, Art and Artificial Intelligence. School of Computer Science, Carnegie Mellon University.

McCrae, R.R. and John, O.P. 1992. An Introduction to the Five Factor Model and Its Application. *Journal of Personality*, 60, pp.175-215.

Mehdi, E. J., Nico P., Julie D. & Bernard P. 2004. Modeling Character Emotion in an Interactive Virtual Environment. In *Proceedings of AISB 2004 Symposium: Motion, Emotion and Cognition*. Leeds, UK.

Metaphone Algorithm. 2005. http://aspell.net/metaphone/.

Ortony, A., Clore, G.L. & Collins, A. 1988. *The Cognitive Structure of Emotions*. Cambridge U. Press.

Picard, R.W. 2000. *Affective Computing*. The MIT Press. Cambridge MA.

Prendinger, H. & Ishizuka, M. 2001. Simulating Affective Communication with Animated Agents. In *Proceedings of Eighth IFIP TC.13 Conference on Human-Computer Interaction*, Tokyo, Japan, pp.182-189.

Sharoff, S. 2005. How to Handle Lexical Semantics in SFL: a Corpus Study of Purposes for Using Size Adjectives. *Systemic Linguistics and Corpus*. London: Continuum.

Watson, D. and Tellegen, A. 1985. Toward a Consensual Structure of Mood. *Psychological Bulletin*, 98, pp.219-235.

Werry, C. 1996. Linguistic and Interactional Features of Internet Relay Chat. In *Computer-Mediated Communication: Linguistic: Social and Cross-Cultural Perspectives*. Pragmatics and Beyond New Series 39. Amsterdam: John Benjamins, pp.47-64.

Wiltschko, W. R. 2003. Emotion Dialogue Simulator. eDrama learning, Inc. eDrama Front Desk.

WordNet, A Lexical Database for the English Language. 2005. Version 2.1 Cognitive Science Laboratory. Princeton University.

Zhe, X. & Boucouvalas, A. C. 2002. Text-to-Emotion Engine for Real Time Internet Communication. In *Proceedings of International Symposium on Communication Systems, Networks and DSPs*, Staffordshire University, UK, pp.164-168.

Network Analysis in Natural Sciences and Engineering

5th – 6th April 2006

Organisers

Susanne Hoche, University of Bristol
Jane Memmott, University of Bristol
Nick Monk, University of Sheffield
Andreas Nürnberger, University of Magdeburg

Programme Committee

Contents

Network Analysis and Dynamic Conflict

Ulrik Brandes[*]

[*]Department of Computer and Information Science, University of Konstanz
Box D 67, 78457 Konstanz, Germany
`Ulrik.Brandes@uni-konstanz.de`

Abstract

After a general introduction to the emerging field of network analysis, we will focus on the analysis of group structure in dynamic networks. By defining groups based on similarity of neighborhoods rather than local density, nodes can be associated to roles representing structural positions in a network. A recently proposed relaxation of discrete role assignment allows for varying degrees of membership in such roles, and also points to dominant representatives within roles. The utility of this new approach is demonstrated on dynamic event data extracted from news reports on conflicts that took place in the Persian Gulf and on the Balkans.

Adaptation in Biological Networks: From the genome to ecosystems

Barbara Drossel[⋆]
[⋆]Institute of Solid State Physics, Technical University of Darmstadt
Hochschulstr. 6, D-64289 Darmstadt
barbara.drossel@physik.tu−darmstadt.de

Abstract

Networks are ubiquitous in biological systems. Two widely studied examples are regulatory genetic networks and foodwebs. Both types of networks show a high degree of adaptation: Genetic networks perform reliably, even though the individual elements are affected by noise. Foodwebs retain their complex structure in spite of highly nonlinear population dynamics and changes in the foodweb composition. This talk will discuss both types of systems from the perspective of a theoretician. After a general introduction, models for both systems will be presented that capture the features that are essential for tackling the question of robustness and stability. By discussing the structure and dynamics of these networks, features that convey stability are identified. Among these are stabilizing topological elements in genetic networks and adaptive foraging behavior in food webs.

Fault tolerance and network integrity measures: the case of computer-based systems

Peter Andras*
*School of Computing Science
University of Newcastle
peter.andras@ncl.ac.uk

Olusola Idowu*
*School of Computing Science
University of Newcastle
o.c.idowu@ncl.ac.uk

Panayiotis Periorellis*
*School of Computing Science
University of Newcastle
panayiotis.periorellis@ncl.ac.uk

Abstract

Fault tolerance is a key aspect of the dependability of complex computer-based systems. Fault tolerance may be difficult to measure directly in complex real world systems, and we propose here to measure it in terms of integrity preservation of the system under the assumption of a particular fault occurrence distribution. We measure the integrity preservation ability of the system by measuring the change of structural integrity of the graph representing the system while it is exposed to random node removal according to the assumed fault distribution. We show how to use such measures to measure the integrity reservation of computer-based systems and in this way indirectly their fault tolerance. We discuss the application of the proposed method in the context of a real world example, the Linux operating system. The results indicate that integrity preservation metrics can serve as an appropriate measure of fault tolerance of complex computer-based systems.

1 Introduction

The concept of fault tolerance of complex computer-based systems, and in particular of computers and software, emerged very early in the 1950s (Lee and Anderson, 1990). It was recognized that unexpected faults may emerge in computer-based systems, and that effective dealing with such faults it is critical for highly dependable systems. Fault tolerance is a key measure of the dependability of computer-based systems (Lee and Anderson, 1990; Laprie, 1992), dependability being defined as reliability, availability, safety, security, survivability, and maintainability of a system (Aviziensis et al, 2001).

Generally systems can be perceived as a set of units that are interconnected by their actions and behaviours (von Bertalanffy, 1973). Computer-based systems can be seen as systems with units which can be computer hardware, software, humans, and possibly a variety of other machines and human artefacts containing sensors and actuators. The interconnecting actions and behaviours of these units can take the form of data entry to the computers, data communications between hardware components, data interchange and processing by software components, and display or communication of data to actuators.

An interesting issue is how to measure the fault tolerance of a computer-based system. Systematic mathematical analysis of fault tolerance of models of computer-based systems started in 1960s (Lee and Anderson, 1990). Typically fault tolerance is evaluated by full probabilistic analysis of the system, by calculating measures such as mean time to failure and mean time to repair under the assumption of a fault occurrence scenario (e.g., identical and independent fault occurrence distribution for each system component) (Amari, 2000; Chang et al, 2004; Ou and Dugan, 2003; Scerrer and Steininger, 2003).

One stream of fault tolerance research is focused on the analysis of graphs that represent computer-based systems (Billinton and Jonnavithula, 1999; Bell, 2003; Cheng and He, 2004). These works assume a fault occurrence scenario in the graph (e.g., node failure or edge failure) and measure the probability of connectedness (Beichelt and Tittmane, 1991; Elperin et al, 1991) or of having flow capacity above a given limit (Chan et al, 1997; Kishimoto, 1997) of the graph as a proxy measure for the fault tolerance of the system represented by the graph. The main drawback of these methods is that they are very computationally intensive and in many cases they are restricted to a narrow range of particular graph topologies (Al-Sadi et al, 2002; Goerdt, 2001; Goerdt and Molloy, 2003).

An alternative way to analyse the robustness of systems is to use structural graph analysis methods that reveal vulnerable components and the sensitivity to structural damage of the system (Albert et al, 1999). These methods assess the integrity of the system and the change of integrity measures after structural damage to the system in terms of structural measures, such as diameter, average minimum path length or average clustering coefficient. The underlying theoretical assumption is that system structural integrity implies functional integrity of the system (Andrews and Beeson, 2003; Ferrandi et al, 2003). This is supported by practical examples, which show that structural integrity and functional integrity of systems are strongly correlated (Albert et al, 1999; Jeong et al, 2001). Consequently, the analysis of the structural integrity of the graph representing a system by appropriate structural measures can provide indicator measures of the functional integrity of the system.

We propose in this paper the use of structural graph analysis methods to measure the integrity of computer-based systems. We measure the likely structural damage as an approximation of likely functional damage due to the presence of faults. In this way we can assess the fault tolerance of the system by measuring the likely change of structural integrity of the system.

The rest of the paper is structured as follows. Section 2 discusses system integrity measures. In Section 3 we analyse the link between fault tolerance and integrity measures. Section 4 presents an example of the application of the proposed methodology to the assessment of fault tolerance of computer-based systems. Finally, in section 5 we draw some conclusions of the paper.

2 System integrity

Systems are sets of component units interconnected by their interactions (von Bertalanffy, 1973). Component units interact by their behaviour modifying the state of the units participating in such interactions. In a stronger sense we may consider systems as only those sets of interacting component units, in which the interactions between components depend primarily on earlier interactions between system components (Andras and Charlton, 2005). We should also point out that system components may also interact with other units, which are not part of the system. Such interactions constitute the system's interaction with its environment.

The integrity of a system can be defined in functional terms as the system's ability to perform the full range of system behaviours (Ferrandi et al, 2003). The system behaviours are possible patterns of behaviours of its component units (Lee and Anderson, 1990). Some of these behaviours may have an effect on the system's environment, while others may cause only a change of the internal behaviour of the system.

Measuring functional integrity directly may be difficult, as the full range of possible system behaviour may not be known (Ferrandi et al, 2003). A way to approximate the functional integrity of a system is to measure its structural integrity ((Andrews and Beeson, 2003). In practical cases of living cells (Jeong et al, 2001), nervous systems of animals (Scannell et al, 1995), and technological systems (Albert et al, 1999) it has been shown that their functional integrity correlates strongly with their structural integrity. Measuring structural integrity is much simpler than measuring functional integrity, in the sense that it requires only the measurement of the existence of components and interactions between components, disregarding the actual functional semantics of interactions and interaction patterns.

Structural integrity measures of systems are based on the measurement of the structural integrity of the underlying graph structure of the system, which is made of nodes representing system units, and edges or arcs representing undirected or directed interactions between system units. (We consider undirected graphs only in what follows).

Simple measures of structural integrity of graphs include the diameter, the average minimum path length and the average clustering coefficient of the graph. The diameter is defined as the largest of the minimal path lengths between nodes of the graph:

$$D(G) = \max\{l(i,j) \mid i,j \in V\}, G = \{V,E\} \quad (1)$$

where V is the set of nodes and E is the set of edges of the graph, and l(i,j) is the minimal length of a path between the nodes i and j. The average minimum path length is defined as the average length of minimal paths between all pairs of nodes of the graph that can be connected (i.e. infinite length shortest paths are ignored):

$$\mu = \frac{1}{|V|^2} \sum_{\substack{i,j \in V \\ l(i,j) < \infty}} l(i,j) \quad (2)$$

where we use the same notations as above. The clustering coefficient of a node is the proportion between the number of existing edges between the neighbours of the node and the number of all possible edges:

$$c(i) = \frac{2 \cdot |\{(i,j) \mid (i,j) \in E\}|}{|\{j \mid (i,j) \in E\}| \cdot (|\{j \mid (i,j) \in E\}| - 1)}, i \in V \quad (3)$$

The average clustering coefficient of the graph is the algebraic average of the clustering coefficients of all nodes:

$$\eta(G) = \frac{1}{|V|} \sum_{i \in V} c(i) \quad (4)$$

We note that the above measures evaluate somewhat different aspects of the graph integrity; none of them provides a comprehensive evaluation of the graph integrity. In order to be on the safe side in practical applications the best practice is to use such a set of simple integrity measures and evaluate the graph integrity using the resulting set of integrity measure values (i.e., by considering a vector of integrity measure values). In particular, if we need a single value measure of the graph integrity on the basis of a vector of integrity measures, the safest is to take the value indicating the greatest amount of integrity loss.

Other more sophisticated measures of graph integrity include the calculation of coefficients of the graph’s characteristic polynomial, and eigenvalues of the graph’s adjacency matrix. These methods can provide a full picture of the graph's integrity and in principle capture all its aspects. The disadvantage of these methods is that they are computationally very expensive, and the calculation of the required numbers may be impractical for very large graphs representing complex systems. The above introduced simple integrity measures are well correlated with the more general measures. The largest eigenvalue of the adjacency matrix is related to the density of the edges, the second eigenvalue is related to the conductivity within the network (Farkas et al, 2001). The second coefficient of the characteristic polynomial is related to the number of edges, while the third coefficient is twice the number of triangles in the network (Biggs, 1994).

An important issue regarding the use of graph integrity measures to assess the integrity of systems is that of how to actually measure the system components and their interactions. One approach can be to consider the design of the system, if this is available. (For technological systems this might often be the case.) However, this approach can lead one to fall into the trap of showing the robustness of the designed system and not of the actual system. We believe that the right approach is to measure the existing components of the real system and their existing interactions in order to assess the integrity and robustness of the actual system. However, we recognize that in some practical cases such measurements might prove to be difficult (e.g., monitoring of human – computer interactions), limiting the applicability of the structural graph analysis based assessment of system integrity evaluation.

In the case of computer-based systems we typically have a set of non-computer related units (e.g., humans, sensors), a set of hardware units making the computer hardware part of the system, and usually a very large set of software modules, constituting the units of the system. In some cases we ignore the non-computer related units and even the hardware part of the system and we focus our attention exclusively on the system made of software units. The interactions between software units take the form of data transactions between them, which can be measured by appropriate monitoring of the system (Periorellis et al, 2004).

3 Fault tolerance and integrity preservation

Faults are unexpected behaviours of system components. Faults in computer-based systems may have a number of origins; they can be classified as design faults, physical faults and interaction faults (Aviziensis et al, 2001). Faults cause errors in the system, which are deviations from the expected behaviour of the system. Errors in computer-based systems may stay latent, until they are detected, when they cause abnormal behaviour at the interface of the system with its environment (Lee and Anderson, 1990). Errors cause failures of the system, when the system is unable to perform its function correctly (Aviziensis et al, 2001).

Faults in the system may occur at various places. An important feature of faults is their occurrence within the system and their distribution at these places within the system. In many cases we may suppose that the faults may appear at any system unit according to the same occurrence distribution, no unit being more susceptible for producing faulty interactions than others (Amari, 2000). In some cases we may also use the hypothesis that the likelihood of faulty interactions is proportional with the likelihood of the unit being involved in interactions. In other cases the fault distribution may follow some peculiar well defined distribution, such as in the case of faults induced by malicious logic (e.g., attacks by hackers). The types of fault distributions determine the fault occurrence environment of the system.

Fault tolerance is the ability of the system to maintain its functionality in the presence of active faults (Lee and Anderson, 1990). Fault tolerance is typically achieved by error detection, recovery and fault handling (Aviziensis et al, 2001). Fault tolerance of computer-based systems depends on the fault occurrence environment of the system (e.g., in presence of naturally occurring faults the system may prove sufficiently fault tolerant, while in the presence of targeted attack by hackers, it may prove fault sensitive).

In general the measure of fault tolerance of the system is a relative measure, which shows to what extent the system preserves its functionality in a certain fault occurrence environment (Lee and Anderson, 1990). To assess the fault tolerant nature of a computer-based system we need to assess the level of functionality of the system within the considered fault occurrence environment. In other words we need to evaluate the functionality preservation ability of the system. Usually some probabilistic approach is used to evaluate fault tolerance measures such as mean time to failure or mean time to repair (Lee and Anderson, 1990). These methods take into consideration the whole system resulting computationally very intensive analyses in case of large systems (Chang et al, 2004; Billington and Jonnavithula, 1999). To perform such exhaustive evaluations may prove difficult in practice, as monitoring and assessing all aspects of the functionality of the system and performing all the required calculations may be extremely time and resource consuming (Ferrandi et al, 2003). Alternative methods were proposed recently, involving game theoretic approaches (Bell, 2003), formal languages inspired analysis (Phoha et al, 2004) and structural network analysis approaches (Albert et al, 1999).

We adopt the structural network analysis approach, which has the key advantage that it implies a relatively low computational load for the evaluation of large systems. We measure the fault tolerance of the system by evaluating the ability of the system to preserve its integrity. The measure of integrity preservation is calculated by using system integrity measures based on a structural graph analysis of the graph representing the system. As structural integrity is strongly correlated with functional integrity, the structural integrity preservation measure provides a proxy measure of the functional integrity preservation measure of the system. Consequently, we can use the structural integrity measures introduced in the previous section to measure the change of the integrity of the system in a given fault occurrence environment.

To measure the effects of faults on the integrity of the system, we simulate the faults by sampling the fault occurrence distributions and then evaluating the integrity measures of the system in the presence of simulated faults. The presence of faults causes the elimination from the graph of the system of edges between nodes or of nodes of the graph. These changes happen according to the fault occurrence distributions and have the effect that the integrity measures of the system graph are modified. The expected changes in terms of integrity measures may be calculated analytically in the case of small systems or can be evaluated by numerical simulations in the case of large and complex systems. The expected changes associated with a fault occurrence environment characterise the system's integrity preservation ability and are used as an approximate measure of the fault tolerance of the system.

To show how to use the calculated integrity measures to assess the fault tolerant nature of a system we consider below a toy example. Let us consider a software system of 1000 units of which corresponding graph representation is shown in Figure 1. The system's structural integrity measures are the following: (1) diameter: $D(S)=19$; (2) average minimum path length: $\mu(S)=3.58151$; (3) average clustering coefficient: $\eta(S)=0.022702$

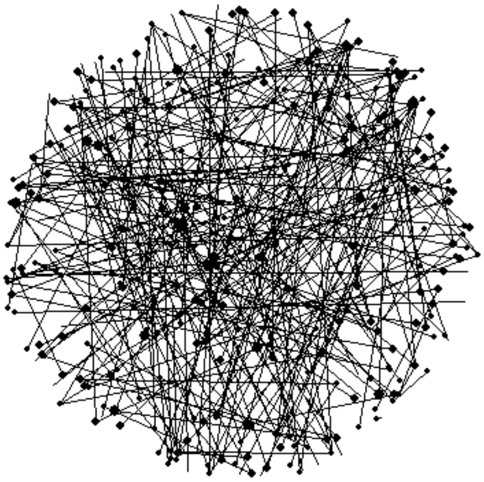


Figure 1: The graph representation of the model system with 1000 nodes. The size of the nodes indicates the number of connections of the node. Only the subset of more connected nodes and the subset of connections between these nodes are displayed to keep the figure comprehensible.

We consider a fault occurrence environment in which the faults occur with equal uniform probability (p=0.15) at each unit of the system and each fault temporarily knocks out the system unit where it occurs. To evaluate the fault tolerance of the system we perform a numerical simulation of the fault occurrences, and evaluate the integrity measures of the system for each simulation. After the simulations we calculate the average values and variances of the system integrity measures. We chose to run 20 simulations in order to get reliable estimates of mean values (the variance of the mean value calculated from n measurements is σ/square-root(n), where σ is the variance of the calculated values). The calculations after the simulations led to the values: (1) diameter: avg(D(S))=23.05, var(D(S))=4.1355; (2) average minimum path length: avg(μ(S))=3.6994, var(μ(S))=0.0397; (3) average clustering coefficient: avg(η(S))=0.023, var(η(S))=0.0014.

To evaluate the integrity preservation ability of the system we calculate first, whether the average values of system integrity measures after the simulation of faults differ significantly or not from the corresponding values calculated for the fully functional system. Next we calculate the normalized distance of the pre-damage and post-damage integrity measure values, which together with their attached statistical significance levels characterize the fault tolerance of the system. In order to be on the safe side, we choose the worst measure (i.e., the largest and most significant distance) to be the numerical evaluation of the fault tolerance of the system. In the case of the above system the normalized distances (z-score, i.e., the distance measured between the mean value and original value in units equal to the standard deviation – $(v_{original} - m)/\sigma_m$) and statistical significance levels (statistical significance levels show how likely is that the original value is the same as the estimated mean value after damage, low p-value indicates that the likelihood of them being the same is very low, or in other words the two values differ significantly) are listed in Table 1.

In the case of the above toy example we have shown how to apply in principle the proposed structural graph analysis based integrity evaluation methods to assess the fault tolerance of a computer-based system. The data shown in the last column of Table 1 shows the values of the likelihoods that original value of the integrity measure is the same as its value after the damage. The results indicate that under the above described fault occurrence environment assumption the system suffers significant damage (p<0.01) in terms of diameter and average shortest path length, the amount of the latter damage being more significant than the former. Considering the most significant damage (i.e., the damage in terms of average path length, p=7.29 x 10^{-14}), we conclude that under the considered fault occurrence assumption the system represented by the graph suffers very significant structural and functional damage, and consequently has low fault tolerance.

Table 1: Summary of integrity measures of the system before and after damage, including the z-score for the original values considering the mean and variance of the after damage values (z-score = (original – damage mean)/(damage variance / square-root(20))), and the statistical significance level of the difference between the original values and the mean values calculated after the damage. The p values above 0.1 are omitted.

Integrity measure	Original	Damage mean	Damage variance	z-score	p-value
Diameter	19	23.05	4.135	4.379	1.2 x 10^{-5}
AvShortPath	3.582	3.699	0.040	13.26	7.29 x 10^{-14}
AvClusCoef	0.023	0.023	0.002	1.079	-

4 Application

Linux is one of the most popular operating systems, which is due to a good extent to its open source based development. It is commonly claimed that Linux is more reliable and secure than many other operating systems. An immediate question is how fault tolerant is Linux actually.

We analysed the network structure of the Linux under typical running conditions with a set of usual programs running. To perform the analysis we considered the calls between the classes present in the Linux kernel (version 2.4.19). We found 6815 classes and 19909 calls between them, by parsing the source code of the classes. The interaction network of the classes (see Figure 2) was then analysed in terms of structural network analysis.

Analysing the connectivity distribution of the processes we found that the distribution follows a power law distribution (with exponent $\gamma = -1.33$; see Figure 3) similar to the case of the Internet (Albert et al, 1999). This indicates that among the Linux classes there are relatively few very highly connected classes (which call and are called by many other classes) and many others with relatively few connections. This implies that similarly to the Internet (Albert et al, 1999) the Linux is very robust and fault tolerant if faults happen randomly following a uniform fault distribution over the processes (run-time representation of classes), while it should be very vulnerable and fault sensitive if faults are distributed such that they affect mostly the most highly-connected processes.

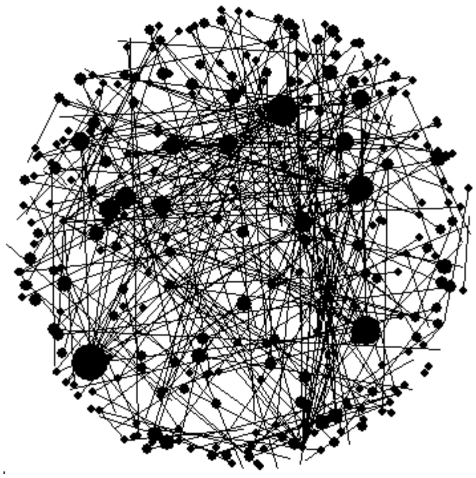


Figure 2: The graph representation of the Linux. The size of the nodes indicates the number of connections of the node. Only the subset of more connected nodes and the subset of connections between these nodes are displayed to keep the figure comprehensible.

We performed an analysis of the Linux class network to evaluate the effects of faults on integrity measures. We simulated a scenario with uniform random distribution with probability of faults at each node. We also performed a simulated a scenario when the likelihood of a node being faulty was proportional with the connectivity of the node. The analysis results are shown in Table 2 and Table 3.

The results show that as we expected Linux is remarkably fault tolerant in a fault occurrence environment characterised by uniform fault distribution, while it is significantly more fault sensitive in the case of a fault distribution centred on the mostly linked processes. This suggests that indeed the common belief about the reliability and fault tolerance of Linux is well founded in case of random uniformly distributed errors, but also highlights that Linux is also a vulnerable system in case of well designed malicious attacks. (Note that the tables show that the average shortest path is decreasing after damage. This is because infinite shortest paths between nodes belonging to isolated sub-networks are ignored.)

Table 2: Summary of integrity measures of Linux before and after random damage, including the z-score for the original values considering the mean and variance of the after damage values (z-score = (original – damage mean)/(damage variance / square-root(20))), and the statistical significance level of the difference between the original values and the mean values calculated after the damage. The p values above 0.1 are omitted.

Integrity measure	Original	Damage mean	Damage variance	z-score	p-value
Diameter	44	38.85	4.869	4.729	2.25 x 10^{-6}
AvShortPath	12.01	11.50	0.740	3.077	0.0021
AvClusCoef	0.133	0.133	0.007	0.350	-

Table 3: Summary of integrity measures of Linux before and after targeted damage, including the z-score for the original values considering the mean and variance of the after damage values (z-score = (original – damage mean)/(damage variance / square-root(20))), and the statistical significance level of the difference between the original values and the mean values calculated after the damage. The p values above 0.1 are omitted.

Integrity measure	Original	Damage mean	Damage variance	z-score	p-value
Diameter	44	35.45	5.062	7.552	1.41 x 10^{-13}

AvShortPath	12.01	10.47	1.444	4.766	1.87 x 10^{-6}
AvClusCoef	0.133	0.132	0.005	1.018	-

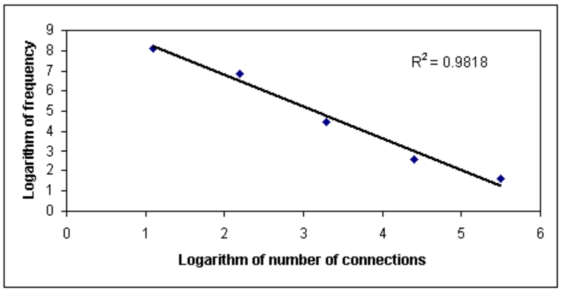


Figure 3: The distribution of connectivities in the case of the Linux classes and calls network

4 Conclusions

Fault tolerance is measurable aspect of the dependability of computer-based systems. Direct measurement of fault tolerance of large real world systems poses considerable problems, considering that most existing work is focused on exhaustive analytical evaluation of relatively simple model systems (Billinton and Jonnavithula, 1999; Bell, 2003; Cheng and He, 2004). An approach to find a proxy measure for the fault tolerance of large systems is to measure their structural integrity preservation under the assumption of a fault occurrence environment. This measure is based on the assumption that functional integrity is strongly correlated with structural integrity (Andrews and Beeson, 2003; Ferrandi et al, 2003), which is supported by experimental analysis of various real world complex systems (Albert et al, 1999; Jeong et al, 2001).

Integrity and integrity preservation of large systems can be measured by structural graph analysis of their graph representation, where nodes represent system units and edges represent interactions between system units. Simpler (e.g., average minimum path length) or more complex (e.g., eigenvalues of the adjacency matrix) measures can be used to estimate the structural integrity preservation of the system while exposed to faults occurring in accordance with an assumed fault distribution. Using these measures we can evaluate the likely amount of loss of integrity (or integrity damage) and the statistical significance of this loss.

A toy example and a real world example were presented to show the application of this fault tolerance measurement approach. These examples show that indeed the proposed methods can be applied effectively and lead to meaningful conclusions about the analysed systems. In the case of the real world example (Linux) the analysis indicates that the system is very fault tolerant under the assumption of uniform fault distribution, but that, not surprisingly, it is very vulnerable under the assumption of a fault distribution driven by properly targeted malicious interventions. However, the methods discussed here enable the extent of this effect to be analyzed in useful detail.

We believe that using relatively simple network integrity measures can simplify considerably the effective analysis of fault tolerance of large real world computer-based systems. Although these methods do not provide an exact measure of fault tolerance they provide good approximations of the actual measure. Such approximate measures can be used to rapidly determine the effects of a variety of fault occurrence environments, allowing the designers and developers of large systems to prepare appropriate defence and repair strategies to support the dependability of their system effectively and efficiently.

Acknowledgement

The authors thank Brian Randell for very helpful comments.

References

A. Aviziensis, J.C. Laprie, B. Randell. Fundamental concepts of dependability. *Newcastle University Report no. CS-TR-739* 2001.

R. Albert, H. Jeong, A.-L. Barabási. Diameter of the World Wide Web. *Nature*, 401:130-131, 1991.

J. Al-Sadi, K. Day, M. Ould-Khaoua. Unsafety vectors: a new fault-tolerant routing for the binary n-cube. *Journal of Systems Architecture*, 47:783-793, 2002.

S.V. Amari. Generic rules to evaluate system-failure frequency. *IEEE Transactions on Reliability*, 49:85-87, 2000.

REPLACE Charlton, B, Andras, P: The nature and function of management - a perspective from systems theory. Philosophy of Management 2004; 3: 3-16.

J.D. Andrews and S. Beeson. Birnbaum's measure of component importance for noncoherent systems. *IEEE Transactions on Reliability*, 52:213-219, 2003.

F. Beichelt and P. Tittmann. A generalized reduction method for the connectedness probability of stochastic networks. *IEEE Transactions on Reliability*, 40:198-204, 1991.

M.G.H. Bell. The use of game theory to measure the vulnerability of stochastic networks. *IEEE Transactions on Reliability*, 52:63-68, 2003.

N. Biggs. *Algebraic Graph Theory*. Cambridge, Cambridge University Press, 1994.

R. Billinton and S. Jonnavithula. Calculation of duration, frequency, and availability indexes in complex networks. *IEEE Transactions on Reliability*, 48:25-30, 1999.

Y. Chan, E. Yim, A. Marsh. Exact and approximate improvement to the throughput of a stochastic network. *IEEE Transactions on Reliability*, 46:473-486, 1997.

Y.-R. Chang, S.V. Amari, S.-Y. Kuo. Computing system failure frequencies and reliability importance measures using OBDD. *IEEE Transactions on Computers*, 53:54-68, 2004.

Y. Cheng and Z. He. Bounds on the reliability of distributed systems with unreliable nodes & links. *IEEE Transactions on Reliability*, 53:205-215, 2004.

T. Elperin, I. Gertsbakh, M. Lomonosov. Estimation of network reliability using graph evolution models. *IEEE Transactions on Reliability*, 40:572-581, 1991.

I.J. Farkas, I. Derenyi, A.-L. Barabasi, T. Vicsek. Spectra of Real-World Graphs: Beyond the Semi-Circle Law. *arXiv*:cond-mat/0102335, 2001.

F. Ferrandi, F. Fummi, G. Pravadelli, D. Sciutto. Identification of design errors through functional testing. *IEEE Transactions on Reliability*, 52:400-412, 2003.

A. Goerdt. Random regular graphs with edge faults: expansion through cores. *Theoretical Computer Science*, 264:91-125, 2001.

A. Goerdt and M. Molloy. Analysis of edge deletion processes on faulty random regular graphs. *Theoretical Computer Science*, 297:241-260, 2003.

H. Jeong, S.P. Mason, A.-L. Barabasi, Z.N. Oltvai. Lethality and centrality in protein networks. *Nature*, 411:41-42, 2001.

W. Kishimoto. Reliable flow with failures in a network . *IEEE Transactions on Reliability*, 46:308-315, 1997.

J.C. Laprie (ed). *Dependability: basic concepts and terminology in English, French, German, Italian and Japanese*. Wien, Springer-Verlag, 1992.

P.A. Lee and T. Anderson. *Fault Tolerance. Principles and Practice*. Wien: Springer-Verlag, 2nd ed, 1990.

Y. Ou and J.B. Dugan. Approximate sensitivity analysis for acyclic Markov reliability models. *IEEE Transactions on Reliability*, 52:220-230, 2003.

P. Periorellis, O.C. Idowu, S.J. Lynden, M.P. Young, P. Andras. Dealing with complex networks of protein interactions: A security measure. In *Proceedings of 9th IEEE International Conference on Engineering of Complex Systems (ICECCS)*, P. Bellini, S.A. Bohner, B. Steffen (eds) pp 29-36, IEEE Computer Society, 2004.

V.V. Phoha, A.U. Nadgar, A. Ray, S. Phoha. Supervisory control of software systems. *IEEE Transactions on Computers*, 53:1187-1199, 2004.

J.W. Scannell, C. Blackmore, M.P. Young. Analysis of connectivity in the cat cerebral-cortex. *Journal of Neuroscience*, 15:1463-1483, 1995.

C. Scherrer and A. Steininger. Dealing with dormant faults in an embedded fault-tolerant computer system. *IEEE Transactions on Reliability*, 52:512-522, 2003.

L. von Bertalanffy. *General System Theory: foundations, development, applications*. Harmondsworth, Penguin, 1973.

Towards Associative Information Access

Michael R. Berthold⋆
⋆Konstanz University
Dept. of Computer and Information Science
Konstanz, Germany
`Michael.Berthold@uni-konstanz.de`

Andreas Nürnberger†
†Otto-von-Guericke Universität Magdeburg
Faculty of Computer Science
Magdeburg, Germany
`nuernb@iws.cs.uni-magdeburg.de`

Abstract

We propose a framework for a unifying, associative access to distributed and heterogenous information resources. The classical index generation is replaced by a process which builds associations between existing information entities and allows for an interactive exploration of information accessible through this structure. Positive ("‘this looks interesting") as well as negative ("‘I know this!") user feedback allows the system to quickly narrow down on interesting pieces of information. The continuous integration of new analysis engines, added sources of information and user feedback allow the formation of a corporate wide memory and expert knowledge repository.

1 Motivation

Large corporations increasingly drown in all sorts of data and other types of information they collect. Modern storage technology essentially sets no limit to the amount of information that can be stored. The huge challenge is the problem of usage — how can users be sure that they did take into account all relevant pieces of information that relate to the current task or problem they are dealing with?

One prime example for this scenario are research departments in many pharmaceutical companies. In order to successfully develop new drugs, many different types of information need to be combined, in the end resulting in a new idea for a medication that has not been patented before, that has no dangerous side effects, or that is not, in some similar form, already being explored elsewhere. Currently this process relies heavily on experts having intuition, long years of experience and hopefully the right insights at the right time. The sources of information these experts rely on are distributed across the entire company (and some also over the entire internet): experimental protocols, patent information, scientific publications, biological information about metabolic pathways just to name a few, and not to forgot, also the colleague down the hall who would have something interesting to say but who our expert did not happen to meet at the coffee pot.

Current approaches try to address this problem by building huge information repositories based on sophisticated database technology. Associative Information Networks, as described here, aim to take an alternate approach – instead of bringing all the information together we propose to build a meta structure that points to the information and helps the user find interesting associations among different pieces of information through means of exploration and context refinement. This meta structure is continuously updated as more sophisticated methods to analyze the information sources arise. In addition, it is possible to naturally incorporate user annotations, capturing expert knowledge and feedback on the way. This process is supported by methods derived from research in the areas of data mining, information retrieval, knowledge management, network and graph theory, data visualization and human computer interaction.

2 Related Work

There has been a lot of work done in the past on the idea of associative information processing, which was in the beginning mainly motivated by the associative information processing capabilities of the human brain (see, e.g., the work of Collins and Loftus (1975)). Thus we can find methods ranging from very general neural network based approaches of Kohonen (1977, 1984), over possibilistic networks or graphs (Borgelt et al., 2000; Cao, 2000) and belonging reasoning methods (Dubois et al., 1994; Gebhardt and Kruse, 1995) to very specific ideas related to document indexing and retrieval, e.g. (Chen, 1995; Chung et al., 1998; Belew, 2000). Furthermore, also ontologies and the Sematic Web (Berners-Lee et al.,

2001) might be considered as an approach to enable linking of semantically associated information.

However, several of the earlier projects failed, since almost all of them are based on the idea that it is possible to know in advance or learn automatically an almost perfect descriptive link from (index-)keywords to documents or in-between documents. This information was then used in some kind of reasoning mechanism to retrieve relevant documents. Unfortunately, in most cases this leads to the retrieval of too few or far too many documents. A further major problem had been the poor visualization methods used.

In order to circumvent these problems, more recently, some projects started in which methods have been studied that are also able to handle more general associative networks by providing interactive visualization methods. In order to navigate and browse complex association networks powerful tools for visualizing relevant subsets for the current exploration (or search) context of the user are required. Recent commercially available approaches that try to tackle this problem are, e.g., the Personal Brain (`http://www.thebrain.com/`), a navigator for indexed data that is however only able to access documents on a local data repository and the iAS KnowledgeSuite (`http://www.knowledgesuite.de/`). The KnowledgeSuite performs a semantic text analysis and creates strong links between previously identified, named entities. In this case, however, association are originate only from primed neurons using positive activity spreading. No interactive refinement or inclusion of uncertain, imprecise information is possible.

In general one might argue that the linking of documents as proposed for the semantic web might solve the problems of linking information sources. However, in the semantic web, one is forced to either link or not link documents, where an existing link has a clear, semantically valid meaning. Even though it is in general possible to introduce mechanisms for context based links (as realized for example in topic maps, see e.g. Biezunski et al. (1999)), no mechanism for storing 'gradual' (e.g. possibilistic, probabilistic, or simply anecdotal or evidential) links between documents are implemented. Furthermore, in the semantic web the whole web is seen as the knowledge base which includes both, links *and* information chunks. In our approach we add a general layer of links over (the possibly already existing link layer within) the considered database of information entities, which could consist of information in the world wide web, a local database or even notes on a local PC. This layer allows to model a personal (or group based) view on the same information, independent of (and not conflicting with) links already present in the data. However, we can easily incorporate general concepts of the (semantic) web, like URIs and existing ontologies in order to model and exploit already available information.

Another aspect that distinguishes our approach from semantic web (or more general logic based) approaches is that we do not use reasoning mechanisms that require a consistent descriptions of relations between information chunks. The main goal of the reasoning mechanism is to detect information that is most likely interesting to the user for any reasons (may be even because its contradicting!). In contrast, the reasoning mechanism itself is able to provide an explanation why some information has been proposed.

One additional differentiator is the ability for continuous learning and updating of the underlying structure. Through integration of new analysis engines, new information sources, or also manual feedback the network continuously refines it's internal structure.

3 Associative Information Networks

3.1 Structure

Associative Information Networks (AI Net in the following) consists of nodes and labelled edges. Each node represents an entity, which can be a concept from the application area (e.g. a disease, or metabolic pathway) or a named entity, such as a gene, a protein, or a specific target. Edges represent links between these entities and are labelled with a reference to the information source(s) and information about the analysis engine that created it from these sources. In addition, each edge holds a weight, modelling the strength of association, and a label indicating the type of the edge. This way, a link can potentially also be derived from an ontology, representing semantic connections between nodes.

3.2 Learning and Refinement

In order to generate the AI Net we need to introduce nodes, and links in between them. Refinement may cause adjustment of links and addition of new nodes. There are two primary ways how both, nodes and links can be added:

- automatic generation: using analysis engines, links between existing nodes can be added or modified. Each analysis engine has a particular purpose and will, for instance, find co-occurrences of words in documents, correlations of genes in gene-expression experiments, structure-activity relationships via the analysis of cell-assay images, or connections between genes and diseases from the analysis of patent information. In comparison, this would resemble the collection and modelling of automatically derivable domain knowledge. Of course, the addition of newly developed analysis engines can continuously update the network.

- manual interaction: throughout usage of the AI Net, the user is able to manually adjust weights of links, mark links as wrong, or insert new links with annotations explaining their purpose. This interactive refinement allows to capture expert knowledge and feedback on the fly and enables the system to model expertise available within a corporation. It is, of course, crucial that this interaction is handled in an intuitive way. The user should not be required to adjust numerical weights or draw links between abstract nodes.

Adding new databases, or more generally, information sources is straightforward – as long as an analysis engine is provided that produces dependencies between entities represented by nodes, new links can easily be added. One further extension of this system would also allow to generate new nodes (and node types) by analyzing external information sources.

3.3 Link Formation: Details

As described above, links can be introduced automatically or through manual refinement. The latter process can be seen as user annotations, incorporating expert knowledge into the network and are therefore mainly an issue of user interface. In the following, we briefly outline, based on a number of examples, how the automatic generation of links and link-weights works.

- semantic links: these are strong links (usually weight $= 1.0$) which are derived from well-known structures, such as ontologies or semantic networks. Those are usually created by an expert. Semantic nets, as extracted (semi-) automatically from data will need to add a component that computes the confidence for each link and convert this to a weight.

- syntactic links: these are links that are generated by a shallow analysis of data. The most prominent example would be a text parser that converts words to stems, eliminates fill words and then produces a set of bi- or trigrams. The corresponding nodes in the AI Net will be connected by weak links. For an example of the corresponding weight computation, see below.

- anecdotal evidence: These are links set by a user, creating links for hypotheses generated by a user (or based on hear-say). Weights of such links are generally low. These links are in contrast to expert-based annotations that generally have very high weights.

- data driven links: These types of links will constitute the vast majority of network weights in most cases. They are generated automatically from data repositories. A few example (here for the context of a pharmaceutical AI Net) could be:

  Gene correlations derived from gene expression data. Links are introduced when, for example, a specific threshold θ for co-occurrence in experimental data is surpassed. The link's weight reflects the correlation strength and for more than two-dimensional correlations the corresponding multi-edges are introduced. In addition each of these links will carry an annotation pointing to the source of it's weight, in this example a link to the experiment and some meta information (threshold θ, date of analysis, reference to exact computation of weight).

  Textual analysis where co-occurrence of named entities within a specific distance (= words in between) results in a weak link to be introduced. The weight depends on distance and quality of text source.

  Links between gene and protein names derived from scientific articles based on a bigram analysis. Weights are derived from the average distance and frequency of occurrence in documents, analogous to the TFIDF-score (Term frequency / inverse document frequency).

- ontology/thesaurus links: Based on an existing ontology links will be introduced to connect entities that are related based on this ontology. This resembles a 1:1 correspondence between each link in the ontology and a link in the network. The resulting links are strong links, i.e. carry a weight of 1.0 since there is (usually) no doubt about the reliability of that particular piece of knowledge. Otherwise it would need to be reflected in the link's weight.

Obviously many other types of links can be generated, since the underlying structure is invariant to origin or meaning of links.

3.4 Exploration: Finding interesting associations

The network's structure can be used in various ways to find potentially interesting pieces of information. Most straightforward would be the search for tightly connected other entities, such as another gene that is related to the ones the user just saw within an experiment. This can be implemented via a simple neighbor-search in the network, finding all genes that are connected to the set of ”query” genes.

More powerful are, however, searches that find related pieces of information via various steps, or so-called bridge concepts. This can be implemented analogous to activity spreading methods, as known from the neural network community (Cohen and Kjeldsen, 1987). The real power, in the concept presented here, lies in the ability to perform this search interactively. Throughout the search the user can weight entities that he finds interesting positively (and the ones he does not care about negatively), instantly affecting the activation pattern and hence the associations the network proposes. Such an interactive scheme will heavily rely on a suitable visualization of the graph network (see, e.g. Chen (2004)) and appropriate adaptive user interfaces.

4 Conclusions

In this paper we have briefly presented the idea of a generalized associative information network. With this concept we try to simulate aspects of the associative capabilities of the human brain in order to support a user in gathering information about a specific problem at hand. The tool is not meant to offer problem solving capabilities, but rather to point out information pieces a user might have otherwise not had the chance to look at, be it for lack of knowledge about their existence or because of a failure to see their importance for the task at hand.

References

R.K. Belew. *Finding out About*. Cambridge University Press, 2000.

T. Berners-Lee, J. Hendler, and O. Lassila. The semantic web. *Scientific American*, May, 2001.

M. Biezunski, M. Bryan, and S.R. Newcomb, editors. *ISO/IEC 13250:2000 Topic Maps: Information Technology – Document Description and Markup Languages*. ISO/IEC, 1999.

C. Borgelt, J. Gebhardt, and Ru. Kruse. Possibilistic graphical models. In G.D. Ricci, R. Kruse, and H.-J. Lenz, editors, *Computational Intelligence in Data Mining*, pages 51–68. Springer-Verlag, Wien, 2000.

T. H. Cao. *Fuzzy Conceptual Graphs: A Language for Computational Intelligence Approaching Human Expression and Reasoning*, pages 115–120. Physica-Verlag, Heidelberg, 2000.

C. Chen. *Information Visualization*. Springer, 2004.

H. Chen. Machine learning for information retrieval: neural networks, symbolic learning, and genetic algorithms. *J. Am. Soc. Inf. Sci.*, 46(3):194–216, 1995.

Yi-Ming Chung, William M. Pottenger, and Bruce R. Schatz. Automatic subject indexing using an associative neural network. In *DL '98: Proceedings of the third ACM conference on Digital libraries*, pages 59–68, New York, NY, USA, 1998. ACM Press.

P.R. Cohen and R. Kjeldsen. Information retrieval by constrained spreading activation in semantic networks. *Information Processing and Management*, 23(2):255–268, 1987.

A.M. Collins and E.F. Loftus. A spreading activation theory of semantic processing. *Psychological Review*, 82:407–428, 1975.

Didier Dubois, Jerome Lang, and Henri Prade. Automated reasoning using possibilistic logic: Semantics, belief revision and variable certainty weights. *IEEE Trans. Data and Knowledge Engineering*, 6 (1):64–71, 1994.

Jörg Gebhardt and Rudolf Kruse. *Reasoning and Learning in Probabilistic and Possibilistic Networks: An Overview*, volume 912 of *Lecture Notes in Artificial Intelligence*, pages 3–16. Springer-Verlag, Berlin, 1995.

Teuvo Kohonen. *Associative Memory - A System Theoretic Approach*. Springer-Verlag, Berlin, 1977.

Teuvo Kohonen. *Self-Organization and Associative Memory*. Springer-Verlag, Berlin, 1984.

Observing Dynamics in Community Structures

Tanja Falkowski
Otto-von-Guericke-Universität Magdeburg
Universitätsplatz 2, 39106 Magdeburg
`falkowski@iti.cs.uni-magdeburg.de`

Myra Spiliopoulou
Otto-von-Guericke-Universität Magdeburg
Universitätsplatz 2, 39106 Magdeburg
`myra@iti.cs.uni-magdeburg.de`

Abstract

Network analysis methods are widely used to detect community structures in static graphs. Since these structures undergo changes caused by internal and external factors it is necessary to provide methods to detect and observe transitions in community structures. For this we partition the interactions of community members by time windows and aggregate them. The resulting static graphs of each interval are analyzed for sub-communities. Through this we detect if communities persist over time or undergo a transition. We briefly present an interactive software environment which supports a temporal community analysis and provides several forms of visualization and analysis settings.

1 Introduction

Communities have proven to be of strategic importance e.g. to improve knowledge sharing or to enhance customer retention. Thus, communities have been studied in many research fields. So far they are mainly regarded as a static phenomenon and aggregated data over longer periods has been used to detect communities. However, the analysis of aggregated interactions between community members has some drawbacks. Old interactions are favored over newer ones and temporal developments in the interaction behavior can not be observed. But since communities are highly dynamic social networks, observing its transitions along the time axis is an important task, e.g., to adapt community platforms in order to support community building or to improve the efficiency of communities.

Tools such as SoNIA (Moody et al., 2005) and TeCFlow (Gloor and Zhao, 2004) visualize temporal social graphs by creating movies of them. Both tools work on the vertex and edge level thus visualizing a changing behavior between single actors. We propose to analyze and visualize temporal changes on the community level to allow for an exploration of sub-group dynamics.

Therefore, we regard a community as an object that exists over time and propose a dynamic temporal observation along the time axis using sliding time windows. The communities are detected in a static representation of interactions that occur in a specified period (cf. Section 2.1). We determine the evolution of the interactions by comparing the communities in different time windows. By this we are able to detect different types of transitions a community might pass through such as a split or a merger (cf. Section 2.2). The changes in the community structure are visualized and the user can choose different analysis settings to further explore the dynamics of the community under investigation in order to detect triggers that caused a community transition (cf. Section 3).

2 Model Community Dynamics

The problem of detecting community structures in networks is of interest in social sciences as well as for many other research fields such as computer science (e.g. WWW, e-mail log files) or biology (e.g. gene or protein networks) (see, e.g., Aggarwal and Yu, 2005, Kleinberg and Lawrence, 2001, Wilkinson and Huberman, 2004). In Section 2.1 we briefly discuss the method we use to detect communities in static graphs. In Section 2.2 we describe how we apply this method to observe the dynamics of community structures.

2.1 Community Detection

First, we model the network of interactions in a way suitable to find communities. We do so by defining a graph G = (V, E), in which V denotes the set of vertices (nodes) and E the set of edges (i, j), with $i,j \in V$. Each community member i is denoted a distinct vertex and an interaction between two members i and j, e.g., an e-mail exchange, is represented as an edge (i, j). We quantify the interaction between two members by assigning a weight $w(i, j)$ to the edge (i, j). Appropriate weights are "number of messages exchanged" or the "total length of all messages measured in characters".

Next, we decompose the graph into communities. We define a (sub) community as a subset of vertices within a graph with a high degree of interaction among the participants. We apply a hierarchical divisive clustering approach that divides the graph by the iterative removal of edges. The edges that are removed should be those that do not contribute to a community.

To determine the edge to be removed in each iteratio we use the *edge betweenness* score proposed by Girvan and Newman (2002). The betweenness of an edge is the number of shortest path between pairs of vertices that run along it. It is based on the assumption, that the few edges between communities have more "traffic", as, e.g., an information flow between vertices in two communities has to travel along these edges. The hierarchical clustering algorithm iteratively removes the edges with the highest edge betweenness score. We apply this method to a *multigraph* as described by Newman (2004) to include weighted edges. Each edge betweenness value is divided by the edge weight. Therefore, the edge betweenness value between two very connected pairs is lowered so that rather weak connected pairs are separated faster than strong connected ones. The algorithm has a high complexity due to the recalculation of the edge betweenness in each iteration – $O(m^2n)$, where m is the number of edges and n the number of vertices. However, it is applicable for small networks with up to a few thousand vertices.

The results of the hierarchical clustering are presented in a dendrogram, a tree diagram, which illustrates the community structure of the graph (see Figure 1). Since we have no a priori knowledge about the number of communities that exist in a network, we need an indicator on where to partition the dendrogram to obtain a meaningful network partition. For this purpose, we use the quality function proposed by Newman and Girvan (2004) to determine the best dendrogram cut which is based on the concept of modularity. The quality function Q is defined as:

$$Q = \sum_{i} e_{ii} - \sum_{ijk} e_{ij} e_{ki} = Tr(e) - \left\| e^2 \right\| ,$$

where e_{ii} is the fraction of edges in the original network that connect two vertices inside the community i and e_{ij} the fraction of edges that connect vertices in community i to those in community j. $\|x\|$ indicates the sum of all elements in x. Q has a value between 0 and 1. Values above 0.3 appear to indicate a significant community structure. Values approaching 1 indicate a strong community structure.

2.2 Community Transitions

Since the interactions between participants and the set of participants are not static but change over time, we use the representation of the network but consider the graph as dynamic. Vertices as well as edges appear and disappear from the graph through time. We define the dynamic graph g_t as a graph which consists of all vertices and edges that are active in the interval t. If all interactions would be aggregated over time to G by summing up all g_i all information about the temporal development would be lost. Therefore, we define g_t as a sliding time window over time interval t that spans a set of interactions. In other words, all interactions that take place in this interval t are aggregated to g_t.

After defining the interval we can partition the graph over time into equidistant time slots, each slot starting when the last slot finished. This modus is called a *non-overlapping sliding window*. An *overlapping sliding window* partially overlaps with the prior window. The degree to which it overlaps must be defined. We apply an overlapping window since it smoothes out the gaps that sometimes occur between two intervals.

Each window is considered a static representation of the network in the chosen interval. At first we apply the community detection mechanism as described in the previous section to obtain a community structure for g_t. $C^{g_t} = \{c_1^{g_t}, \ldots, c_n^{g_t}\}$ is the set of communities that are detected in g_t. To determine whether a community persists over time we must be able to assess if a community $c_i^{g_x}$ is the same as a community $c_j^{g_y}$. Qualitatively we would define that a community in a subsequent interval is the same, if the characteristic features are similar. In the easiest case we would say that this is the set of participants. We therefore define that community $c_i^{g_x}$ and community $c_j^{g_y}$ are the same if they share a given percentage of members. The appropriate percentage depends on the community type, the type of relations and the intent of the observer. If a community e.g. consists of a small set of very active core members and a high number of less connected members that often change, the percentage should be rather small. Otherwise, the community might not be considered the same just because many of the other "uncharacteristic" members changed, even though the most active core members are still detected as a community in different intervals.

Besides deciding whether two communities are the same, which would mean that a community persists over time, we can also observe if a community merges with another community or if it splits into separated communities. These developments can be triggered by internal factors such as a change in leadership as well as external factors, e.g. advertisements. The challenge is to determine the factors that positively trigger the community development to offer appropriate organizational and technological infrastructures.

3 Temporal Community Analysis

To track the development of established communities and to visualize the transitions we developed a software environment that supports temporal graph analysis based on the community detection methods described in Section 2. In the following section, we briefly describe the functionality of the software.

3.1 Visualizing Community Structures

The hierarchical divisive clustering algorithm is used as described in Section 2.1 to find the communities in each interval. The results are displayed in a dendrogram. The user can experiment on the impact of different clusterings on the quality measure by moving a slider as can been seen in Figure 1.

Figure 2 has two areas: a list of all detected communities in the left window and the curves on the right. The horizontal axes represent the respective time windows.

The lowermost curve displays the total number of interactions between the chosen group and the total number of interactions of all group members with other participants of other communities. It can be seen that the chosen group is only active for about 6 weeks, but some members have an active relation with external participants. The middle and the uppermost diagram show how similar the internal community interaction behavior is over time. In the middle diagram the vertical axis depicts the correlation distance as a similarity measure for the groups in different periods. It can be seen that the group shows up in two time windows with almost the same members but the structure changes very quickly and the group disappears. In the uppermost diagram the y-axis displays the Euclidian distance as a similarity measure. The more similar a group interaction in two periods, the lower is the value of the Euclidian distance. For both measures we compare the similarity between the chosen interval and all other intervals and for two succeeding intervals. The statistics can for example be used to find point in time where the interaction behavior of a group changes compared to previous intervals. If several groups show similar behavior, this might be an indication for a change in the overall community structure.

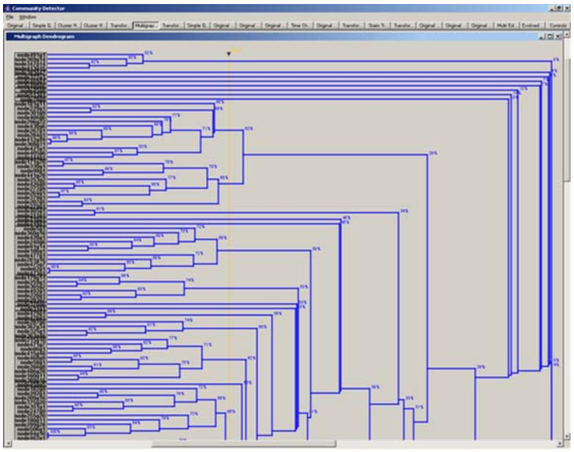


Figure 1: Clustering results in a dendrogram

Figure 2: Statistics for temporal development for a chosen community

3.2 Visualizing Community Dynamics

In Figure 3 we see a static representation of a temporal community evolution. In this visualization, each detected community is represented as a vertex. The size of the vertex corresponds to the size of the community. Vertices that are connected by an edge are similar. Communities with the same members over several periods are positioned closer in the graph whereas communities with no members in common are more separated from each other. Furthermore, the different colors help to distinguish between similar communities and those that are not.

The user can choose for how many periods the community must at least exist to be displayed. If a long period is chosen, the user obtains only long-term community whereas in another case it might be of interest to find only short-term communities. Another slider for the time distance defines how continuous the communities are connected, separating communities by a maximum distance. Furthermore, one can define the observation period and filter the vertices so that the communities are displayed only in a selected period. The described properties can be used to filter communities so that the graph only shows data that is useful for a current analysis.

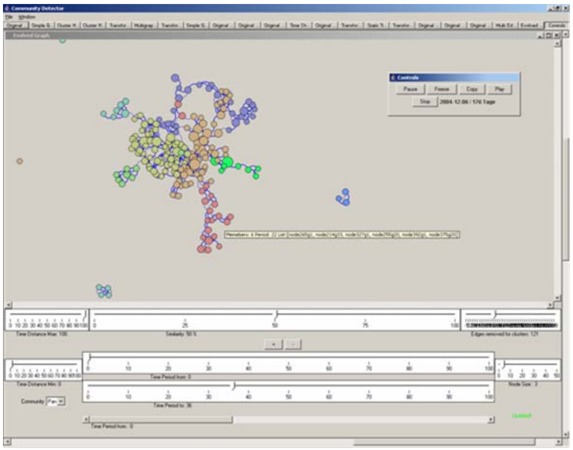


Figure 3: Visualization of communities based on Euclidian distance

Note that in the obtained graph in Figure 3 the temporal development can not be observed, as the communities are only displayed according to their similarity. In a next step, the filtered and clustered data is copied to a community history view, which allows seeing temporal developments by using the coordinates from the graph and putting the vertices on the horizontal axis according to the period they appear in (see Figure 4). The position transformation allows tracking the development along the time axis. Each community is now represented as a rectangle where its height corresponds to the size of the community. All communities that are considered as similar according to the actual settings are connected by edges and have the same color.

The left side of the screenshot in Figure 4 shows all communities over time in an overview window. The x-coordinate of each community is the same as in Figure 3. The y-coordinate is determined by the interval in which the community was detected. Thus, communities on the left are observed in earlier periods than those on the right.

The analyzed community shows different developments. In the lower part of the left side we can see an insolated community in light blue and one in red. Both existed over just a few periods. Some members of the red community joined another community which is shown in dark blue. In the cutout view on the right we can furthermore observe a community in yellow that existed over a longer period but died at some time. We can see that smaller communities merge to a bigger one but split very fast to several smaller communities. Points in time when these changes occur might be of interest for the user, as they indicate an external or internal event that influenced the community development. These effects might then be deliberately used to improve the establishment of communities.

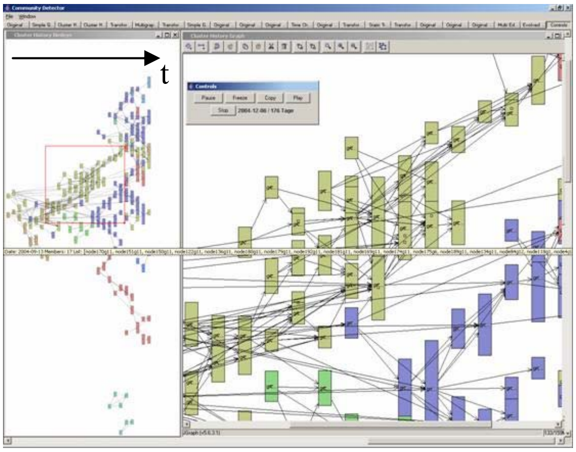


Figure 4: Community History View

4. Summary

We presented a software platform which allows for the analysis of the temporal development of communities. The gained insights can be helpful to understand community transition types and its triggers. This knowledge can be used to provide an appropriate technological as well as organizational infrastructure to foster community building.

References

Charu C. Aggarwal and Philip S. Yu. Online Analysis of Community Evolution in Data Streams. *Proceedings of SIAM International Data Mining Conference*, 2005.

Michelle Girvan and Mark E. J. Newman. Community structure in social and biological networks. *PNAS*, 99(12): 7821-7826, 2002.

Peter A. Gloor and Yan Zhao. TeCFlow - A Temporal Communication Flow Visualizer for Social Networks Analysis. *CSCW'04 Workshop on Social Networks*, ACM, 2004.

Jon M. Kleinberg and Steve Lawrence. The Structure of the Web. *Science*, 294(5548): 1849-1850, 2001.

James Moody, Daniel Mc Farland and Skye Bender-deMoll,. Dynamic Network Visualization. *American Journal of Sociology*, 110(4): 1206-1241, 2005.

Mark E. J. Newman. Analysis of weighted networks. *Physical Review*, E 70 (056131), 2004.

Mark E. J. Newman and Michelle Girvan. Finding and evaluating community structure in networks. *Physical Review*, E 69 (026113), 2004.

Dennis M. Wilkinson and Bernardo A. Huberman. A method for finding communities of related genes. *PNAS*, 10(1073), 2004.

Protein interaction network analysis reveals the importance of proteins with low degree of connectivity in neurodegenerative diseases

Joaquín Goñi[⋆][§]
[⋆]Center Appl. Med. Res. (CIMA)
University of Navarra
31080, Pamplona (Spain)
jgoni@unav.es

Francisco J. Esteban[†]
[†]Dept. Exptal. Biology
University of Jaén
23071, Jaén (Spain)
festeban@ujaen.es

Nieves Vélez de Mendizábal[⋆][‡]
[‡]Dept. Comput. Sci. Artif. Intell.
University of the Basque Country
20018, San Sebastián (Spain)
nvelez@unav.es

Jorge Sepulcre[⋆]
jsepulber@unav.es

Sergio Ardanza-Trevijano[§]
[§]Dept. Physics Appl. Mathematics
University of Navarra.
31080, Pamplona (Spain)
sardanza@unav.es

Pablo Villoslada[⋆]
pvilloslada@unav.es

Abstract

Recent developments have meant that network theory is making an important contribution to the study of the topology of biological networks such as protein interaction (PI) networks. The identification of differentially expressed genes in DNA-microarrays experiments is a known source of information regarding the main molecular pathways involved in disease. Thus, considering PI analysis and gene expression studies together may provide us with a better understanding of the topological aspects of multifactorial neurodegenerative diseases such as Multiple Sclerosis (MS) and Alzheimer Disease (AD). The aim of this study was to assess whether degree of connectivity is a key property that differentiates between implicated (seed-proteins) and non-implicated nodes (neighbors) in MS and AD. We used PIs experimentally validated and an interaction distance threshold from each seed group to obtain two networks (one MS-network and one AD-network). Results revealed for both diseases a lower degree of connectivity in seed proteins than in their neighbors in each PI network. Furthermore, we found that the MS-network and AD-network included multiple pathways and followed a very similar exponential degree distribution but with different clustering coefficient behavior. These findings reinforce the multifactorial nature of both diseases and might lead to new therapeutic strategies.
keywords: Alzheimer Disease, differentially expressed genes, microarrays, Multiple Sclerosis, network topology

1 Introduction

The structural and functional relationships underlying the organization of living systems imply the need to coordinate molecular interactions, principally those involving gene expression and protein activity. Although the genome is (almost) the same in each cell population of a given organism, dynamic changes in gene expression and thus in the protein content depend on the functional state of the cell (Lodish, 2002).

Genome-wide expression profiles using DNA microarrays, together with the development of bioinformatics approaches (Villoslada, 2006), enable us to model both genetic and protein interaction (PI) networks and thus to understand how a biological network operates (Xia, 2004). From a systems point of view (Hiesinger, 2005), the arrangement of biomolecular networks from gene expression data based on known interactions permits to understand the basic mechanisms upon which the complexity and adaptability of a living cell is founded. This information also helps us to decipher processes involved in illness, for instance the molecular heterogeneity of cancer (Rhodes, 2005). However, and consistent with the model of multifactorial diseases, it is difficult to find genes that account for direct genotype-phenotype correlation (Gunsalus, 2005). Thus, network modelling and topological analysis may provide additional knowledge about common properties of genes and proteins involved in many severe diseases of multifactorial nature, where the cause of the pathogenesis does not depend on the malfunction of a single gene or protein. In addition, from a medical point of view, systems biology approaches to complex diseases might represent a standpoint to identify new therapeutic targets. In this case, the analysis of genes, proteins and pathways interactions might suggest common properties of the best candidates to be targeted by therapy. In addition, the understanding of the emergent properties of a system might allow the identification of new targets that will not be captured with a molecular approach (Kitano, 2004).

Multiple Sclerosis (MS) is a chronic inflammatory and neurodegenerative disease of the central nervous system (CNS) (Steinman, 2001). Its etiology remains elusive, but the interplay between environmental and genetic factors is ultimately thought to be critical to the development of the disease. MS is considered to be an autoimmune disease because of the presence of inflammatory infiltrates in the brain, in absence of infection, and its association with HLA alleles, among other factors (Oksenberg, 2001). The chronic inflammatory activity within the CNS is the main mediator of tissue damage, even in the late neurodegenerative stage of the disease, which includes widespread demyelination and axon loss (Bruck, 2005). In addition to the autoimmune processes, MS also has a neurodegenerative component whereby axons and neurons are lost through unknown processes in the late chronic stages of the disease. Several lines of evidences suggest that dying-back degeneration of demyelinated axons is the most important factor in MS neurodegeneration (Imitola, 2006). MS is a multifactorial disease in which many pathways of the immune system and CNS are involved (Fernald, 2005). Current therapies ameliorate in part the inflammatory process, but more effective therapeutical approaches are required to completely stop disease progression and prevent neurodegeneration.

Alzheimer Disease (AD) is the most common neurodegenerative disease, representing one of the biggest unmet needs in modern medicine (Walker, 2004). AD is characterized by the loss of neurons in association with the presence of oxidative stress, axonal dystrophy, mature senile plaques and neurofibrillary tangles (Cummings, 2004). A set of mutations in genes involved in amyloid beta and tau pathways have been associated with hereditary AD and in conjunction with neuropathological findings, the amyloid and tau hypothesis for the pathogenesis of AD has been put forward. However, current evidence suggests that sporadic AD is a multifactorial disease in which many pathways are involved (Cummings, 2004). Because the available therapies are only symptomatic (Scarpini, 2003) and considering the epidemic proportions of this disease in western countries, the development of new therapies to stop its progress is a major health priority.

In order to understand more about the basis of neurodegenerative diseases, the aim of this study was to assess the degree of connectivity between proteins whose genes were differentially expressed in MS and AD and their protein neighbors. In short, we tested whether the degree of connectivity is a property that differentiates between implicated (seed-proteins) and non-implicated nodes (neighbors). We also studied the topological properties of both MS-network and AD-network with a special focus on degree and clustering coefficient distributions.

2 Materials and Methods

2.1 Definitions

Some definitions were introduced to better explain the development of our topological studies. There are tree concept definitions for each disease and two general terms.

-MS seed-proteins: proteins whose genes were differentially expressed in previous microarray studies of MS (Bomprezzi, 2003).

-MS-neighbors: nodes selected as consequence of adding experimentally validated interactions starting from MS seed-proteins.

-MS-network: network that includes MS seed-proteins, MS-neighbors and their interactions.

Thus, MS-network nodes can be partitioned into two groups: MS seed-proteins and MS-neighbors.

-AD seed-proteins: proteins whose genes were differentially expressed in previous microarray studies of AD (Walker, 2004).

-AD-neighbors: nodes selected as consequence of adding experimentally validated interactions starting from AD seed-proteins.

-AD-network: network that includes AD seed-proteins, AD-neighbors and their interactions.

Thus, AD-network nodes can also be partitioned into two groups: AD seed-proteins and AD-neighbors.

-Disease-networks: this term is used to refer to both MS-network and AD-network.

-Degree: The so-called degree of connectivity. In this paper, it represents the number of experimentally validated interactions (links) that are connecting one protein (node) to its neighbors.

Table 1: Genes identified in MS expression profile study by Bomprezzi (2003).

Gene name	Brief protein description	k^*
JUN	Transcription factor	88
HSPA1A	Heat shock protein	66
BCL2	Apoptosis regulator	60
ZAP70	Tyrosine kinase	32
ATM	Serine kinase	26
SPTAN1	Spectrin α chain	24
MADH7	Mothers ag. decapentaplegic	18
ITGA6	Integrin alpha-6	17
TRAC	T-cell receptor region	14
HLA-DRA	MHC class II antigen	13
SCYE1	Multisynthetase complex	11
PAFAH1B1	PAF acetylhydrolase	11
SCYA3	Cytokine A3 precursor	10
IL7R	IL-7 receptor precursor	8
DNAJA1	DnaJ homolog	8
XPC	DNA-repair protein	7
SEC34	Golgi complex component	7
PPP2R5C	Ser/Thr phosphatase	4
DNTT	Nucleotidylexotransferase	4
TIMP1	Metalloproteinase inhibitor	3
SPTBN1	Spectrin β-chain	3
SERPINH2	Sphingosine kinase 2	3
TNFRSF7	TNF receptor precursor	3
GOLGA4	Golgi autoantigen	2
PTP4A1	Tyr-phosphatase	2
ZNF148	Zinc finger protein	2
CCR7	C-C chemokine receptor	2
IKKE	Inhib. NFκ-B kinase	1
NKTR	NK-tumor protein	1
DPPIV	Seprase	1
CSK2	Cyclin kinase subunit	1
DGKA	Diacylglycerol kinase	1
PIK3R4	PI-3-kinase	1
BRF1	Butyrate response factor	0
CD83	CD83 antigen precursor	0
BAZ2B	Bromodomain	0
TTC3	Tetratricopeptide protein	0
ZNF43	Zinc finger protein	0
9235	NK cells protein precursor	0
IFI30	Thiol reductase precursor	0
PDE7A	cAMP-phosphodiesterase	0
SLC35A1	CMP-sialic acid transporter	0
TCF7	Transcription factor	0
MAL	T-lymphocyte maturation	0
H1F2	Histone H1.2	0

*Degree (see section 2.3).

It is important to remark that we did not consider neighbors as new proposals for proteins implicated in disease, but they were taken to capture the network context where seed-proteins were involved.

2.2 Gene expression data

2.2.1 Multiple Sclerosis

For MS-network construction and analysis, we selected seed-proteins from previously published data (Bomprezzi, 2003) listed in Table 1. These included 45 genes differentially expressed in peripheral blood mononuclear cells from 24 MS patients with respect to 17 controls, identified by using cDNA microarrays.

2.2.2 Alzheimer Disease

The set of selected seed-proteins for AD-network modelling and analysis is listed in Table 2. It contains the gene products of 37 differentially expressed genes detected elsewhere (Walker, 2004) using cDNA-microarrays from postmortem cerebral RNA extractions in 5 normal and 4 clinically diagnosed AD patients.

2.3 Network modelling

Starting from seed-proteins involved in MS, we obtained a PI network (MS-network) throughout the interactions of these proteins. Figure 1 shows a general scheme of the approach performed in this paper. We considered a minimum of one thousand neighbors as an appropriate size to analyze the network context where the seed-proteins were involved. Hence, we expanded each disease-network until the one thousand nodes mark was reached. A depth-2 configuration allowed us to obtain 1127 neighbors in the MS-network, which included proteins directly interacting with seed-proteins. We applied the same approach using AD seed-proteins, obtaining 331 neighbors using depth-2, and 1640 neighbors using depth-3 expansion (which includes direct and with one intermediary interactions). Thus, we used depth-3 for AD-network as expanding threshold.

The growth of each network was carried out with `STRING` database (von Mering, 2003, 2005)[1]. The parameters used as criteria for network growing in the `STRING` database were: *active prediction method*: experiments; *confidence score*: 0.7 -high confidence-; *network depth*: 2-3; and *edge scaling factor*: 80%. This configuration involves just those experimental

---

[1] http://string.embl.de/

Table 2: Genes identified in AD expression profile study by Walker (2004).

Gene name	**Brief protein description**	k^*
EEF1A1	Elongation factor	70
B2M	β-2-μglobulin precursor	17
GRP58	Disulfide isomerase	16
CLU	Clusterin	14
DTNA	Dystrobrevin α	14
CD81	Surface antigen	13
HLA-B	Histocompatibility antigen	12
ATF4	Transcription factor	11
KRT8	Keratin, cytoskeletal	11
APLP1	Amyloid-like precursor	8
C4B	Complement C4 precursor	8
RAPD1GDS1	Stimulatory GTP exch.	7
183	Angiotensinogen precursor	6
CDC10	Septin 7	6
RANGAP1	GTPase-activating protein	6
FTL	Ferritin light chain	5
NEDD5	Septin 2	5
HBB	Hemoglobin beta chain	4
DMPK	DM protein-kynase	3
GSTM2	Glutathione S-transferase	3
PRDX1	Peroxiredoxin	3
MT1G	Metallothionein	2
P60201	Proteolipid protein	2
PLEKHB1	Evectin	2
HBG1	Hemoglobin epsilon chain	1
HBG2	Hemoglobin epsilon chain	1
IGHM	Ig α-1 chain C region	1
LIMS2	Senescent antigen-like	1
10099	Tetraspanin 3	0
ADD3	Gamma adducin	0
CHN2	β-chimaerin	0
OSBPL3	Oxysterol binding protein	0
PCL1	PrenylCys oxidase prec.	0
PCSK1N	Proprotein convertase inh.	0
PTS	6-pyruvoyl THB-synthase	0
RPL31	60S ribosomal protein L31	0
TU3A	TU3A protein	0

*Degree (see section 2.3).

evidences of interactions with high confidence, which were extracted from the database as valid links for each PI network. A detailed description of each parameter can be found elsewhere (von Mering, 2003). We did not consider neither the direction of each protein interaction nor self interactions.

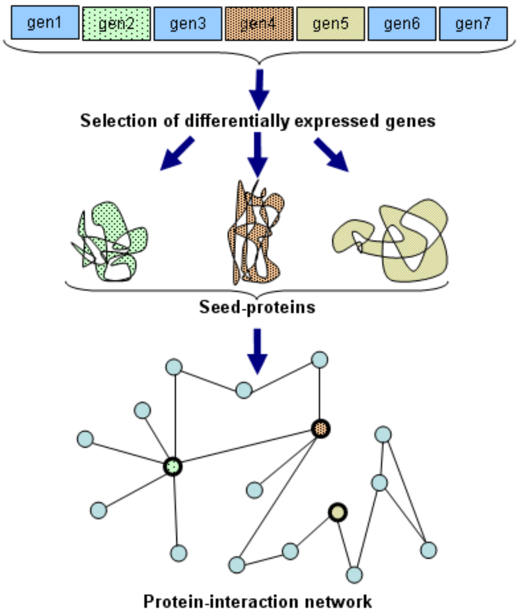


Figure 1: General scheme of the approach for each disease.

2.4 Topological analysis

We analyzed the degree distribution *P(k)*,

$$P(k) = \frac{n(k)}{N}\,, \tag{1}$$

where *k* is the number of links connected to a given node and *n(k)* the number of nodes with degree *k*.

In order to assess the degree distribution, a power law approximation (Barabási, 1999),

$$P(k) \sim k^{-\gamma}\,, \tag{2}$$

was first studied plotting *P(k)* versus non-zero *k* in log-log scale; the so-called frequency-degree. The scaling exponent γ was obtained from the slope absolute value of the least-squares fit.

The clustering coefficient for a node *i* with k_i neighbors, $C_i(k_i)$, represents the ratio of the number of actual connections between the neighbors of node *i* to the number of possible connections:

$$C_i(k_i) = \frac{2n_i}{k_i(k_i-1)}\,, \tag{3}$$

where n_i is the number of links connecting the k_i neighbors of node *i* to each other (Nacher, 2004).

The average clustering coefficient (i.e., the clustering of nodes with respect to *k*) provides information about the modular organization of networks (Almaas, 2006),

$$C(k) = \frac{\sum_{i:k_i=k} C_i(k_i)}{n(k)} \,. \qquad (4)$$

The rank-degree distribution, *Rank(k)*, represents the number of nodes with a degree greater than *k*,

$$Rank(k) = n(K > k) = \sum_{k'=k+1}^{m} n(k') \,, \qquad (5)$$

with *m* being the maximum degree found (Tanaka, 2005b).

It is important to notice that *Rank(k)* provides precise information to elucidate whether degree distribution is better explained either as a power law or as an exponential distribution (Tanaka, 2005b).

P(k), *C(k)* and *Rank(k)* were calculated from the `STRING` files (simple tab delimited flatfiles), which contain all the nodes and interactions obtained using the methodology described in section 2.3.

2.5 Gene Ontology

In order to assess the multifactorial character of the biological pathways in which seed-proteins were involved following the *Gene Ontology*[2] (GO), we studied those pathways arising when genes corresponding MS seed-proteins or AD seed-proteins were compared to complete human proteome (Swissprot identifiers). This approach was carried out using the `FatiGO` web tool (Al-Shahrour, 2004)[3].

2.6 Statistical analysis

Frequency-degree linear approximation were carried out with the least squares fitting method. Rank-degree curves were also fitted as a second order exponential decay. To compare the degree distribution between all the network nodes and the seed-nodes, we performed an ANOVA test using `Statgraphics Plus 5.1` software. The regression lines were compared using the *Comparison of Regression Lines Analysis Dialog Box*, which automatically constructs the necessary indicator variables for comparing two or more simple regression models. Finally, we used a U Mann-Whitney test to compare degree between seed-proteins and neighbors for each disease.

The level of significance was set at p<0.01. P-values associated to pathways under `FatiGO`[3] analysis were corrected by false discovery rate multiple comparison method.

3 Results

3.1 Multiple sclerosis

The MS-network is shown in Figure 2. This map contains 1172 nodes, including 45 seed-proteins and 1127 neighbors. Twelve seed-proteins had no links (i.e., no experimental evidence of interactions), and 8 nodes (including 1 seed-protein) formed an independent small fully interlinked net. The degree corresponding to each MS seed-protein node is listed in Table 1.

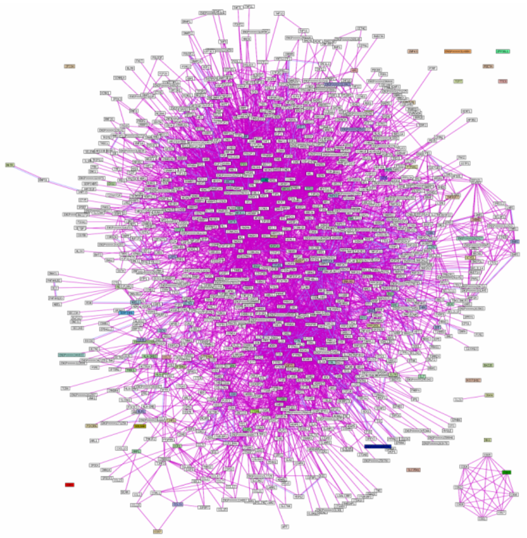


Figure 2: Protein interaction of the Multiple Sclerosis network (MS-network).

The frequency-degree distribution, the clustering coefficient and the rank-degree distribution of MS-network nodes are shown in Figure 3. We found a dependence of *P(k)* respect to *k* in both MS-network nodes and MS seed-proteins. According to equation (2), the linear approximations (slope = -1.48 and slope = -0.43) explained these dependencies as power laws $P(k) = k^{-1.48}$ and $P(k) = k^{-0.43}$ respectively. Furthermore, clustering coefficient also followed a power law respect to *k*, with (γ = 0.68). On the other hand, rank-degree plot revealed that degree distribution was better explained as exponential ($R^2 = 0.99$) instead of linear ($R^2 = 0.90$).

The comparison of MS-network and MS seed-proteins regression lines provided a statistically significant difference (p<0.001). Furthermore, degree group comparison indicated that seed-proteins degree was significantly lower than MS-neighbors degree (p=0.003).

Finally we assessed whether some functional pathways were overrepresented in the genes set corre-

---

[2] http://www.geneontology.org/

[3] http://fatigo.bioinfo.cipf.es/

sponding to MS seed-proteins. We found that none of the 22 biological modules detected (GO level 3) were statistically overrepresented.

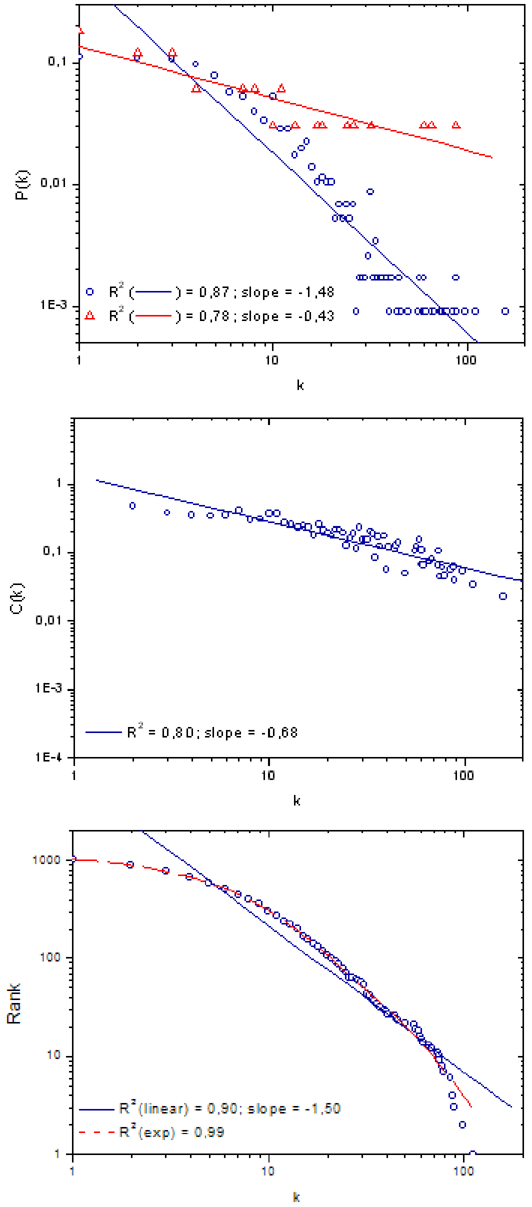


Figure 3: Topological analysis of the MS-network (log-log plots). Top: frequency-degree distribution of MS-network nodes (blue-circles) and MS seed-proteins (red-triangles). Middle: clustering coefficient distribution of MS-network nodes. Bottom: rank-degree distribution of MS-network nodes.

3.2 Alzheimer Disease

The AD-network contains 1687 nodes, including 47 seed-proteins and 1640 neighbors (figure 4). Table 2 includes the degree of connectivity for each seed-node.

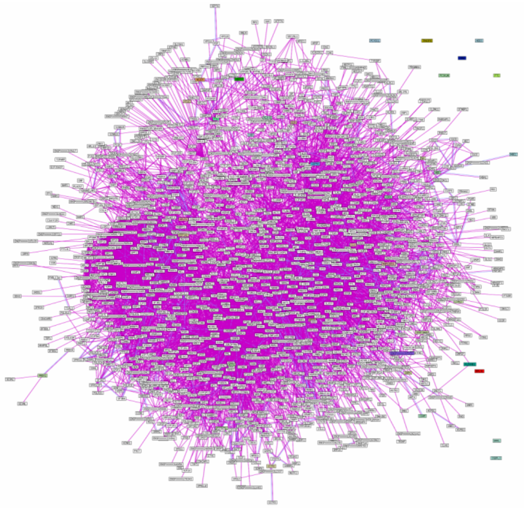


Figure 4: Protein interaction of the Alzheimer Disease network (AD-network).

The frequency-degree distribution, the clustering coefficient and the rank-degree distribution of AD-network nodes are shown in Figure 5. We found a dependence of *P(k)* respect to *k* in both AD-network nodes and AD seed-proteins. According to equation (2), the linear approximations (slope = -1.58 and slope = -0.36) explained these dependencies as power laws $P(k) = k^{-1.58}$ and $P(k) = k^{-0.36}$ respectively. On the other hand, clustering coefficient resulted to be independent from *k* (γ = 0.28 and $R^2 = 0.08$). Finally, rank-degree plot revealed that degree distribution was better explained as exponential ($R^2 = 0.99$) instead of linear ($R^2 = 0.81$).

The comparison of AD-network and AD seed-proteins regression lines provided a statistically significant difference ($p<0.001$). Furthermore, degree group comparison indicated that seed-proteins degree was significantly lower than AD-neighbors degree ($p=0.005$). These results were very similar to the obtained in the MS study in section 3.1.

Finally we assessed whether some functional pathways were overrepresented in the genes set corresponding to AD seed-proteins. We found that none of the 9 biological modules detected (GO level 3) were statistically overrepresented.

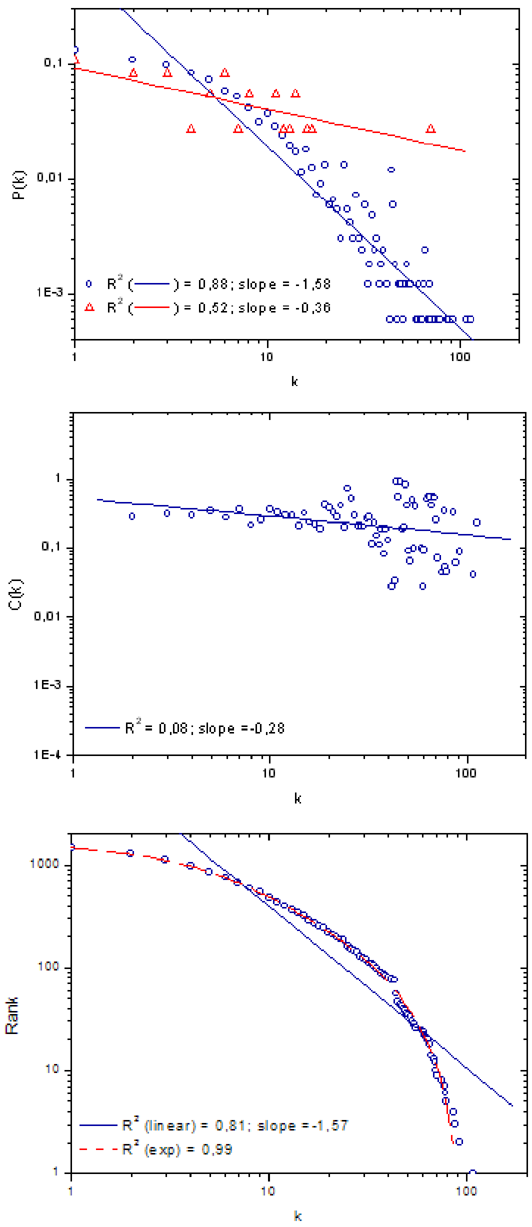


Figure 5: Topological analysis of the AD-network (log-log plots). Top: frequency-degree distribution of AD-network nodes (blue-circles) and AD seed-proteins (red-triangles). Middle: clustering coefficient distribution of AD-network nodes. Bottom: rank-degree distribution of AD-network nodes.

3.3 Common characteristics between MS and AD networks

As indicated in Tables 1 and 2, we found a low degree in seed-proteins respect to the degree of its PI neighbors in both diseases, with only 3 (MS) and 1 (AD) highly connected seed-proteins (k>60). In addition, direct interactions between seed-proteins were very low: 4 direct interactions in MS-network, and 5 in AD-network. There were 586 common proteins to MS and AD networks. In order to detect the possible topological distribution relationship between both neurodegenerative disorders, the disease networks and the seed-proteins sets were independently compared through regression line analysis (Figure 6), and no significant differences between slopes were detected neither between disease-networks (p=0.31) nor between seed-proteins (p=0.52).

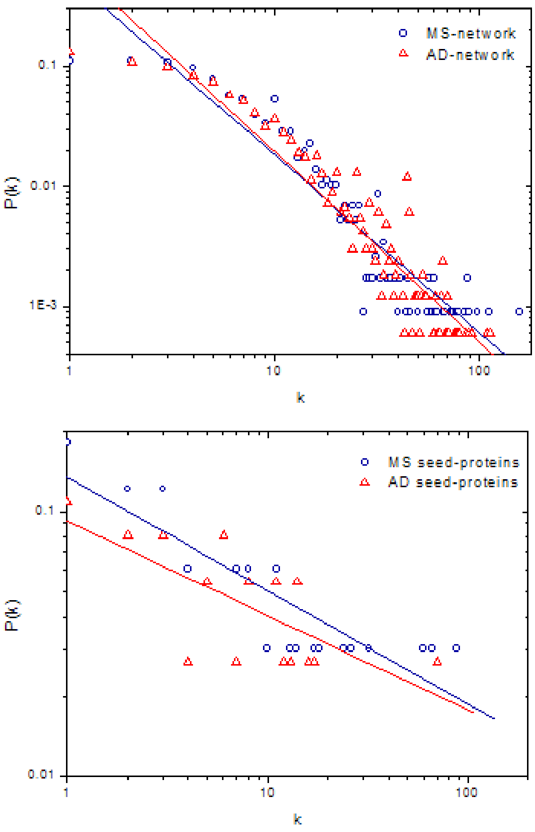


Figure 6: Linear regression comparison between disease-networks and seed-proteins.

4 Discussion

4.1 Topology of MS and AD networks

Network theory can provide a useful tool to study the complexity of neurodegenerative diseases. In the present study we report a novel approach to study PI networks, based on the products of differentially expressed genes of MS and AD. The network growth was carried out expanding the network through experimentally validated protein interactions.

Network stability, dynamics and function is generally characterized by determining the topology of the map, i.e., the configuration of its nodes and the connecting edges (Han, 2005). One of the most succeeding features analyzed has been the degree distribution, and whether or not it followed a power law; the so-called scale-free property. Roughly speaking, these studies were aimed to characterize the properties of real networks in basis on their topological features. For example, networks with a SF topology are known to be resistant to random failure and vulnerable to targeted attack, specifically against the most connected nodes (hubs). However, it has recently been shown that metabolic networks are supported by different modular scales, with a power law degree distribution of the global system but an exponential behavior in modules. As such they are described as scale-rich (SR) networks (Tanaka, 2005a).

Our results reveal that MS-network and AD-network may be better adjusted to exponential than to power law distributions. The power law behavior of *P(k)* together with the dependence of *C(k)* versus *k*, point to a scale-free distribution (SF) with an inherent modularity of the MS-network. However, the exponential behavior of the rank distribution agree with a SR -instead of a SF- topology. The particular degree distribution of the seed-proteins involved in MS also showed a power law topology. In the case of AD, γ exponent values of AD-network nodes and AD seed-proteins were very similar to the obtained in MS-network nodes and MS seed-proteins respectively. Although a constant behavior of *C(k)* in AD-network also point to a scale-free distribution (SF), the exponential behavior of the rank distribution agree with a SR topology. In this sense, recent studies support that other biological networks, such as the complete human PI map, filtered yeast interactome dataset and metabolic networks, could be better explained as SR networks (Tanaka, 2005a,b). These networks would not be so hub-dependent as SF are, and could be formed by exponential subnetworks and critical nodes that might not be so *k*-dependent (according with both MS-network and AD-network). In addition, the clustering coefficient dependence of *k* point to a inherent modularity of the MS-network, but not completely reaching the characteristic hierarchical network. Hierarchical modularity is detected by the scaling of the clustering coefficient, which should follow $C(k) \sim k^{-1}$ (i.e., a straight line of slope -1 on a log-log plot) (Barabási, 2004).

On the other hand, both the MS and AD network may be considered as proportional SR samples of the complete human protein-interaction map recently studied (Tanaka, 2005b). This is particularly the case if we take into account that the rank-degree distribution is exponential and the frequency-degree distribution is linear, with very similar exponents. Regarding the identification of common properties among those genes involved in neurodegenerative disorders, these facts provided very interesting results under a topological analysis: multiple pathways affected by proteins with low degree, following very similar SR distributions (Tanaka, 2005a).

4.2 Seed-proteins connectivity

During the last decade, network studies have been applied to biological data bearing in mind that the degree of connectivity is a key property of any network. The most common approach to identify key nodes consists of obtaining networks from high throughput data and having obtained the network, searching for the highest connectivity nodes (hubs). The underlying assumption was that these hubs could be critical to explain pathogenesis of diseases.

Our study was performed from a novel viewpoint. Starting from critical nodes (in terms of differentially expressed genes) we analyzed whether the connectivity was higher or lower than the connectivity of the PI neighbors. Thus, we found that seed-proteins connectivity was lower than PI neighbors degree. To our knowledge, these properties have not been reported before and they situate seed proteins in peripheral regions of the network, distributed along several pathways that could be involved in disease. Somehow this enters into conflict with the hub relevance hypothesis (Jeong, 2001), at least in these two neurodegenerative diseases. Therefore our results support the application of strategies other than those previously applied, whereby only hubs that could compromise the robustness of networks were generally searched (Barabási, 2004).

The etiology of MS and sporadic AD still remains elusive and many environmental and genetic factors have been proposed. In this sense, our results strongly support the multifactorial nature of these diseases, due to the fact that many pathways participates somehow in both diseases, any of them being predominant.

4.3 Etiopathogenic and therapeutical implications

We can consider complex diseases as an evolutionary stage in which the pathogenesis process hijacks the robustness of the biological pathways; this may be followed by cascading failures in such pathways (Kitano, 2004, 2006). In this sense, it may be necessary to target many of the pathways involved, although following a systems biology rationale based on the dynamics and topology of the networks involved. The aim of this therapy would be to drive those pathways to a non-pathological state or at least, to a less deleterious state.

The topological implications of the claimed SF properties of biological networks suggests that the best therapeutic targets, in order to modify the network behavior, should be the genes (or proteins) corresponding to hubs in the network. However, our findings suggest that low connected proteins might be more appropriate therapeutic targets, at least in neurodegenerative diseases, than hyper-connected ones.

The fact that in both diseases (MS and AD) and in two different tissues analyzed (blood and cerebral tissue), seed-proteins were low connected nodes taking part in many different pathways, strengthen the multifactorial pathogenesis of neurodegenerative diseases. Our results suggest that in order to modify the disease course we need to target many genes or proteins in several pathways. Another reason why hubs might not be good therapeutic targets is because their critical role in the network modules might prevent them from fluctuating substantially. For the same reason, we can speculate that networks will poorly tolerate the modification of hub behavior without spreading such changes along the network and, in this way, inducing significant side effects.

4.4 Conclusions

The results presented in this paper indicate that both neurodegenerative diseases (MS and AD) share as common characteristics the low degree of seed-proteins and the degree-distribution similarities found between disease networks, even though many different pathways are involved depending on the disease. These findings locate seed-proteins mainly in peripheral regions of the PI map (in terms of degree), involved in many pathways (as FatiGO and low direct interactions results indicated) and integrated in two subnetworks (respect to Human complete proteome network) with very similar exponential degree distributions but with different modular organization. In addition, and as stated before, no significant biological process were overrepresented to the seed-proteins of MS and AD analyzed when compared to the whole human genome. This results are likely to be explained as a consequence of the multifactorial nature of both diseases.

Acknowledgements

We thank Cristian von Mering and Iván Martínez for helping us with the use of `STRING` database, Marc Sefton for English revision and Ricard Solé and Bartolomé Bejarano for their comments and suggestions. J.G, N.VdM. and J.S. are fellows supported by *Navarra Government* (Spain), *Basque Country Government* (Spain) and *Spanish Ministry of Health* (CM05/00222), respectively. This work was partially supported by *Junta de Andalucía* (CVI-302; to F.J.E.), *Spanish Ministry of Science and Technology* (BFM2003-05878 to S.A.T.) and *Spanish Ministry of Health* (FIS: 04/1445; to P.V.).

References

E. Almaas and A.L. Barabási. Power Laws in Biological Networks. *Power Laws, Scale-free Networks and Genome Biology*, E.V. Koonin, G. Karev and Y. Wolf (Eds.), Springer, New York (in press) 2006.

F. Al-Shahrour, R. Díaz-Uriarte and J. Dopazo. FatiGO: a web tool for finding significant associations of Gene Ontology terms with groups of genes. *Bioinformatics,*, 20(4):578–580, 2004.

A.L. Barabási and R. Albert. Emergence of scaling in random networks. *Science*, 286(5439):509–512, 1999.

A.L. Barabási and Z.N. Oltvai. Network biology: understanding the cell's functional organization. *Nature Reviews Genetics*, 5(2):101–113, 2004.

R. Bomprezzi, M. Ringner, S. Kim, M.L. Bittner, J. Khan, Y. Chen, A. Elkahloun, A. Yu, B. Bielekova, P.S. Meltzer, R. Martin, H.F. McFarland and J.M. Trent. Gene expression profile in multiple sclerosis patients and healthy controls: identifying pathways relevant to disease. *Human Molecular Genetics*, 12 (17):2191–2199, 2003.

W. Bruck. The pathology of multiple sclerosis is the result of focal inflammatory demyelination with axonal damage. *Journal of Neurology*, 252 (Suppl.5):v3–v9, 2005.

J.L. Cummings. Alzheimer disease. *New England Journal of Medicine*, 351(1):56–67, 2004.

G.H. Fernald, R.F. Yeh, S.L. Hauser, J.R. Oksenberg JR and S.E. Baranzini. Mapping gene activity in complex disorders: Integration of expression and genomic scans for multiple sclerosis. *Journal of Neuroimmunology*, 167(1-2):157–169, 2005.

K.C. Gunsalus, H. Ge, A.J. Schetter, D.S. Goldberg, J.D. Han, T. Hao , G.F. Berriz, N. Bertin, J. Huang, L.S. Chuang, N. Li, R. Mani, A.A. Hyman, B. Sonnichsen, C.J. Echeverri, F.P. Roth, M. Vidal, F. Piano. Predictive models of molecular machines involved in Caenorhabditis elegans early embryogenesis. *Nature*, 436(7052):861-865, 2005.

J.D. Han, D. Dupuy, N. Bertin, M.E. Cusick and M. Vidal. Effect of sampling on topology predictions of protein-protein interaction networks. *Nature Biotechnology*, 23(7):839–844, 2005.

P.R. Hiesinger and B.A. Hassan. Genetics in the age of systems biology. *Cell*, 123 -1173-1174, 2005.

J. Imitola, T. Chitnis, S.J. Khoury. Insights into the molecular pathogenesis of progression in multiple sclerosis: potential implications for future therapies. *Archives in Neurology*, 63(1):25–33, 2006.

H. Jeong, S.P. Mason, A.L. Barabási and Z.N. Oltvai. Lethality and centrality in protein networks. *Nature*, 411(6833):41–42, 2001.

H. Kitano, K. Oda, T. Kimura, Y. Matsuoka, M. Csete, J.Doyle and M. Muramatsu. Metabolic syndrome and robustness tradeoffs. *Diabetes*, Suppl 3: S6–S15, 2004.

H. Kitano and K. Oda. Robustness tradeoffs and host-microbial symbiosis in the immune system. *Molecular Systems Biology*, 2(1):msb4100039-E1–msb4100039-E10, 2006

H. Lodish, A. Berk, S.L. Zipursky, P. Matsudaira, D. Baltimore and J. Darnell. *Molecular Cell Biology (4th Ed.)*, W.H. Freeman and Company, New York, 2000.

J.C. Nacher, N. Ueda, T. Yamada, M. Kanehisa and T. Akutsu. Study on the clustering coefficients in metabolic network using a hierarchical framework. *International Workshop on Bioinformatics and Systems Biology*, Poster Abstracts:34–35, 2004

J.R. Oksenberg, S.E. Baranzini, L.F. Barcellos and S.L. Hauser. Multiple sclerosis: genomic rewards. *Journal of Neuroimmunology*, 113(2):171–184, 2001.

D.R. Rhodes and A.M. Chinnaiyan. Integrative analysis of the cancer transcriptome. *Nature Genetics*, 37(Suppl):S31–S37, 2005.

E. Scarpini, P. Scheltens, H. Feldman. Treatment of Alzheimer's disease: current status and new perspectives. *Lancet Neurology*, 2(9):539–547, 2003.

L. Steinman. Multiple sclerosis: a two-stage disease. *Nature Immunology*, 2(9):762–764, 2001.

M. Stetter, B. Schürmann, and M. Dejori. Systems level modeling of gene regulatory networks. In: *Artificial Intelligence Methods and Tools for Systems Biology*, W. Dubitzy and F. Azuaje (*Eds.*), Springer, The Netherlands, 2004.

R. Tanaka. Scale-rich metabolic networks. *Physical Review Letters*, 94(16):168101, 2005.

R. Tanaka, T.M. Yi and J. Doyle. Some protein interaction data do not exhibit power law statistics. *FEBS Letters*, 579(23):5114–5144. 2005.

P. Villoslada and J. Oksenberg. Neuroinformatics in clinical practice: are computers going to help neurological patients and their physicians?. *Future Neurology*, 1(2):1-12, 2006.

C. von Mering, M. Huynen, D. Jaeggi, S. Schmidt, P. Bork and B. Snel. STRING: a database of predicted functional associations between proteins. *Nucleic Acids Research*, 1(31):258–261, 2003.

C. von Mering, L.J. Jensen, B. Snel, S.D. Hooper, M. Krupp, M. Foglierini, N. Jouffre, M.A. Huynen and P. Bork. STRING: known and predicted protein-protein associations, integrated and transferred across organisms. *Nucleic Acids Research*, 1(33):D433–D437, 2005.

P.R. Walker, B. Smith, Q.Y. Liu, A.F. Famili, J.J. Valdes, Z. Liu and B. Lach. Data mining of gene expression changes in Alzheimer brain. *Artificial Intelligence in Medicine*, 31(2):137–154, 2004.

Y. Xia, H. Yu, R. Jansen, M. Seringhaus, S. Baxter, D. Greenbaum, H. Zhao and M. Gerstein. Analyzing cellular biochemistry in terms of molecular networks. *Annual Review of Biochemistry*, 73: 1051–1087, 2004.

Analytical and numerical results for entrainment in large networks of coupled oscillators.

Peter Grindrod*
*Lawson Software
Magdalen Centre, Oxford Science Park, Oxford, UK
Peter.Grindrod@Lawson.com

Zhivko Stoyanov†
†Department of Mathematical Sciences, University of Bath
Bath BA2 7AY
mapzvs@maths.bath.ac.uk

Abstract

In this paper we consider large networks of coupled oscillators. We choose to illustrate this using a general class of range dependent networks where the pairwise coupling is a probabilistic function of distance (range) between the nodes, and each node represents an oscillator with its own intrinsic phase and natural frequency of oscillation. Range dependent networks exhibit the "small world" phenomenon, being effectively superpositions of many networks each operating over different range lengths. We provide an asymptotic analysis in terms of a network coupling parameter that gives a simple analytic description of the coupled dynamics and which agrees well with numerical experiments.

1 Introduction

The emergent behaviour of populations of dynamical systems brought about by local pairwise (weak) coupling is of interest both from the point of view of the group dynamics and the theory and characterisation of the underlying networks. Perhaps the simplest nontrivial examples are coupled oscillatory systems where a local "diffusive" type of coupling gives rise to spatio-temporal patterns such as localised non planar waves, target patterns, or spiral waves. When the underlying network allows "longer range" couplings also even simple entrainment phenomena are not straightforward. By entrainment (or synchronisation) of a system of oscillators, we mean a state of the system, in which all oscillators move together as one with a possible difference in their phases, which remains constant for large time. This is a key concept in the understanding of self-organisation phenomena of coupled oscillators (see, for example, (K84)).

In (K75) Kuramoto considered networks of oscillators, in which the coupling between every pair of oscillators was identical. Although simple at a glance, his model was hard to analyse but due to his ingenious heuristics and assumptions, he was able to derive some properties about the system he considered.

In this paper we analyse entrainment in large networks of coupled oscillators (see, for example, (S00), (AS04)). We choose to illustrate this using a general class of range dependent networks where the pairwise coupling is a probabilistic function of distance (range) between the nodes, and each node represents an oscillator with its own intrinsic phase and natural frequency of oscillation. Range dependent networks exhibit the "small world" phenomenon, being effectively superpositions of many networks each operating over different range lengths (see, for example, (G02)). We provide an asymptotic analysis in terms of a network coupling parameter that gives a simple analytic description of the coupled dynamics and which agrees well with numerical experiments.

2 Twin coupled oscillators

First, consider the simplest case of two coupled oscillators:

$$\begin{aligned}\dot{\theta}_1 &= \lambda_1 + \varepsilon A_{12}\sin(\theta_2 - \theta_1)\\ \dot{\theta}_2 &= \lambda_2 + \varepsilon A_{21}\sin(\theta_1 - \theta_2)\end{aligned}$$

which has state space the torus with coordinates $\theta_i \mod (2\pi)$ for $i = 1, 2$. Here the A_{jk} are nonnegative coupling coefficients; ε is a nonnegative overall "strength" parameter to scale the coupling; and the $\lambda_i > 0$ represent the uncoupled frequencies of the separate oscillators. Setting $\phi = \theta_2 - \theta_1$ we obtain a single equation for the phase difference:

$$\dot{\phi} = \lambda_2 - \lambda_1 - \varepsilon(A_{12} + A_{21}) \sin\phi, \qquad (1)$$

which is integrable and so a closed form solution is available. However, qualitative information about the oscillation can be obtained directly from (1). First note that the frequencies become entrained for large time (with ϕ tending to a stable rest point) if and only if ε is such that

$$|\lambda_2 - \lambda_1| < \varepsilon(A_{12} + A_{21}).$$

If this condition does not hold one of the oscillators repeatedly "laps" the other.

3 N oscillators coupled via a directed graph

Let us generalise the above situation to N coupled oscillators. We shall think of them as vertices connected by a directed graph with entraining couplings defining the non negative weights of directed edges. Each oscillator is represented by a single phase variable, $\theta_i \mod (2\pi)$, having a natural, uncoupled frequency: whilst each coupling term, say from oscillator k acting on oscillator i, affects to increase or retard the rate of increase of the phase of oscillator i, so as to approach the phase of oscillator k. The state space for the full coupled system is an N dimensional torus with coordinates $\theta_i \mod (2\pi)$ for $i = 1, \ldots, N$. Specifically, we consider the following system on the N-torus:

$$\dot{\theta}_i = \lambda_i + \varepsilon \sum_{k=1}^{N} A_{ik} \sin(\theta_k - \theta_i), \quad i = 1, \ldots, N. \qquad (2)$$

Introduce the $n \times n$ coupling matrix A with zeros on the diagonal and jk^{th} component A_{jk}, which represents the weight of the coupling, or edge, from vertex j to vertex k. The parameter ε is a nonnegative overall "strength" parameter to scale the impact of A; and the $\lambda_i > 0$ represent the uncoupled frequencies of the separate oscillators.

Our interest is in whether and how the oscillators can become entrained with one another, for large time; producing a baulk oscillation, with their phases moving together, possibly separated by a constant set of phase shifts. Like the simple twin-oscillator case this behaviour depends upon the strength and nature of the couplings as well as the distribution of their natural frequencies.

3.1 No Baulk Oscillations for small ε

For any i and j we have

$$\dot{\theta}_i - \dot{\theta}_j = \lambda_i - \lambda_j +$$

$$\varepsilon \left(\sum_{k=1}^{N} A_{ik} \sin(\theta_k - \theta_i) - \sum_{k=1}^{N} A_{jk} \sin(\theta_k - \theta_j) \right).$$

The left hand side of this equation must vanish when oscillators i and j are entrained (that is when their phases differ by a constant amount through time). Set

$$\varepsilon^* = \max_{1 \le i,j \le N} \frac{|\lambda_i - \lambda_j|}{\sum_{k=1}^{N} (A_{ik} + A_{jk})}. \qquad (3)$$

Then if $\varepsilon < \varepsilon^*$, $\dot{\theta}_i = \dot{\theta}_j$ is impossible for at least one pair of oscillators and there can be no baulk oscillation. Note this condition is necessary and sufficient for no baulk oscillation to exist when $N = 2$.

3.2 Asymptotic Analysis of Baulk Oscillations for large ε

We seek an asymptotic solution, valid in the limit of large ε, representing a baulk oscillation, so that for some function, $\theta_0(t)$ say, we have

$\theta_i(t) = \theta_0(t) +$ an ε-dependent phase shift for oscillator i

for each $i = 1, \ldots, N$.

Setting $\vec{\theta}(t) = (\theta_1(t), \theta_2(t), \ldots, \theta_N(t))^T$, $\vec{\lambda} = (\lambda_1, \lambda_2, \ldots, \lambda_N)^T$ and $\vec{1} = (1, 1, \ldots, 1)^T \in \mathbb{R}^N$, we shall seek a solution which is in the form of a baulk oscillation (that is, all phases entrained) where the phase shifts are represented by a regular expansion in inverse powers of ε:

$$\vec{\theta}(t) = \theta_0(t)\vec{1} + \frac{1}{\varepsilon}\vec{\theta}_1 + \frac{1}{\varepsilon^2}\vec{\theta}_2 + \mathcal{O}\left(\frac{1}{\varepsilon^3}\right). \qquad (4)$$

Here $\vec{\theta}_1$ and $\vec{\theta}_2$ are vectors orthogonal to $\vec{1}$, so that the individual phase shifts are distinct from the baulk oscillation term.

Substituting (4) into (2) and expanding out the sine terms, we obtain

$$\dot{\theta}_0\vec{1} = \vec{\lambda} + \triangle\vec{\theta}_1 + \frac{1}{\varepsilon}\triangle\vec{\theta}_2 + \mathcal{O}\left(\frac{1}{\varepsilon^2}\right). \qquad (5)$$

Here $\triangle$ denotes the "Laplacian" matrix associated with the network coupling matrix A (replacing the zeroes on the diagonal of A with the negative of the corresponding row sums):

$$\triangle = A - \text{diag}(A\vec{1}).$$

The Laplacian matrix $\triangle$ contains information about the connected nature of the network: it is of huge importance in graph theory (B95). It is easy to see that zero is an eigenvalue of $\triangle$ with multiplicity equal to the number of distinct connected sub networks. Without loss of generality we shall assume zero is a simple eigenvalue - otherwise we may consider each connected sub network separately. In that case $\triangle\vec{1} = \vec{0}$.

Let e denote the corresponding left unit eigenvector: $e^T \triangle = 0^T$. Then pre-multiplying (5) with e^T we have

$$\dot{\theta}_0 e^T \vec{1} = e^T \vec{\lambda} + \mathcal{O}\left(\frac{1}{\varepsilon^2}\right), \qquad (6)$$

which determines $\theta_0(t)$. (In the case when $\triangle$ is a symmetric matrix the term $\mathcal{O}\left(\frac{1}{\varepsilon^2}\right)$ in the right hand side of (6) vanishes.) Then to $\mathcal{O}(1)$ and $\mathcal{O}\left(\frac{1}{\varepsilon}\right)$ we have $\vec{\theta}_1$ and $\vec{\theta}_2$ respectively, determined as the unique solutions, orthogonal to $\vec{1}$, of the matrix equations:

$$\left(\frac{e^T\vec{\lambda}}{e^T\vec{1}}\right)\vec{1} - \vec{\lambda} = \triangle\vec{\theta}_1, \qquad \triangle\vec{\theta}_2 = \vec{0}. \qquad (7)$$

First note that $\vec{\theta}_2 = \vec{0}$. Next, we may write

$$\vec{\theta}(t) = \left(t\left(\frac{e^T\vec{\lambda}}{e^T\vec{1}}\right) + C\right)\vec{1} + \frac{1}{\varepsilon}\vec{\theta}_1 + \mathcal{O}\left(\frac{1}{\varepsilon^3}\right), \quad (8)$$

where C is a constant, and $\vec{\theta}_1$ can be found by solving (7) in the subspace orthogonal to $\vec{1}$.

Hence by calculating e^T, the left eigenvector of $\triangle$ and solving for $\vec{\theta}_1$ from (7), we can use (8) to estimate the behaviour of the oscillators for large coupling parameter ε. In fact the experiments in the next section show that ε needs not be too large. Indeed, for values of ε not too much greater than ε^*, (8) provides an accurate representation of the behaviour of the system.

Finally, we note that for the network considered here the second eigenvalue of $\triangle$ is small (equalling -0.01528) with corresponding eigenvector $\vec{v}$, often called the Fiedler vector (F75). Hence $\vec{\theta}_1$ will typically be rich in the direction of $\vec{v}$. Now $\vec{v}$ is often used to explain certain network features (for example, clustering) and this suggests that the Fiedler vector $\vec{v}$ might also provide information to help understand different features in the solutions of (2).

4 Numerical Example

Example: We take $N = 100$, A a symmetric random range dependent matrix with values lying between zero and 0.96, and the λ_i as independent uniformly distributed random numbers within the interval $[0.5; 1.5]$. Then by direct calculation, $\varepsilon^* = 0.47884$.

In this case $e^T = \frac{1}{\sqrt{N}}\vec{1}$ and so we have from (6)

$$\dot{\theta}_0 = \frac{1}{N}\sum_{i=1}^{N}\lambda_i =: \hat{\lambda}.$$

Hence (8) gives

$$\theta_i(t) = \hat{\lambda}t + C + \frac{1}{\varepsilon}\theta_1^{[i]} + \mathcal{O}\left(\frac{1}{\varepsilon^3}\right),$$

where $\theta_1^{[i]}$ denotes the i^{th} component of $\vec{\theta}_1$, and

$$\theta_i(t) - \theta_j(t) = \frac{1}{\varepsilon}(\theta_1^{[i]} - \theta_1^{[j]}) + \mathcal{O}\left(\frac{1}{\varepsilon^3}\right). \qquad (9)$$

In Figure 1 we plot the phase differences, $\theta_i(t) - \theta_1(t)$ for $i = 2, \ldots, 100$, obtained directly from the numerical solution, for $t \in [0; 50]$, for various values of ε ($\varepsilon = 0.5, 0.6, 0.8, 2.0, 5.0, 10.0$).

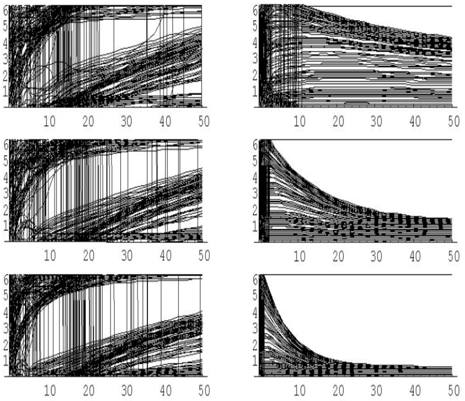


Figure 1: Plot of $\theta_i - \theta_1$, for $i = 2, \ldots, 100$, versus time t, for $\varepsilon = 0.5, 0.6, 0.8, 2.0, 5.0, 10.0$.

The entrainment as ε increases is clearly seen in Figure 1. Indeed, for $\varepsilon = 2.0$ the system settles to baulk oscillation before $t = 250$. In Figure 2 we compare the values of $\theta_i(t) - \theta_1(t)$ obtained by numerical solution with $\frac{1}{\varepsilon}(\theta_1^{[i]} - \theta_1^{[1]})$ in order to test the validity of (9), and hence the validity of the asymptotic analysis leading to equation (8). Clearly, even for ε not so large there is very good agreement between the asymptotic expression and numerical experiment, with the maximum error being around 1.3×10^{-4}.

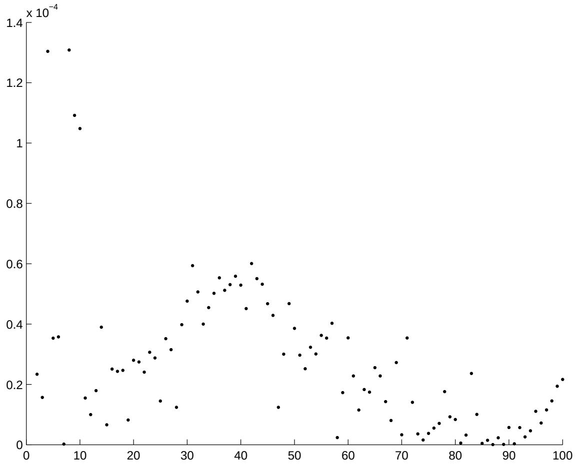


Figure 2: In this Figure we plot, for $i = 2, \ldots, 100$, the absolute value of the difference between $\theta_i(t) - \theta_1(t)$ (obtained by numerical solution of (2)) and $\frac{1}{\varepsilon}(\theta_1^{[i]} - \theta_1^{[1]})$, see (9). Here $\varepsilon = 2$ and $t = 250$.

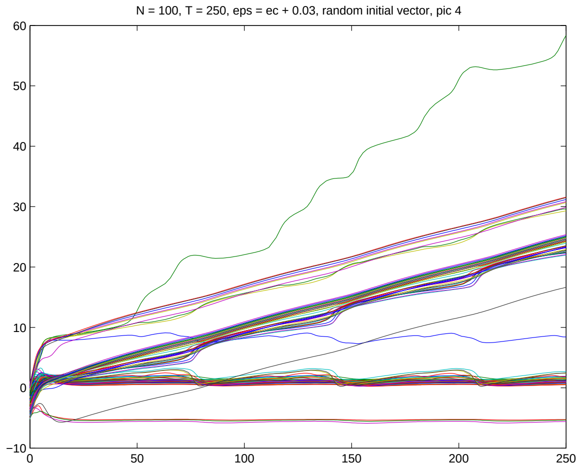


Figure 3: Plot of $\theta_i - \theta_{i_0}$ versus time, for $\varepsilon = \varepsilon^* + 0.03$.

Lastly, in Figure 3 we show the solution behaviour for the system with $\varepsilon = \varepsilon^* + 0.03$ for random starting values. This Figure represents the plot of the terms $\theta_i(t) - \theta_{i_0}(t)$, where $t \in [0; 250]$ and i_0 is such that $\lambda_{i_0} \leq \lambda_i$ for $1 \leq i \leq 100$. In our simulation in Figure 3 we observe that there are two clusters of oscillators entrained with θ_{i_0} and two other clusters which drift away from them. There is an “extreme” oscillator, which is not entrained to any of the groups, and two other oscillators, which seem to be attracted by the clusters of oscillators.

References

[AS04] D. Abrams and S. Strogatz, “Chimera States for Coupled Oscillators”, Phys. Rev. Let., 93, (17), (2004), 174102.

[B95] B. Bollobas, “Random Graphs”, Academic Press, New York (1995).

[F75] M. Fiedler, “A property of eigenvectors of nonnegative symmetric matrices and its applications to graph theory”, Czech. Math. Journal, 25, (1975), pp 619-633

[G02] P. Grindrod, “Range dependent random graphs and their application to modelling small world proteomic data sets”, Phys. Rev. E, 66, (2002), 0667021-0667027.

[K75] Y. Kuramoto, “Self-entrainment of a population of coupled nonlinear oscillators”, H. Araki (ed.) “International Symposium on Mathematical Problems in Theoretical Physics”, Lecture Notes in Physics, 39, 420-422, Springer, New York, (1975).

[K84] Y. Kuramoto, “Chemical Oscillators, Waves, and Turbulence”, Springer Verlag, Heidelberg, (1984)

[S00] S. Strogatz, “From Kuramoto to Crawford: exploring the onset of synchronization in populations of coupled oscillators”, Physica D 143 (2000) 1-20.

Scale-free structure is not the best for the shortest path length and the robustness

Yukio Hayashi[⋆†]
[⋆]Japan Advanced Institute of Science and Technology
1-1, Asahidai, Nomi, Ishikawa, Japan
yhayashi@jaist.ac.jp

Jun Matsukubo[†]
[†]Kitakyusyu National College of Technology
5-20-1 Shii, Kokuraminami, Fukuoka, Japan
jmatsu@kct.ac.jp

Abstract

We study geographical networks on a planar space, and show that the cutoff of degree improves the path length and the tolerance to failures and attacks. We also compare them with the randomly rewired non-geographical versions. These results are useful for constructing sensor or ad hoc networks.

1 Introduction

In complex network science, the topological structure called *small-world* or *scale-free* attracts interdisciplinary research fields, since it has been commonly found in many social, biological, and technological systems (Barabási, 2002). The heterogeneous structure with many low degree nodes and a few hubs has good properties in the meanings of economical and efficient communication by small number of hops in a connected network with a few links (Cancho and Solé, 2003) and the robustness against failures (Albert and Barabási, 2000). Moreover, the restriction of link lengths has been observed, e.g. Internet at both router and Autonomous System levels (Yook et al., 2002), road networks, and flight-connection in a major airline (Gastner and Newman, 2006) on geographical spaces. Recent studies of scaling relation between the path lengths and network size get much attention with the statistical physics approach.

In this paper, considering geographical networks for urban planning, electric circuits, distributed robots, sensor networks, communication networks, and so on, we investigate the effects of geographical structures on the path length and the robustness in a family of the planar networks: *Random Apollonian* and *Delaunay triangulation*. The planarity is important not only to avoid interference of the wireless beam, or to construct communication lines on the surface of earth, but also to design efficient routing methods taking into account the graph properties such as spanner. In particular, online routing algorithms (Bose and Morin, 2004) that guarantee delivery of messages using only local information about positions of the source, destination, and the adjacent nodes to a current node in the routing have been developed for planar networks.

2 Geographical Networks

2.1 Planar triangulation

Planar triangulation is a mathematical abstraction of sensor or ad hoc networks, in which the positions of nodes are temporarily fixed as base stations of backbone networks. Thus, the mobility of node is out of our scope to simplify the discussion. In computer science, it is well-known that Delaunay triangulation is the optimal planar triangulation in some geometric criteria (Imai, 2000), and widely used in practical applications for facility location and computer graphics. Moreover, there is an inclusion relation: nearest neighbor graph $\subset$ relative neighbor graph $\subset$ minimum spanning tree $\subset$ Gabriel graph $\subset$ Delaunay triangulation (Kranakis and Stacho, 2006). One of the fundamental techniques for equipping such properties is diagonal flipping. In a Delaunay triangulation, diagonal flips are globally applied to the triangles until the minimum angle of triangles is not increased by the exchange of links in a quadrilateral. Such global process is unsuitable for dynamically constructed networks. In contrast, a random Apollonian network can be generated by local procedures for the subdivision of a randomly chosen triangle at each time step in the evolution of network.

Thus, we investigate the communication efficiency measured by the average distance (defined by the sum of link lengths on a path) or hops on the optimal paths and the robustness of connectivity in the typical planar network models: random Apollonian network in complex network science, Delaunay triangulation in computer science, and a modification to bridge them.

2.2 Delaunay-like scale-free network

Although random Apollonian networks have the several advanced scale-free properties and the small-world effect with a small diameter of graph (Zhou et al., 2005), some long-range links naturally appear near the boundary edges. To reduce the long-range links, we propose a modified model (Hayashi and Matsukubo, 2005) as follows. The main idea is based on a strategy for connecting nodes in distances as short as possible by adding with the diagonal flips in a Delaunay triangulation.

Step 0: Set an initial planar triangulation in a space.

Step 1: Select a triangle at random and add a new node at the barycenter. Then, connect the new node to its three nodes. Moreover, by iteratively applying diagonal flips, connect it to the nearest node (or more than one of the neighbor nodes) within a radius defined by the distance between the new node and the nearest node of the chosen triangle.

Step 2: The above process is repeated until the required size N is reached.

We call our model RA+NN(one/all) that means the combination with the triangulation in Random Apollonian and the rewiring to the one or all Nearest Neighbors within a radius as the localization.

Fig. 1 illustrates the linking procedures by iterative diagonal flips: in a quadrilateral that consists of the shaded triangles, the long-range (crossing) link is diagonally exchanged to the red link for maximizing the minimum angle of triangles. The dashed lines are new links from the barycenter, and form new five triangles with contours in the left of Fig. 1; The intersected solid lines with dashed ones are removed after the 2nd flips.

Fig. 2 shows the topological characteristic that our model has the intermediate structure between random Apollonian networks and Delaunay triangulations. We can see a heterogeneous structure with dense and sparse parts: the dense-get-denser may be corresponded to the subdivision of a service area according to the increasing of population with preference of aggregation. As shown in Fig. 3, we find that the degree distributions follow a power-law: $k^{-\gamma_{RA}}$ in random Apollonian networks (marked by circles), log-normal: $\exp((\ln k - \mu)^2/2\sigma^2)$ in Delaunay triangulations (triangles), and power-law with exponential cutoff: $k^{-\gamma}\exp(-ak)$ in our models (pluses and crosses).

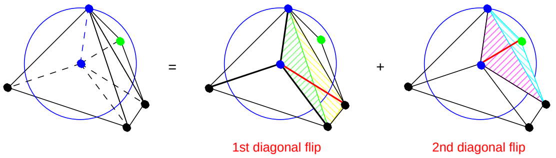


Figure 1: Linking procedures in a Delaunay-like scale-free network. The intersected lines are exclusive in each shaded quadrilateral.

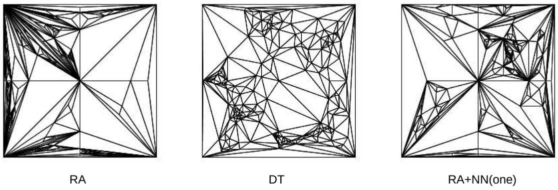

RA DT RA+NN(one)

Figure 2: Examples of the geographical networks. RA: random Apollonian, DT: Delaunay triangulation, and RA+NN: our Delaunay-like scale-free network.

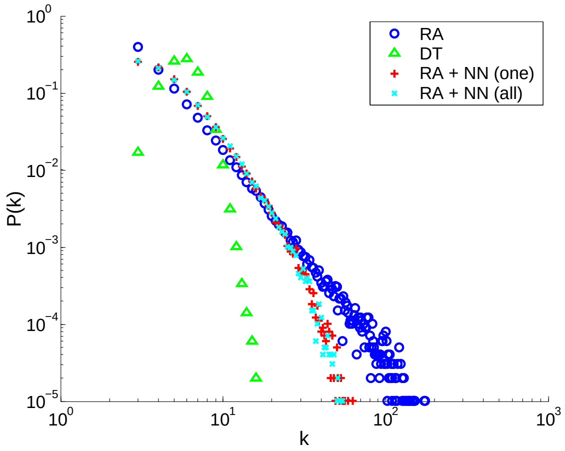


Figure 3: Degree distribution $P(k)$.

3 Path Length and Robustness

3.1 Weak disorder

In the studies of the optimal path in disordered complex networks (Braunstein et al., 2003; Kalisky et. al, 2005), each link length is associated with a weight assumed by $\exp(\delta\varepsilon)$, where the parameter δ controls the strength of disorder, and ε is a random number taken form a uniform distribution between 0 and 1. As a network approaches the strong disorder limit at $\delta \to \infty$, only the longest link becomes dominant in the shortest path length defined by the smallest sum of link lengths on a path between two nodes. At the limit, the scaling relations of the average shortest path length $\langle D \rangle \sim N^{1/3}$ for $\gamma > 4$ and $\langle D \rangle \sim N^{(\gamma-3)/(\gamma-1)}$ for $3 < \gamma \leq 4$ has been theoretically predicted (Braunstein et al., 2003) from the percolation on scale-free networks (Cohen et. al, 2002). Although the relation is unknown for $2 < \gamma \leq 3$ because of the singularity in the analysis at $\gamma = 3$, $\langle D \rangle \sim (\ln N)^{\gamma-1}$ has been also numerically suggested (Braunstein et al., 2003).

However, the assumption of length distribution may be violated on a geometric space, in addition the strong disorder limit is an extreme case. Thus, to investigate the strength of disorder in random Apollonian networks, Delaunay triangulations, and the proposed models, we compare the length distributions. Fig. 4 shows the distribution $P(l_{ij})$ of link length l_{ij} in each network. The dashed lines with an equal gap from top to bottom are corresponded to the distributions of weight $2\exp(\delta\varepsilon)/\exp(\delta)$ for $\delta = 1, 2, 4, 8, 16$, respectively. The factor $2/\exp(\delta)$ is due to the normalization for the maximum length of the boundary edges of the initial rectangle (see Fig. 2). We find that random Apollonian networks and RA+NN(one/all)s have weak disorder with small δ (Kalisky et. al, 2005), while Delaunay triangulations have a slightly broad range of disorder as similar to the exponential decay in the domestic airline flight-connection (Hayashi, 2006).

3.2 Weak small-world effect

We investigate the average distance of path length $\langle D \rangle$ on the shortest paths, the distance $\langle D' \rangle$ on the paths of the minimum hops, the average number of hops $\langle L \rangle$ on these paths, and the number of hops $\langle L' \rangle$ on the shortest paths between any two nodes in the geographical networks. The average means a statistical ensemble over the optimal paths in the above two criteria (w.r.t distance and hop) for networks in randomly generated 100 realizations at each size N. Figs. 5(a)(b) show that RA+NN(one) has the shortest distance and the intermediate number of hops in a weak small-world effect, which means the $\langle L \rangle$ is slightly larger than $O(\ln N)$ known as the effect in a scale-free network without geographical structure.

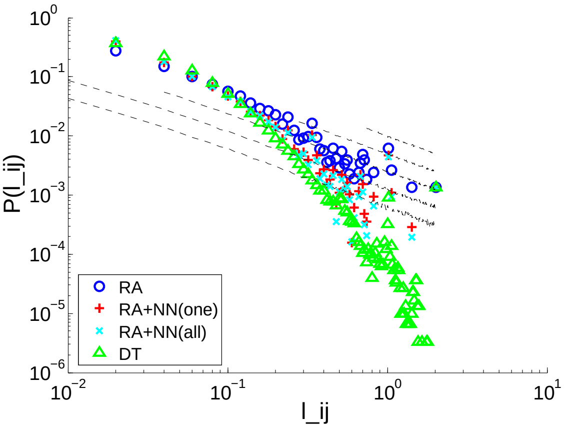


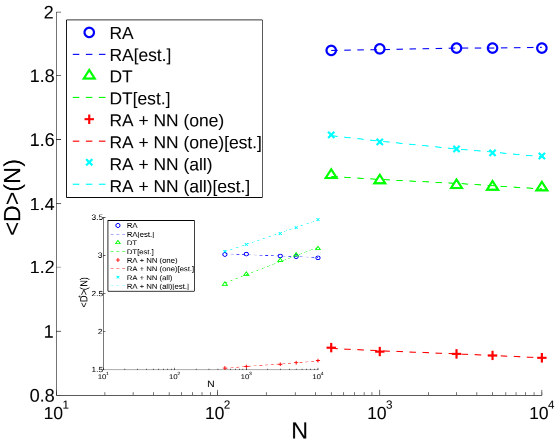
Figure 4: The distribution of link lengths with weak disorder.

Note that the shortest path and the path of the minimum hops may be distinct, these measures are related to the link cost or delay and the load for transfer of a message. It is better to shorten both the distance and the number of hops, however their constraints are generally conflicted, indeed, see Fig. 5.

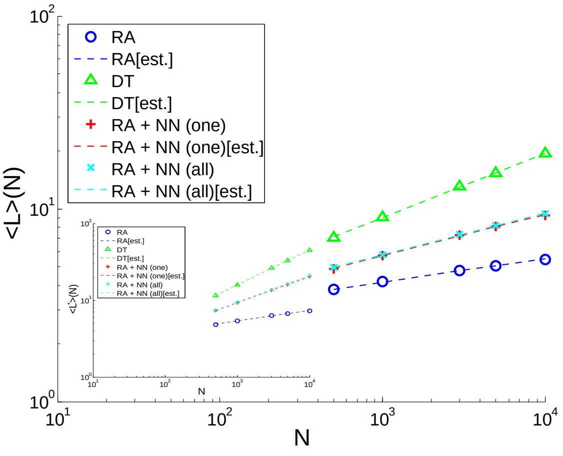
As shown in Table 1, we find the scaling relations. We remark that the values of β_d and $\beta_{d'}$ differ from $\gamma_{RA} - 1 \approx 2$ numerically suggested at the strong disorder limit (Braunstein et al., 2003), although the values of β_l and $\beta_{l'}$ are relatively close to it. In addition, the values of α_l and $\alpha_{l'}$ are close to $1/3$ predicted at the limit (Braunstein et al., 2003) for the Erdös-Rényi model as the classical random network and the Watts-Strogatz model as a small-world network. The nearest α_l in Delaunay triangulations is probably caused by that the lognormal degree distribution resembles the unimodal shapes in Erdös-Rényi and Watts-Strogatz models rather than a power-law.

Table 1: Estimated values of the exponents in the forms $\langle D \rangle \sim (\ln N)^{\beta_d}$, $\langle D' \rangle \sim (\ln N)^{\beta_{d'}}$, $\langle L \rangle \sim (\ln N)^{\beta_l}$, $\langle L' \rangle \sim (\ln N)^{\beta_{l'}}$, $\langle L \rangle \sim N^{\alpha_l}$, $\langle L' \rangle \sim N^{\alpha_{l'}}$ by a mean-square-error method.

model	β_d	β'_d	α_l	α'_l	β_l	β'_l
RA	0.012	-0.039	0.121	0.136	0.920	1.036
DT	-0.068	0.416	0.332	0.455	2.525	3.452
RA+NN (one)	-0.080	0.151	0.213	0.341	1.622	2.587
RA+NN (all)	-0.106	0.320	0.216	0.346	1.641	2.628



(a) average distance



(b) average num. of hops

Figure 5: The shortest distance and the intermediate number of hops in our model (marked by red pluses). The dashed lines correspond to the estimations in Table I. Insets show $\langle D' \rangle$ and $\langle L' \rangle$ on the paths of the minimum hops and the shortest, respectively.

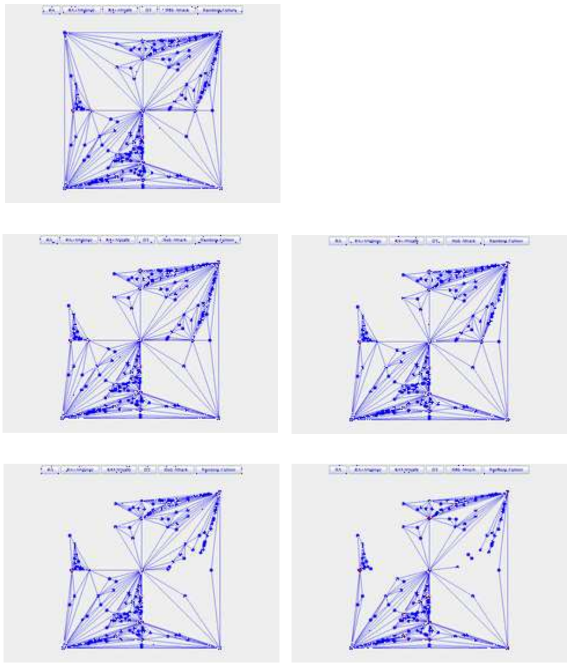
3.3 Tolerance to failures or attacks

The fault tolerance and attack vulnerability are known as the typical scale-free properties (Albert and Barabási, 2000), however the geographical effect on them are unknown. We compare the tolerance of connectivity in the giant component of the geographical and the non-geographical rewired networks with the same degree distribution (Maslov et al., 2004), when a small fraction f of the nodes is removed.

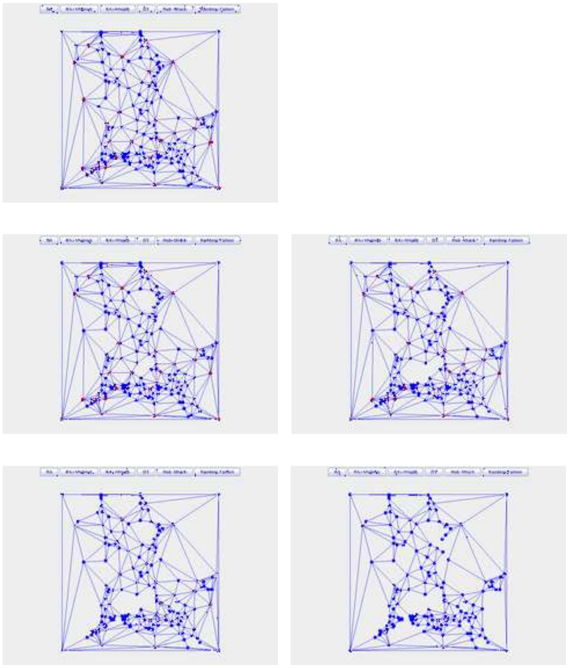
Figs. 6 and 7 show examples of random failures in the geographical networks at a small size $N = 200$ to visualize them. In the similar results, each initial component remains without isolated clusters. On the other hand, Figs. 8 and 9 show examples of targeted attacks to hubs. The random Apollonian network is the most vulnerable with many isolated clusters since the star-like stubs at the four corners and the center nodes of the initial rectangle are disconnected, while the Delaunay triangulation is relatively robust without such structure.

We investigate these differences quantitatively. The following results are obtained from the averages over 100 realizations at a size $N = 1,000$. We should remark that all networks have the same average degree $\langle k \rangle = 2(3N - 7)/N = 5.986$ and the minimum degree $k_{min} = 3$. Therefore, we investigate the tolerance at the same level with the total number of links $N \times \langle k \rangle / 2$. Fig. 10(a) shows the relative size S/N for the fraction of random failures in random Apollonian networks, Delaunay triangulations, and our models, where S denotes the size of giant component. Fig. 10(b) show the robustness of connectivity in the rewired networks, whose high tolerance is similar to Barabási-Albert model (Albert and Barabási, 2000) without geographical structure. As the geographical effect, it becomes weaker in the order of random Apollonian networks, RA+NN(one/all)s, and Delaunay triangulations with degree distributions from a pure power-law to the strong cutoff. These results are not contradictory to the theoretical prediction under the power-law degree distribution with exponential cutoff (Callaway et al., 2000), since the average degree $\langle k \rangle$ is not constant but smaller as the cutoff is stronger; the connectivity is weaker in sparse networks, however the corresponding strength of cutoff is in the inverse order of random Apollonian networks, RA+NN(one/all)s, and Delaunay triangulations.

Against the attack on hubs selected in the decreasing order of degrees, Figs. 11(a)(b) show the improvements in RA+NN(one/all)s from the extremely vulnerable random Apollonian networks. By the geographical effect, each network also becomes more vulnerable than the rewired version. In other words, the improvement by rewiring is consistent with recent results for an Inet-generated graph as a modeling of the Internet (Beygelzimer et al., 2005), although it includes another bias effect of removing links from higher degree nodes. Note that the weakly inhomogeneous Delaunay triangulation is different from a homogeneous random network, which has the same behavior against the failures and the attacks at a fraction of removed nodes (Albert and Barabási, 2000).

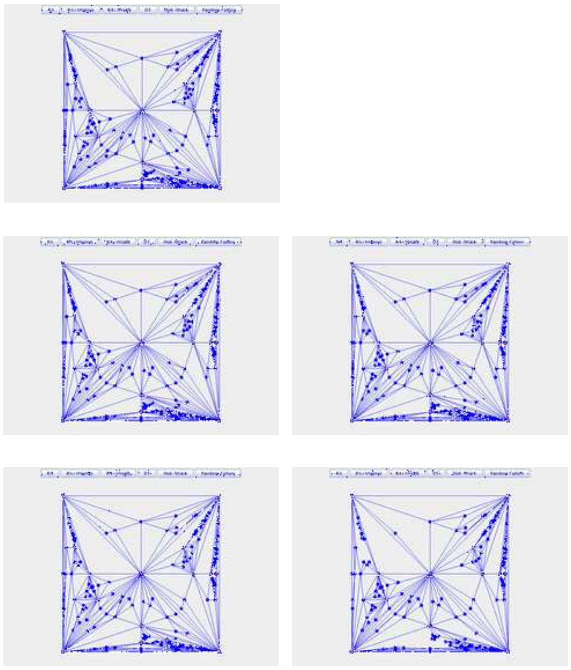


(a) RA

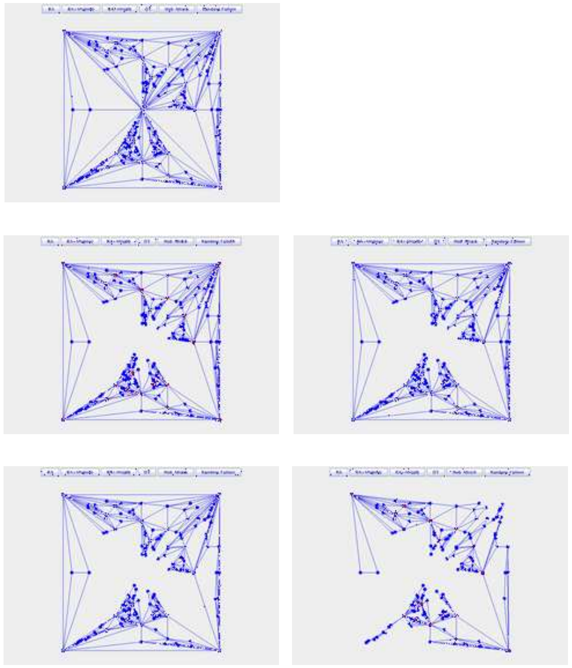


(b) DT

Figure 6: Progress of disconnection by random failures of 0, 4, 8, 16, 32 nodes from top-left to down-right for (a) random Apollonian network and (b) Delaunay triangulation.

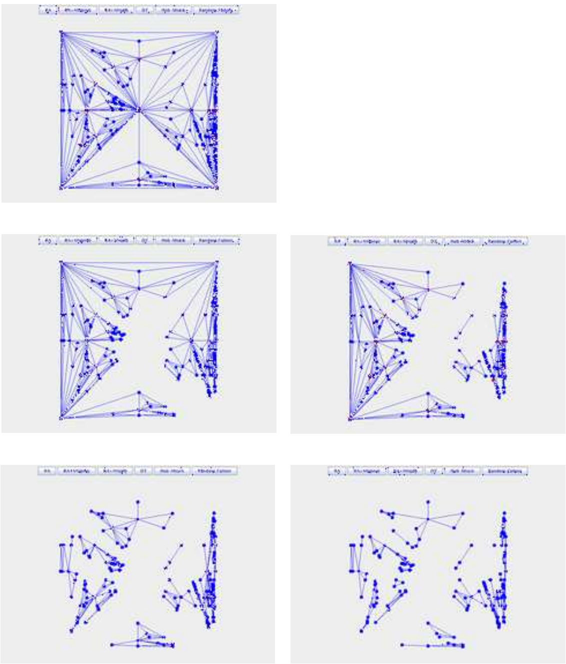


(a) RA+NN(one)

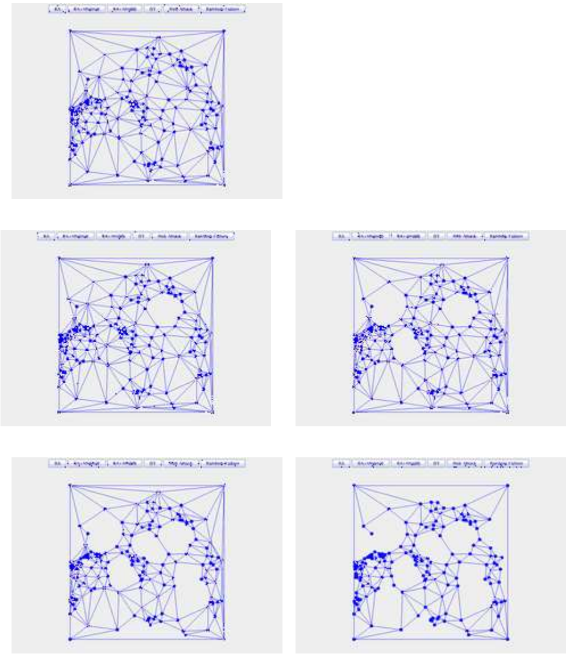


(b) RA+NN(all)

Figure 7: Progress of disconnection by random failures of 0, 4, 8, 16, 32 nodes from top-left to down-right for our models: (a) RA+NN(one) and (b) RA+NN(all).

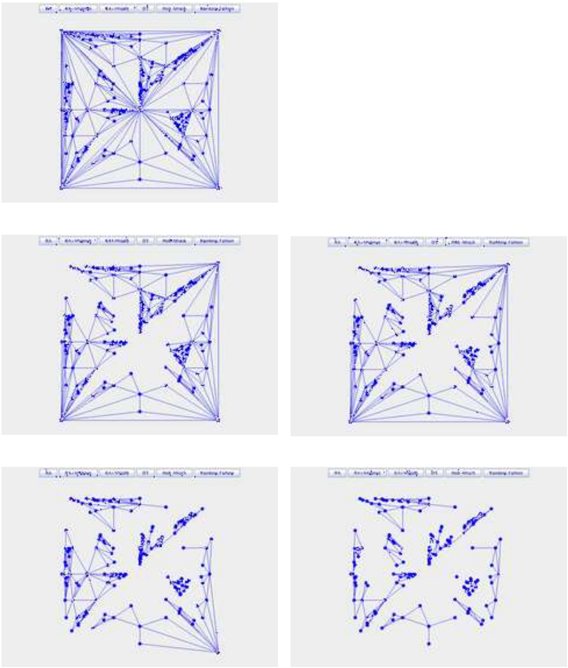


(a) RA

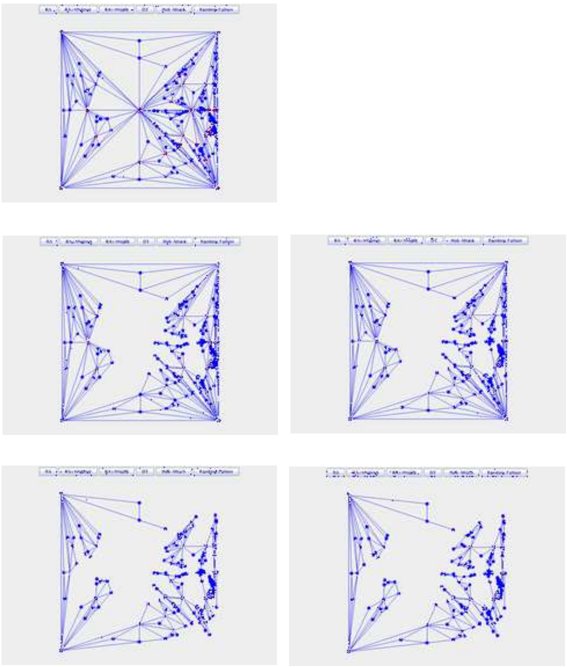


(b) DT

Figure 8: Progress of disconnection by targeted attacks on 0, 2, 4, 8, 16 nodes in decreasing order of degrees from top-left to down-right for (a) random Apollonian network and (b) Delaunay triangulation.

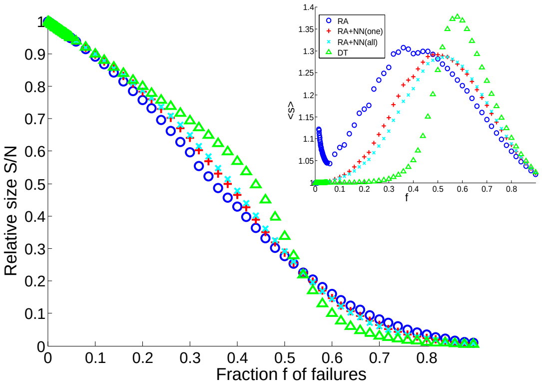


(a) RA+NN(one)

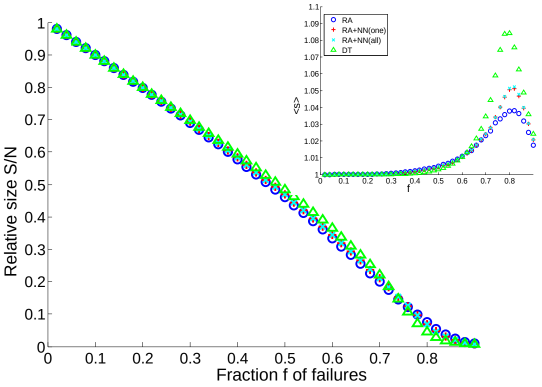


(b) RA+NN(all)

Figure 9: Progress of disconnection by targeted attacks on 0, 2, 4, 8, 16 nodes in decreasing order of degrees from top-left to down-right for our models: (a) RA+NN(one) and (b) RA+NN(all).

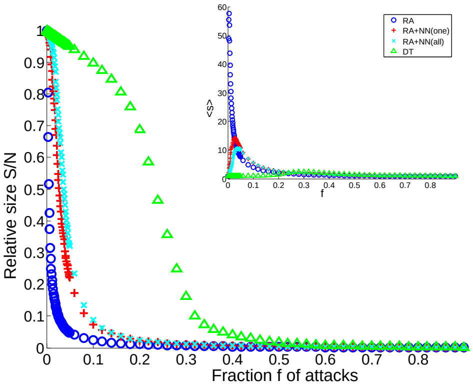


(a) geographical nets

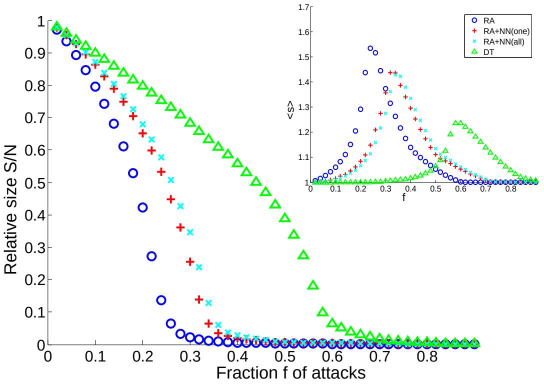


(b) randomly rewired nets

Figure 10: Relative sizes S/N of the giant component against random failures in the geographical and the rewired networks. Inset show the average size of isolated clusters except the giant component. At the peak, the giant component disappears.



(a) geographical nets



(b) randomly rewired nets

Figure 11: Relative sizes S/N of the giant component against attack on hubs in the geographical and the rewired networks.

4 Conclusion

We investigate the effect of geographical structure on the path length and the robustness of connectivity, focusing on a family of planar networks called random Apollonian network and Delaunay triangulation for communication systems. To reduce long-range links, we propose a modified model whose degree distribution follows a power-law with exponential cutoff. We find the weak disorder in the distributions of link lengths, and suggest the scaling relations of the shortest path length $\langle D \rangle \sim (\ln N)^{\beta_d}$ and of the minimum hop $\langle L \rangle \sim N^{\alpha_l}$ as similar to the case at the strong disorder limit (Braunstein et al., 2003). From the simulations, we conclude that random Apollonian networks have a path connected by a few hops but the path length becomes long including some long-range links, while Delaunay triangulations have a zig-zag path connected by many hops but each link is short. Instead of the superior geometric properties (Imai, 2000), Delaunay triangulations are no longer optimal in this criteria of the minimum hops. Our model is totally balanced: the shortest path length is the best, while the number of hops is the intermediate.

Moreover, we find that the tolerance to failures and attacks is weakened by the geographical effect. In particular, random Apollonian networks with a pure power-law degree distribution are extremely vulnerable. Although Delaunay triangulation is the most robust in these models, only it requires global configuration procedures that is unsuitable for ad hoc communication. Thus, there is a trade-off between the localization and the robustness. We will further investigate the above effect in more wide classes related to a family of scale-free networks.

References

R. Albert and A.-L. Barabási. Error and attack tolerance of complex networks. *Nature*, 406:378–382, 2000.

A.-L. Barabási. *Linked: The New Science of Networks*. Perseus, 2002.

Beygelzimer et al. Improving network robustness by edge modification. *Physica A*, 357:593–612, 2005.

P. Bose and P. Morin. Online routing in triangulations. *SIAM J. of Computing*, 33(4):937–951, 2004.

L.A. Braunstein et al. Optimal paths in disordered complex networks. *Physical Review Letters*, 91 (16):168701, 2003.

D.S. Callaway et al. Network robustness and fragility: Percolation on random graphs. *Physical Review Letters*, 85(25):5468–5471, 2000.

R.F.i Cancho and R.V. Solé. Optimization in complaex networks. In M. Rubi R. Pastor-Satorras and A. Diaz-Guilera, editors, *Statistical Mechanics in Complex Networks*, Chapter 6. Springer, Berlin, 2003. Lecture Notes in Physics **625**.

R. Cohen et al., Structural properties of scale-free networks. In S. Bornholdts and H.G. Shuster, editors, *Handbook of Graphs and Networks*, Chapter 4. Wiely-VCH, New York, 2002.

M.T. Gastner and M.E.J. Newman. The spatial structure of networks. *Eur. Phys. J. B*, 49(2):247–252, 2006.

E. Kranakis and L. Stacho. Routing and Traversal via Location Awareness in Ad Hoc Networks. In A. Boukerche, editor, *Handbook of Algorithms for Wireless Networking and Mobile Computing*, Chapter 8. Chapman & Hall/CRC, 2006.

S. Maslov, K. Sneppen and A. Zaliznyak. Detection of topological patterns in complex networks: correlation profile of the internet. *Physica A*, 333:529–540, 2004.

Y. Hayashi and J. Matsukubo. Scale-free networks on a geographical planar space for efficient ad hoc communication. In *Proc. of NOLTA*, 118–121, 2005.

Y. Hayashi, A Review of Recent Study of Geographical Scale-Free Networks. to appear in *IPSJ Journal*, Special Issue on Network Ecology, 47(3):2006 or *arXiv:physics/0512011*, 2005.

K. Imai. Structures of triangulations of points. *IEICE Trans. on Infor. and Syst.*, 83(3):428–437, 2000.

T. Kalisky, et al., Scaling of optimal-path-lengths distribution in complex networks. *Physical Review E*, 72: 025102, 2005.

S.-H. Yook, H. Jeong and A.-L. Barabási. Modeling the internet's large-scale topology. *PNAS*, 99(21): 13382–13386, 2002.

T. Zhou, G. Yan and B.-H. Wang. Maximal planar networks with large clustering coefficient and power-law degree distribution. *Physical Review E*, 71:046141, 2005.

A Discovery Method of Research Communities

Ryutaro Ichise[⋆]
[⋆]Intelligent Systems Research Division,
National Institute of Informatics
2-1-2 Hitotsubashi, Chiyoda-ku,
Tokyo 101-8430, Japan
ichise@nii.ac.jp

Hideaki Takeda[†]
[†] Research Center for
Testbeds and Prototyping,
National Institute of Informatics
2-1-2 Hitotsubashi, Chiyoda-ku,
Tokyo 101-8430, Japan
takeda@nii.ac.jp

Taichi Muraki[‡]
[‡]TriAx Corporation
4-29-3 Yoyogi, Shibuya-ku,
Tokyo, 151-0053 Japan
muraki@triax.jp

Abstract

Since research trends can change dynamically, researchers have to keep up with new research trends and undertake new research topics. Therefore, research communities for new research domains are important. In this paper, we propose a method to discover research communities. The key feature of our method is a network model of papers and a word assignment technique for the communities obtained. We show the performance of the proposed method using experiments with real world data.

1 Introduction

As information technologies progress, we can obtain research information faster then before. However, technologies covering a wide area can change just as rapidly. Therefore, all researchers must not only continuously follow new trends of research but also investigate new research topics. When we undertake new research topics, we need to know the research communities of researchers with the same research topic or same interest. As a result, we need an effective community mining method for finding them. In order to find research communities, we usually use bibliography information. The methods include co-citation analysis (Small, 1973; Chen and Paul, 2001) and bibliographic coupling (Kessler, 1963). Although these methods are very useful for analyzing research topics from the global viewpoint of all bibliography data, we cannot always understand what the discovered communities represent. CiteSeer (CiteSeer.IST, 2004) and Google Scholar (Google, 2004) are able to handle research communities from a micro viewpoint because they handle co-author and citation information from bibliographies and use the information for individual researchers. Although these systems are good for finding local communities involving an author, they are not suitable for finding research communities close to the author. Börner et al. (Börner et al., 2005) proposes to use co-author networks to find research communities by using weighted graphs. Their system uses heuristics to separate communities without interaction. Ichise et al. (Ichise et al., 2005) proposed a community mining method based on the interaction of users. Although the proposed system of Ichise et al. supports community mining for both a global view and a local view with several mining indexes, it does not identify the research topics of the communities obtained. In this paper, we propose a method to discover research communities with identified topics.

The present paper is organized as follows. In Section 2, we discuss the proposed method for research community mining. In Section 3, we describe the experimental evaluation of our method and then discuss the results. Finally, in Section 4, we present our conclusions.

2 Research Community Discovery

2.1 Network Model of Research Community

Although several network models using bibliographies to represent research communities have been proposed (Ichise et al., 2005), in this paper, we focus on the co-author relationships of a research paper to find the research communities. First, we assume a simple paper model. This model consists of keywords and author names. In this case, we can consider an author's work on a research topic by noting the keywords. As a result, authors who write a paper collaboratively share the same interest, represented by the keywords. If we consider the authors as nodes and the keywords as edges, we can represent the bibliog-

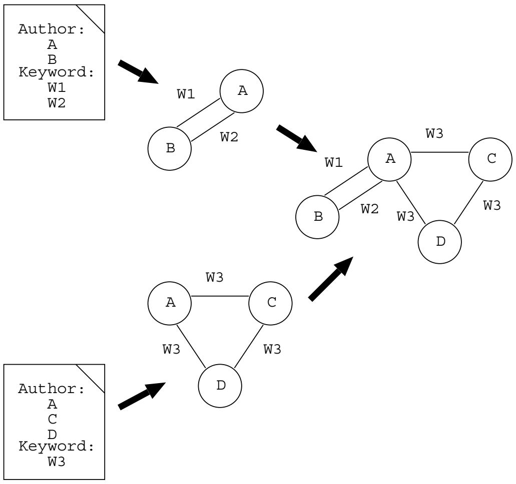


Figure 1: Network model of researchers.

raphy information as researcher networks.

Let us explain our model using an example. Assume that we have two papers, as shown on the left side of Figure 1. One was written by two authors, A and B, and has two keywords, W_1 and W_2. Another was written by three authors, A, C and D, and has the keyword W_3. We can compose graphs of the authors and edges from the two papers, as shown in the center of Figure 1. Then, the joint representation generated from the two bibliographies of the two papers is shown on the right side of Figure 1.

As one can see, we can obtain a labeled graph from the bibliography data with our modeling. Then, the next question is how do we discover research communities from this graph. We define a research community as a cluster which is densely connected by the same research interest or topic. Therefore, the research communities we want to obtain are clusters which have their edges labeled by the same keywords. Since our network model provides the research keywords on the edges, we can obtain the research communities by eliminating the edges of no interest to the system user. In other words, after the user specifies the research keywords, most of the edges which are not labeled by the specified keywords can be deleted. This process produces the research communities. For example, when the user specifies W_3 for the networks in Figure 1, the edges of W_1 and W_2 are eliminated. As a result, researcher B is isolated from the graph and we can find the research community consisting of researchers A, C and D.

2.2 Keyword Assignment for Communities

Since the clusters obtained by our method are only connected by user-specified relationships, we can consider each cluster as a research community. However, each cluster does not have its own property or identification. In other words, if the user does not have enough knowledge about the researchers, the user may not understand the meaning of the communities because there is no information about them. In order to solve this problem, we propose a method of assigning keywords for each obtained community.

In our paper model, the papers written by the authors in each community have keywords. If some words appear often in such papers, we can consider these words as a property for the community. However, if we simply counted the occurrences of the keywords in these papers, the relationships between keywords would be lost. In order to avoid this problem, we consider frequent keywords as units of the papers. The algorithm is follows:

1. Select papers, which are written by the authors in the community, from a paper database. Note that the papers are selected for each user. For example, if a paper is written by two authors in the same community, the paper is selected twice.

2. The selected papers are analyzed by the Apriori algorithm (Agrawal and Srikant, 1994). In this process, the keywords in a paper are treated as an item, and the papers are treated as transactions.

As a result, we can obtain word pairs for each community. We assign those word pairs as the property of the community.

3 Experiments

3.1 Bibliography Data

In order to evaluate our method, we conducted experiments using actual bibliography data. In the present study, we used the CiNii database (NII, 2004) to obtain bibliography information. We used 128,000 records, 90,000 records, 358,000 records, and 519,000 records for the paper, researcher, author and co-author, respectively. The author entries denote the number of authors for each paper. For example, the record is 3 when three researchers write a paper collaboratively. The co-author entries denote the number of combinations of authors for a paper. For example, the record is counted as ${}_4C_2 = 6$ for

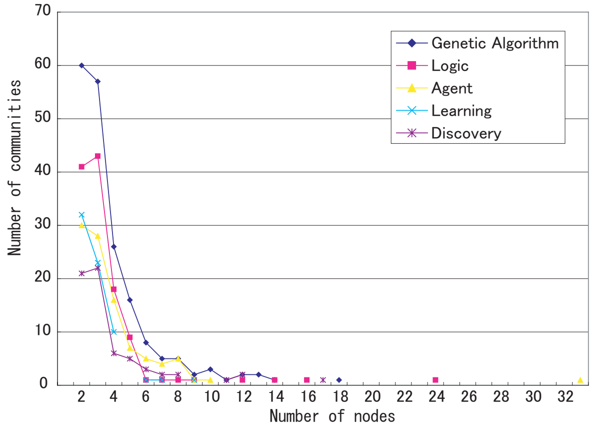


Figure 2: Number of discovered communities.

a paper when the paper is written by four authors. It was necessary to have keywords for our paper model. Therefore, we used the words in the title as keywords.

3.2 Experimental Results

The co-author network tends to have large clusters. In fact, the network constructed by all the bibliography data consists of a few large clusters and many small clusters. The number of clusters for all the data is shown in Table 1.

However, our method can successfully split a large cluster into readable research communities. We utilized five words to show the discovered communities. The five words are as follows: genetic algorithm, logic, agent, learning and discovery. We counted the number of nodes and clusters for each word. Figure 2 illustrates the result. The horizontal axis denotes the number of nodes and the vertical axis denotes the number of communities. As you can see from Figure 2, our method successfully discovers readable-sized communities. In addition, communities related to particular topics of interest to the user can be mined by our method.

Next, in order to evaluate our method for qualitative aspect, we analyzed the communities obtained by our method. The communities were constructed using the word ”Discovery”. Although our method discovered many communities, we selected five communities shown in Figure 3. Since our bibliography data mainly included papers written in Japanese, the communities obtained also consisted of Japanese researchers. The assigned keywords for each community are shown in Table 2. The longest pair of keywords, which frequently appeared in the top three, were selected for the table.

Table 2: Keywords obtained for the topic ”discovery”.

Community ID	Keywords
No. 1	{discovery} {algorithm} {special issue}
No. 2	{Japanese poem, similarity} {poem, similarity, extraction} {English sentence, technology}
No. 3	{heuristics, method} {database, heuristics} {knowledge, exception, discovery}
No. 4	{definition, occurrence, lambda calculus} {logic, program} {unification, extension}
No. 5	{*th, workshop} {scientific, discovery} {*th, report}

Community No. 2, No. 3, No. 4 and No. 5 represent research groups in universities. Most of the assigned words for the communities are valid. However, our method assigns meaningless words such as ”workshop” and ”report”. In our future work, we plan to develop a method to suppress the assignment of meaningless words. We believe such words can be identified by a simple method of retrieving stop words. Community No. 1 is the largest community in Figure 3. Since Prof. Setsuo Arikawa at Kyushu University is one of the most famous Japanese scientists in the ”discovery” domain, he bridged several communities. For example, the bottom left part of the community is the research community at Kyushu University, and the upper left part of the community is a community in machine learning. As a result, our system assigns general words for this community such as ”discovery” and ”algorithm.” In addition, the system cannot generate a long pair of words for this community. In our future work, we will develop a method to identify a person who is a bridge between different communities. Incidentally, the word ”special issue” was assigned in Community No. 1 because the community members edit a special issue for a journal.

4 Conclusion

In this paper, we propose a research community mining method. The key feature of our research is the modeling of papers and researchers. This modeling enables us to eliminate the edges in large clusters.

Table 1: Distribution of clusters for all the bibliography data.

Number of Nodes	76375	39	28	27	26	...	10	9	8	7	6	5	4	3	2	1
Number of Clusters	1	1	1	1	1	...	31	39	57	64	143	214	328	646	1054	7173

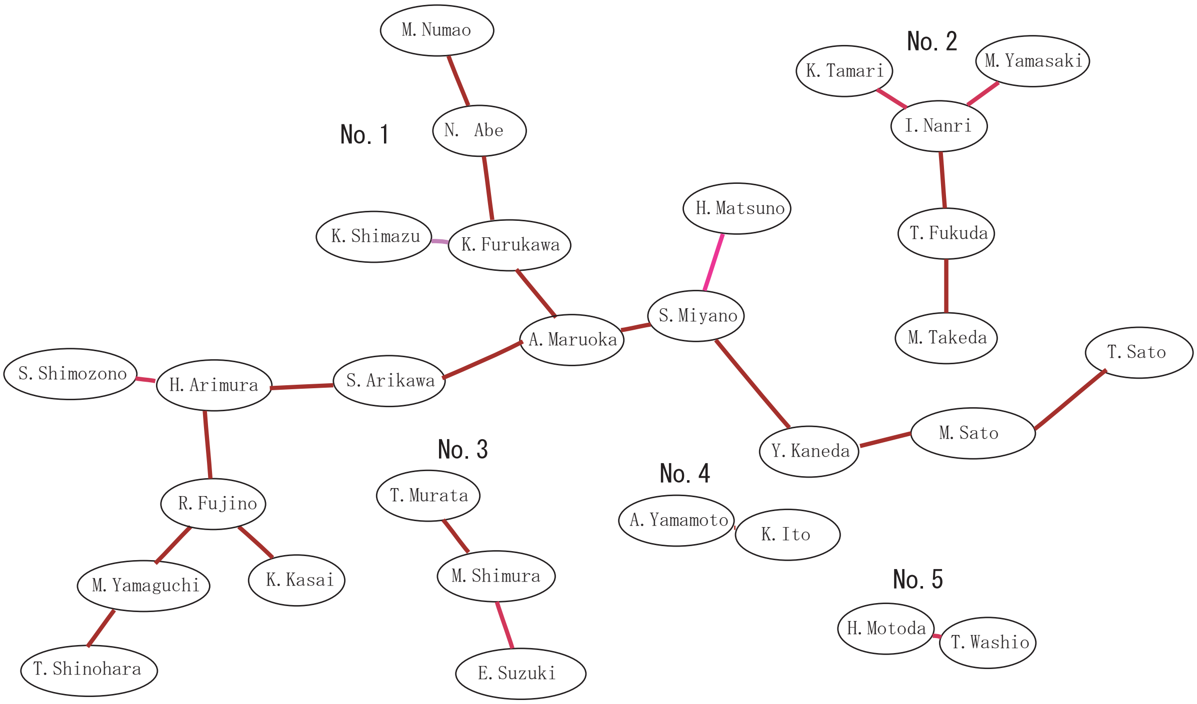


Figure 3: Research communities of discovery in Japan.

In addition, the modeling can also help to retrieve communities for particular topics. We also propose a method to assign a word to each cluster. We implemented our method and show how to investigate bibliography data with our system. The experimental results show that the performance of our method looks promising.

References

Rakesh Agrawal and Ramakrishnan Srikant. Fast algorithms for mining association rules. In *Proc. 20th Int. Conf. Very Large Data Bases, VLDB*, pages 487–499. Morgan Kaufmann, 1994.

Katy Börner, Luca Dall'Asta, Weimao Ke, and Alessandro Vespignani. Studing the emerging global brain: Analyzing and visualizing the impact of co-authorship teams. *Complexity*, 10(4):58–67, 2005.

Chaomei Chen and Ray J. Paul. Visualizing a knowledge domain's intellectual structure. *Computer*, 34 (3):65–71, 2001.

CiteSeer.IST. Scientific literature digital library, 2004. http://citeseer.ist.psu.edu/.

Google. Google scholar, 2004. http://scholar.google.com/.

Ryutaro Ichise, Hideaki Takeda, and Kosuke Ueyama. Community mining tool using bibliography data. In *Proceedings of the 9th International Conference on Information Visualization*, 2005.

M. M. Kessler. Bibliographic coupling between scientific papers. *American Documentation*, 14(1): 10–25, 1963.

NII. Cinii (citation information by national institute of informatics), 2004. http://ci.nii.ac.jp/.

Henry Small. Co-citation in the scientific literature: A new measure of the relationship between two documents. *Journal of the American Society of Information Science*, 24:265–269, 1973.

Clustering Coefficients for Weighted Networks

Gabriela Kalna[†]
[⋆]Department of Mathematics,
University of Strathclyde,
Glasgow G1 1XH, UK.
`aas04105@maths.strath.ac.uk`

Desmond J. Higham[⋆]
[†]Department of Mathematics,
University of Strathclyde,
Glasgow G1 1XH, UK.
`aas96106@maths.strath.ac.uk`

Abstract

The clustering coefficient has been used successfully to summarise important features of unweighted, undirected networks across a wide range of applications. Recently, a number of authors have extended this concept to the case of networks with non-negatively weighted edges. After reviewing various alternatives, we focus on a definition due to Zhang and Horvath that can be traced back to earlier work of Grindrod. We give a natural and transparent derivation of this clustering coefficient and then analyse its properties. One attraction of this version is that it deals directly with weighted edges and avoids the need to discretise, that is, to round weights up to 1 or down to 0. This has the advantages of (a) retaining all edge weight information, and (b) eliminating the requirement for an arbitrary cutoff level. Further, the extended definition is much less likely to break down due to a 'divide-by-zero'. Using our new derivation and focussing on some special cases allows us to gain insights into the typical behaviour of this measure. We then illustrate the idea by computing the generalised clustering coefficients, along with the corresponding weighted degrees, for pairwise correlation gene expression data arising from microarray experiments. We find that the weighted clustering and degree distributions reveal global topological differences between normal and tumour networks.

1 Introduction

Many complex data sets have natural representations as networks. Summarising, comparing, categorising and modelling these data sets are important activities that are taking place simultaneously across a wide range of disciplines (Newman, 2003). It is accepted that typical real-life networks are neither random graphs in the classical Erdös-Rényi sense nor regular lattices (Watts and Strogatz, 1998). Various quantities can be computed in order to characterise a network; most prominently the concepts of *pathlength*, *degree* and *clustering coefficient* have proved extremely useful.

Watts and Strogatz (1998) coined the phrase *small world network* to describe the commonly occurring situation where a sparse network is highly clustered (like a regular lattice) yet has small pathlengths (like a random graph). Since that landmark paper, many complex networks have been analysed and labelled as small worlds.

Similarly, the so-called *scale–free* property of the degree distribution, (Barabasi and Albert, 1999; Newman, 2003), has become accepted as a hallmark of many real data sets, although there is now some doubt as to its true prevalence (Khanin and Wit, to appear; Pržulj et al., 2004).

Both the small world and scale–free properties have been widely studied for unweighted, or binary, undirected networks. In the case of more general weighted edges it is of course possible to create a binary network by normalising, imposing a cutoff and rounding to 0 and 1 (Rougemont and Hingamp, 2003). However, it is our tenet that the original weights should be respected where possible. Recently, a number of authors have attempted to generalise the clustering coefficient concept to the case of weighted edges (Barrat et al., 2004; Lopez-Fernandez et al., 2004; Onnela et al., 2005; Zhang and Horvath, 2005), producing a range of possible definitions.

We present here a natural and transparent derivation of a clustering coefficient for weighted graphs. The resulting definition coincides with those in (Grindrod, 2002; Zhang and Horvath, 2005) and hence we argue for the use of this Grindrod-Zhang-Horvath clustering coefficient as a generalised measure of clustering. We believe that this measure, along with the corresponding weighted degree distribution, gives an informative high-level picture that can be used for classifying, comparing and modelling weighted networks, just as in the unweighted case. We do some analysis to provide insight into the usefulness of this clustering coefficient, and then show some results for gene expression microarray data.

Many methods for microarray data analysis monitor differences in the expression of genes under various experimental conditions: normal/tumour (Chen et al., 2002), multiclass cancers (Golub et al., 1999; Ramaswamy et al., 2001), treatment/survival (Segal, 2005). Pair-wise gene expression correlation has long been used to predict relationships between genes. Recently, gene co-expression networks have emerged (Stuart et al., 2003; Zhang and Horvath, 2005) connecting genes with high correlation. However, despite the fact that genome-wide gene expression data sets are available, their full potential is often not used and information from only a subset of genes, usually with highest variation, is extracted. Hence, we view these weighted networks as ideal candidates on which to apply the new clustering coefficient framework.

Using available microarray data we construct two distinct gene coexpression networks that represent normal and tumour states. We examine weighted clustering coefficients and weighted degree distributions of these networks with the aim of finding tumour-related differences. We emphasize that our aim is to characterize overall network topology rather than to categorize individual genes or samples.

The rest of this article is organised as follows. In section 2 we start with the binary definition of clustering coefficient and list some generalisations that have been proposed for weighted networks. In section 3 we give a natural derivation that leads to the Grindrod-Zhang-Horvath definition, and show how this can be easily computed via matrix products. We then use some simple examples to explore the properties of this coefficient. In section 4 we give some realistic computations on pairwise correlation networks arising from microarray data.

2 Clustering Coefficient and its Generalisations

Consider an undirected graph with normalised weights $0 \leq w_{ij} \leq 1$ between nodes i and j. In the binary case $w_{ij} \in \{0,1\}$ the clustering coefficient, or curvature, for node k is defined as

$$\mathrm{clust}(k) := \frac{t}{v(v-1)/2}, \qquad (1)$$

where v is the number of immediate neighbours of node k, and t is the number of triangles incident to node k (Rougemont and Hingamp, 2003; Watts and Strogatz, 1998). In words, $\mathrm{clust}(k)$ answers the question "given two nodes that are both connected to node k, what is the likelihood that these two nodes are connected to each other?" It is straightforward to see that the definition breaks down when $v < 2$, that is, node k has less than two immediate neighbours, and otherwise $0 \leq \mathrm{clust}(k) \leq 1$.

Recently, a few different extensions of the clustering coefficient to the general weighted case have emerged. In Lopez-Fernandez et al. (2004) the weighted clustering coefficient for node k is defined as

$$\mathrm{wclust}_{\mathrm{LF}}(k) := \frac{\sum_{i \neq j \in N(k)} w_{ij}}{v(v-1)},$$

where the term $\sum_{i \neq j \in N(k)} w_{ij}$ can be seen as the total weight of relationship in the neighbourhood $N(k)$ of node k.

Barrat et al. (2004) introduced a measure of clustering that combines topological information with the weight distribution of the network

$$\mathrm{wclust}_{\mathrm{B}}(k) := \frac{1}{s(v-1)} \sum_{i,j} \frac{(w_{ki} + w_{kj})}{2} a_{ik} a_{kj} a_{ij}.$$

Here $s = \sum_j w_{kj}$ denotes the weighted degree of node k and a_{ij} is an element of the underlying binary adjacency matrix. The normalisation factor $s(v-1)$ ensures that $0 \leq \mathrm{wclust}_{\mathrm{B}}(k) \leq 1$. This definition of weighted clustering coefficient considers only weights of edges adjacent to node k but not the weights of edges between neighbours of the node k.

Onnela et al. (2005) took into account weights of all edges: adjacent to node k and between-neighbours. They considered weights $0 \leq w_{ij} \leq 1$ and replaced the number of triangles t in (1) with the sum of triangle intensities

$$\mathrm{wclust}_{\mathrm{O}}(k) := \frac{2 \sum_{i,j} (w_{ik} w_{kj} w_{ij})^{1/3}}{v(v-1)}.$$

We remark that the three clustering coefficient definitions above suffer from the drawback that they require an underlying binary network to be generated; if this is not available as a separate set of data, then presumably it must be obtained by discretizing the weighted edges. Hence, as in the case where the original binary definition is used for weighted networks (Rougemont and Hingamp, 2003), they are dependent upon some thresholding parameter. Further, they break down in the case where the number of binary neighbours, ν, is less than 2.

A definition that uses only the network weights was proposed by Zhang and Horvath (2005)

$$\mathrm{wclust}_{\mathrm{HZ}}(k) := \frac{\sum_{i\neq k}\sum_{j\neq i, j\neq k} w_{ki}w_{ij}w_{jk}}{(\sum_{i\neq k} w_{ki})^2 - \sum_{i\neq k} w_{ki}^2}. \quad (2)$$

The numerator in (2) was obtained by finding a lower bound for the denominator, this ensuring that $\mathrm{wclust}_{\mathrm{HZ}}$ is in the range $[0, 1]$.

We also mention that rather than one clustering coefficient per node, Schank and Wagner (2005) presented a single weighted clustering coefficient for the whole network as

$$\mathrm{wclust}_{\mathrm{S}} := \frac{1}{\sum_v w(v)} \sum_v w(v)c(v).$$

Here $c(v)$ is a clustering coefficient for node v and $w(v)$ a weight function. One of possible choices of weight function is the weighted degree.

3 Weighted Clustering

3.1 Definition and Properties

Consider now an undirected weighted network of M nodes that is fully connected with weights $0 \leq w_{ij} = w_{ji} \leq 1$ between nodes i and j and $w_{ii} = 0$. Some simple algebra allows the binary clustering coefficient (1) to be rewritten as

$$\mathrm{wclust}(k) := \frac{\sum_{i=1}^{M}\sum_{j=1}^{M} w_{ki}w_{kj}w_{ij}}{\sum_{i=1}^{M}\sum_{j=1, j\neq i}^{M} w_{ki}w_{kj}}. \quad (3)$$

This formula directly extends to the real value case where $w_{ij} \geq 0$ and hence gives a natural definition for weighted networks. We also mention that the same formula was used in Grindrod (2002) in the context where w_{ij} represents the probability of an edge between nodes i and j in a random network model. Closer inspection shows that the formula (3) has a simple interpretation that is analogous to that of the binary case:

$$\tfrac{1}{2}\sum_{i=1}^{M}\sum_{j=1}^{M} w_{ki}w_{kj}w_{ij}$$

is a reasonable measure of how many “big triangles” involve node k and

$$\tfrac{1}{2}\sum_{i=1}^{M}\sum_{j=1 j\neq i}^{M} w_{ki}w_{kj}$$

says how many “big pairs of neighbours” there are. It is easy to verify that (3) retains the property $0 \leq \mathrm{wclust}(k) \leq 1$.

Computationally, note that the numerator of (3) is

$$\begin{aligned} \tfrac{1}{2}\sum_{i=1}^{M} w_{ki} \sum_{j=1}^{M} w_{kj}w_{ij} &= \tfrac{1}{2}\sum_{i=1}^{M} w_{ki}(W^2)_{ki} \\ &= \tfrac{1}{2}(W^3)_{kk} \end{aligned}$$

and the denominator is

$$\begin{aligned} \tfrac{1}{2}\Big(\sum_{i=1}^{M}\sum_{j=1}^{M} w_{ki}w_{kj} \quad &- \quad \sum_{i=1}^{M} w_{ki}^2\Big) \\ &= \tfrac{1}{2}\big((e^T w_k)^2 - \|w_k\|_2^2\big). \end{aligned}$$

Here, $(W^p)_{ij}$ denotes the (i, j) element of the pth power of W, w_k denotes the kth row (or column) of W and e denotes the vector with all elements equal to one. Hence, a neater representation of (3) is

$$\mathrm{wclust}(k) = \frac{(W^3)_{kk}}{(e^T w_k)^2 - \|w_k\|_2^2}, \quad (4)$$

which shows that the weighted clustering coefficient can be computed across all nodes in $O(M^3)$ operations. The formula (4) also makes it clear that (3) is entirely equivalent to the Zhang-Horvath definition (2).

Having derived this definition from what we believe to be a natural and informative viewpoint, we now attempt to gain further insights by focussing on particular types of weighted network.

3.2 Limit Forms of Clustering

We now zoom to a particular node K of a graph and explore its weighted clustering coefficient (3) in specific cases. Starting with a binary network $w_{ij} \in \{0, 1\}$ we replace zero weights with a small weight $0 < \epsilon << 1$ (weak connections) and unit weights with $1 - \epsilon$ (strong connections). Thus, we are dealing with fully connected graph.

(A) In the first case, let node K have $m > 1$ strong and $n > 1$ weak connections to other nodes in the graph. Then there are (a) $m(m-1)/2$ strong-strong, (b) mn strong-weak and (c) $n(n-1)/2$ weak-weak neighbour pairs. Let there be r, s and u strong edges between neighbours in cases (a), (b) and (c) respectively. It is easy to show that equation (3), for $\epsilon \rightarrow 0$, results in $\mathrm{wclust}(K) = 2r/m(m-1)$. In words, r strong triangles are built over $m(m-1)/2$ strong neighbour pairs. Thus the weighted clustering coefficient (3) approaches the binary value (1).

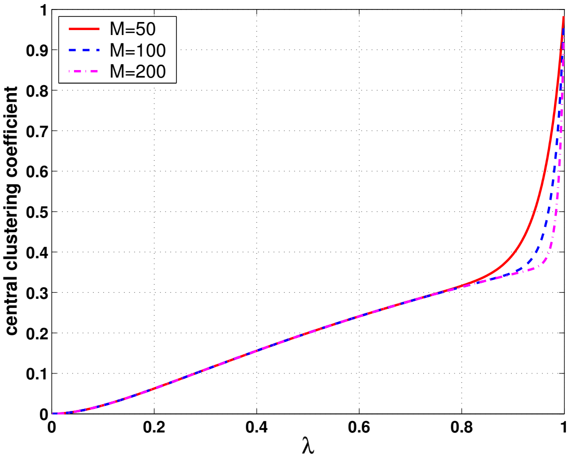


Figure 1: Clustering coefficient of the central node in the weighted graph defined by (5).

(B) In the second case we consider the marginal setting $v = 1$: node K has strong connection, $1 - \epsilon$, only to one node P and n weak, ϵ, connections to all other nodes in a complete graph. Then n out of all possible neighbour pairs involve the strong edge between nodes K and P and $n(n-1)/2$ pairs are formed by n weak edges adjusted to node K. Between-neighbour edges will be again strong or weak. Let there be r strong edges with one end in node P and s strong edges between "weakly" connected neighbour nodes of node K. Then from (3) we get $\mathrm{wclust}(K) = (r\epsilon(1-\epsilon)^2 + (n-r)\epsilon^2(1-\epsilon) + s\epsilon^2(1-\epsilon) + (n(n-1)/2 - s)\epsilon^3)/(n\epsilon(1-\epsilon) + n(n-1)\epsilon^2/2)$. This expression results in r/n for $\epsilon \to 0$. In words, we can get to r out of n "weakly" connected neighbours of the node K through the strong edge KP and strong edges connecting node P with these r nodes. It is clear that $\mathrm{wclust}(k) = 1$ only if $r = n$, that means there is a strong edge between P and all nodes weakly connected to K. Because $\mathrm{wclust}(k) = 0$ if $r = 0$, the strong edge between nodes K and P is the only edge involving node P. That means this edge would be separated from the graph in the corresponding discretised network.

Case **B** reveals an important advantage of the generalised definition (3). It continues to provide useful information in the small ϵ regime where *any discretization process based on thresholding to a binary network would result in $v = 1$ and hence an undefined clustering coefficient in (1).*

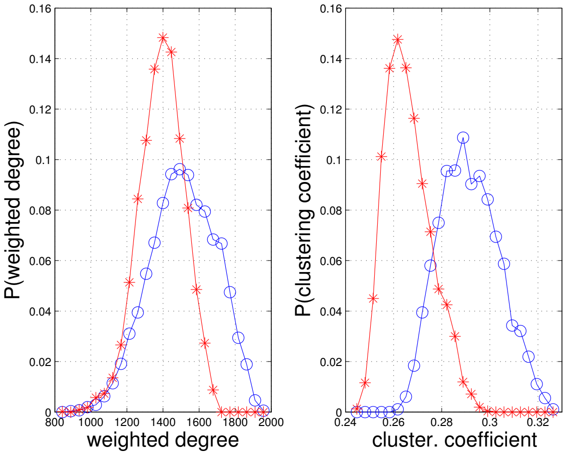


Figure 2: Probability of weighted degree (left) and curvature (right). Breast cancer: normal (circles) and tumour (stars).

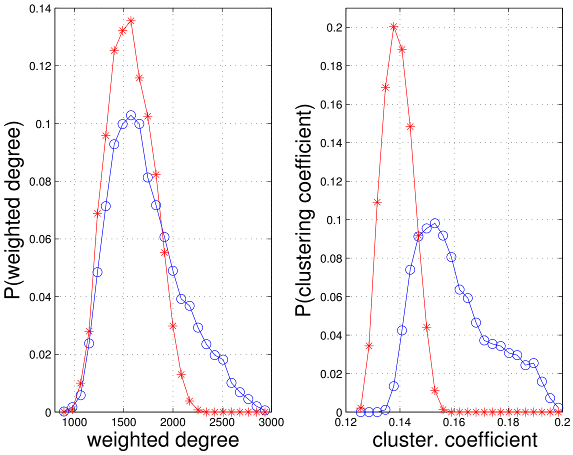


Figure 3: Probability of weighted degree (left) and clustering coefficient (right). Liver cancer: normal (circles) and tumour (stars).

3.3 Uniform Connectivity

Another case where the clustering coefficient simplifies arises when node K has equal weights with all other nodes: $w_{Kj} = \text{constant}$ for all $j \neq K$. In this case we have

$$\mathrm{wclust}(K) = \frac{\sum_{i=1}^{M}\sum_{j=1}^{M} w_{ij}}{(M-1)(M-2)} \approx \frac{\sum_{i=1}^{M}\sum_{j=1}^{M} w_{ij}}{M^2},$$

and we see that $\mathrm{wclust}(K)$ then reflects the average connectivity between the other nodes in the network.

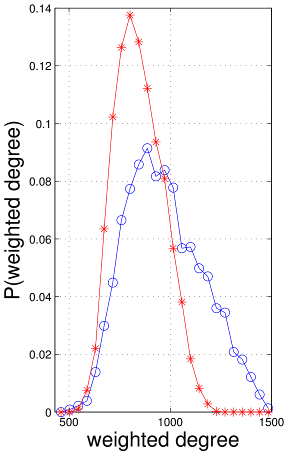


Figure 4: Probability of weighted degree (left) and clustering coefficient (right). Lymphoma: normal (circles) and tumour (stars).

3.4 Range Dependent Weights

The concept of a *range-dependent weighted random graph*, or RENGA, was introduced and analyzed by Grindrod (2002) and further studied by Higham (2005). We may adapt this idea to the case of non-random range-dependent weights. Suppose that the nodes are ordered $1, 2, 3, \ldots, M$ and that the connectivity weight decays as a function of lattice distance. To be specific, we let

$$w_{ij} = w_{ji} = \lambda^{|i-j|}, \tag{5}$$

for some $\lambda \in [0, 1]$. At one extreme, $\lambda \approx 0$, there are no edges after discretising to a binary network, and hence the traditional clustering coefficient is undefined. At the other extreme, $\lambda = 1$, all edges are present after discretising to a binary network, and hence the traditional clustering coefficient is 1 for each node. In Figure 1 we use networks of size $M = 50, 100, 200$ and compute the generalised clustering coefficient (3) for the central node, $k = \frac{1}{2}M$, as λ ranges from 0 to 1. Note that the definition (3) makes sense for any $\lambda > 0$. We see that the clustering coefficient approaches the value zero as λ approaches zero from above; this is perfectly reasonable behaviour. Further as λ increases away from zero, the clustering coefficient monotonically increases, and it matches the binary value of 1 at $\lambda = 1$. Overall, the generalised version provides a natural, informative interpolation of the classical clustering coefficient.

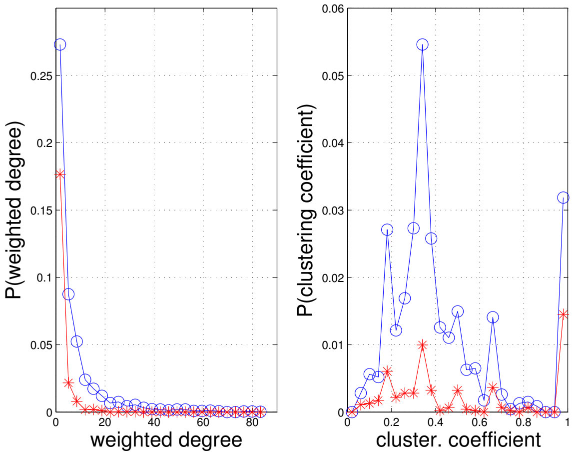


Figure 5: Probability of weighted degree (left) and clustering coefficient (right). Threshold 0.8. Lymphoma: normal (circles) and tumour (stars).

4 Microarray Illustration

We now examine the distribution of the clustering coefficient (3) in practice, along with that of the corresponding weighted degree, using pairwise correlation networks arising from cDNA microarray data. Most importantly, we would like to explore differences in character of weighted degree and clustering coefficient distributions of two different networks: normal and tumour.

The initial gene expression data arising from cDNA microarray experiments is a rectangular $M \times N$ matrix A of log-transformed ratios a_{ij} of $i = 1, \ldots, M$ genes in a set of $j = 1, \ldots, N$ samples. We consider the Pearson correlation

$$\mathrm{cor}(i, j) = \frac{\sum_{k=1}^{N}(a_{ik} - \mu_i)(a_{jk} - \mu_j)}{\sigma_i \sigma_j},$$

where μ_i and σ_i are respectively the mean and the standard deviation of gene i log-ratios, as a measure of similarity between the gene expression profiles. We define pairwise gene similarity weights $w_{ij} = |\mathrm{cor}(i, j)|$, for $1 \le i, j \le M$, with $w_{ij} = w_{ji} \in [0, 1]$ and $w_{ii} = 0$. A large weight w_{ij} indicates that genes i and j are highly co-expressed (or anti-expressed). In this representation $M \times M$ matrix W denotes the symmetric weight matrix encoding the strength of connection between pairs of genes.

Aware of the fact that different number of genes as well as samples in data sets can affect values of correlation and consequently distort comparisons of both weighted degrees and clustering coefficients,

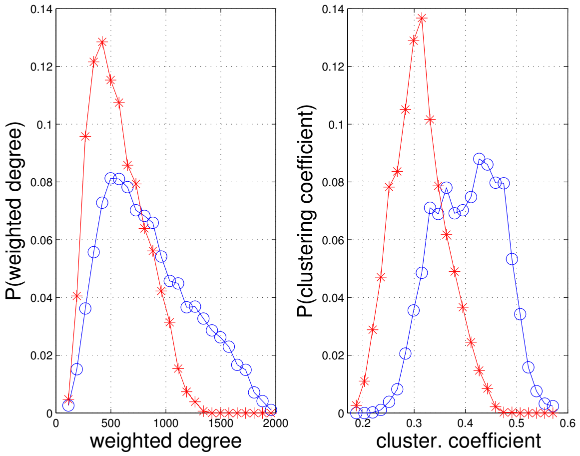


Figure 6: Probability of weighted degree (left) and clustering coefficient (right). P value 0.05. Lymphoma: normal (circles) and tumour (stars).

we looked for data consisting of the same number of normal and tumour samples for the same set of genes. In this experiment we used cDNA microarray data for normal and tumour tissues, taken from (Choi et al., 2005) and downloaded from http://centi.kribb.re.kr/MMA/index.html. Data processing performed by the authors included filtering of genes with more than 70% missing values or less than 4 observations, UniGene mapping, and imputation of missing values. The original data can be downloaded from the Stanford Microarray Database.

We selected data sets with more than ten samples in normal and tumour subsets. We present three of the results in this paper: breast cancer (5603 genes, 13 samples), liver cancer (12065 genes, 76 samples) lymphoma (4615 genes, 31 samples). Figure 2 shows the distribution of the weighted clustering coefficient (right), and also the distribution of the weighted degree (left) arising from breast cancer data. Circle-line and star-line represent the distributions of normal and tumour networks respectively. Figure 3 and Figure 4 show results for liver cancer and lymphoma.

We emphasize that our aim is to study the 'big-picture' issue of overall network topology, as opposed to the 'fine-detail' issue of clustering individual genes and/or samples (Kluger et al., 2003). The figures reveal global topological differences between the two networks. In general the tumour samples give rise to smaller and more peakily distributed clustering and degree. Degree ranges of normals and tumours start from a similar value but the degree range of tumours is narrower. Large numbers of genes in normal samples show a high degree of connection to other genes. Differences in clustering coefficient distributions are more striking. Distribution ranges of normal and tumour networks only partly overlap: most genes in normal networks have higher correlation than any gene in tumour networks.

Given that the weighted clustering coefficient produces interesting results, it is pertinent to ask whether careful thresholding to a discretised binary network (Rougemont and Hingamp, 2003) can also reproduce these findings. Clearly there is a whole parameterized family of such binary networks. In particular, high thresholding may exclude interesting features of the networks. For example, when weights above the threshold of 0.8 are re-set to 1 and the remaining weights are re-set to zero, the clustering coefficient and weighted degree distributions could not reveal the differences observed from original networks; see Figure 5.

For a more systematic approach, P values may be used to decide on significance of correlation. Even in this case, however, somewhat arbitrary thresholds must be imposed. For the lymphoma networks, suppose we take the view that correlations ≥ 0.355 are significant (corresponding to $P \leq 0.05$) and correlations ≥ 0.456 are highly significant (corresponding to $P \leq 0.01$). This would mean that only 18% (12%) of all possible edges are significant and 8% ($< 5\%$) are highly significant in the normal (tumour) lymphoma network, so that a large amount of data is being discarded. (Of course, there are computational benefits from introducing sparsity, but for the network sizes in these experiments this is not a significant issue.) In Figure 6 we plot data for the $P \leq 0.05$ binary networks. Comparing with Figure 4, we see that very similar topology is revealed. This allows us to conclude that the parameter-free weighted clustering coefficient approach is not affected by insignificant or "by chance" values, and automatically produces results consistent with the P value version.

5 Summary

Our aim here was to argue that out of the possible ways that have been proposed to generalise the clustering coefficient to the case of a weighted network, there is one very promising candidate; namely the Grindrod-Zhang-Horvath version (Grindrod, 2002; Zhang and Horvath, 2005). We gave a natural derivation and illustrated its behaviour on specific classes of network. Particular advantages of the definition are:

- It is a true generalisation, collapsing smoothly to the binary case when edge weights tend to $\{0, 1\}$ values.

- It can provide meaningful results in cases where any type of binary thresholding produces breakdown.
- It reveals natural topological properties of real networks, and can do this without the need to specify parameters or discard potentially useful data.

Acknowledgements

Both authors are supported by EPSRC grant GR/S62383/01.

References

A. Barabasi and R. Albert. Emergence of scaling in random networks. *Science*, 286:509–512, 1999.

A. Barrat, M. Barthélemy, R. Pastor-Satorras, and A. Vespignani. The architecture of complex weighted networks. *PNAS*, 101:3747–3752, 2004.

X. Chen, S.T. Cheung, S. So, S.T. Fan, C. Barry, J. Higgins, K.M. Lai, J. Ji, S. Dudoit, I.O. Ng, M. Van De Rijn, and P.O. Botstein, D. Brown. Gene expression patterns in human liver cancers. *Molecular Biology of the Cell*, 13(6):1929–1939, 2002.

J.K. Choi, U. Yu, O.J. Yoo, and S. Kim. Differential coexpression analysis using microarray data and its application to human cancer. *Bioinformatics*, 21: 4348–4355, 2005.

T.R. Golub, D.K. Slonim, P. Tamayo, C. Huard, M. Gaasenbeek, J.P. Mesirov, H. Coller, M.L. Loh, J.R. Downing, M.A. Caliguri, C.D. Bloomfield, and E.S. Lander. Molecular classification of cancer: class discovery and class prediction by gene expression monitoring. *Science*, 286:531–537, 1999.

P. Grindrod. Range-dependent random graphs and their application to modeling large small-world proteome datasets. *Physical Review E*, 66:066702, 2002.

D. J. Higham. Spectral reordering of a range-dependent weighted random graph. *IMA Journal of Numerical Analysis*, 25:443–457, 2005.

R. Khanin and E. Wit. How scale-free are gene networks? *Computational Biology*, to appear.

Y. Kluger, R. Basri, J.T. Chang, and M. Gerstein. Spectral biclustering of microarray data: coclustering genes and conditions. *Genome Research*, 13 (4):703–716, 2003.

L. Lopez-Fernandez, G. Robles, and J.M. Gonzalez-Barahona. Applying social network analysis to the information in cvs repositories. *In Proc. of the 1st Intl. Workshop on Mining Software Repositories (MSR2004)*, pages 101–105, 2004.

M.E.J. Newman. The structure and function of complex networks. *SIAM Review*, 45:167–256, 2003.

J.-P. Onnela, J. Saramki, J. Kertsz, and K. Kaski. Intensity and coherence of motifs in weighted complex networks. *Phys. Rev. E*, 71(6):065103, 2005.

N. Pržulj, D.G. Corneil, and I. Jurisica. Modeling interactome: Scale-free or geometric? *Bioinformatics*, 20(18):3508–3515, 2004.

S. Ramaswamy, P. Tamayo, R. Rifkin, S. Mukherjee, C.-H. Yeang, M. Angelo, C. Ladd, M. Reich, E. Latulippe, J.P. Mesirov, T. Poggio, W. Gerald, M. Loda, E.S. Lander, and T.R. Golub. Multiclass cancer diagnosis using tumor gene expression signatures. *PNAS*, 98(26):15149–15154, 2001.

J. Rougemont and P. Hingamp. DNA microarray data and contextual analysis of correlation graphs. *BMC Bioinformatics*, 4(15), 2003.

T. Schank and D. Wagner. Approximating clustering coefficient and transitivity. *J. of Graph Algorithms and Applications*, 9(2):265–275, 2005.

M.R. Segal. Microarray gene expression data with linked survival phenotypes: Diffuse large-b-cell lymphoma revisited. *Biostatistics*, 2005.

J.M. Stuart, E. Segal, D. Koller, and S.K. Kim. A gene-coexpression network for global discovery of conserved genetic modules. *Science*, 302(5643): 249–255, 2003.

D. Watts and S. Strogatz. Collective dynamics of small-world networks. *Nature*, 393:440–442, 1998.

B. Zhang and S. Horvath. A general framework for weighted gene co-expression network analysis. *Statistical Applications in Genetics and Molecular Biology*, 4, 2005.

Scale-free Paradigm in Yeast Genetic Regulatory Network Inferred from Microarray Data

Kevin Kontos⋆
⋆ULB Machine Learning Group
Computer Science Department
Université Libre de Bruxelles
1050 Brussels – Belgium
http://www.ulb.ac.be/di/mlg/
kkontos@ulb.ac.be

Gianluca Bontempi†
†ULB Machine Learning Group
Computer Science Department
Université Libre de Bruxelles
1050 Brussels – Belgium
http://www.ulb.ac.be/di/mlg/
gbonte@ulb.ac.be

Abstract

A major challenge of computational biology is the inference of genetic regulatory networks and the identification of their topology from DNA microarray data. Recent results show that scale-free networks play an important role in this context. These networks are characterized by a very small number of highly connected and relevant nodes, and by numerous poorly connected ones. In this paper, we experimentally assess the predictive power of the scale-free paradigm in a supervised learning framework. The hypotheses we intend to test in this framework are: (i) regulatory genes are effective predictors of the expression of the genes they regulate; (ii) a subset of regulatory genes may explain most of the variability of the measures. More precisely, we use the expression levels of a subset of regulatory genes, returned by feature selection, as input of a learning machine which has to predict the expression levels of the target genes. We will show that (i) each gene can be predicted by a small subset of regulatory genes, and (ii) on a global scale, a small subset of regulatory genes, called the hubs, can have a non-negligible predictive power on all the target genes. Also, most of the regulatory genes returned by the application of this approach to Gasch et al. (2000) data were identified by Segal et al. (2003) and form a biologically coherent set of genes.

1 Introduction

The *inference* of genetic regulatory networks from DNA microarray data is one of the major challenges in systems biology. A critical issue in network inference is the identification of the *network topology* from noisy data. Recent results tend to show that *scale-free networks* play an important role in systems biology (Jeong et al., 2000; Barabási and Oltvai, 2004; Barabási et al., 2004), notably for the yeast *Saccharomyces cerevisiae* model organism (Farkas et al., 2003; van Noort et al., 2004). These networks are characterized by a very small number of highly connected and relevant nodes, called the *hubs*, and by numerous poorly connected ones.

In this paper, we experimentally assess the role of the scale-free paradigm in a *supervised learning* approach to network inference in the case of the *Saccharomyces cerevisiae* organism. The idea is that relevant genes should emerge as good *predictors* in a multi-input multi-output supervised learning approach where the inputs are the *regulatory genes*[1] and the outputs are the *target genes* (Zhou et al., 2004). In particular, we will show that (i) for each gene, few other genes have an important predictive power, and (ii) a few genes (the hubs) have an important predictive power on *all* the target genes.

The main contributions of the paper are described hereafter. First, a *supervised learning framework* for network inference is introduced in Sect. 2. This means that the dependency between genes is estimated by the predictive power that the *expression levels* of a set of regulator genes have on the expression levels of some target genes. Also, because of the large ratio between the number of genes and the number of experimental conditions, we propose a *feature selection* strategy based on the Gram-Schmidt (GS) orthogonalization procedure (Stoppiglia et al., 2003). This procedure generates an *individual variable rank-*

---

[1] Transcriptional regulatory genes, also known as *regulator genes* or simply *regulators*, produce transcription factors (TF), which are regulatory proteins that regulate non-coding DNA segments (so called TF binding motifs) of target genes and initiate the transcription process.

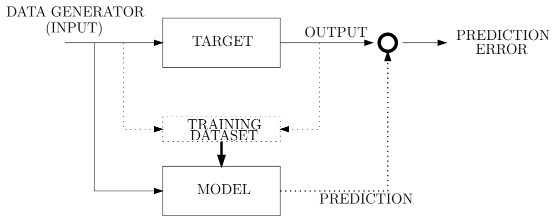


Figure 1: Supervised learning setting.

ing for each target gene. The first genes of each of these rankings constitute the small subsets of genes to be used as *predictors* in the supervised learning problem. According to the scale-free paradigm, only a small subset of genes plays a crucial role and these genes are thus expected to be well ranked in all the individual rankings. In order to assess this hypothesis, an *aggregated ranking* is built by "averaging" the individual rankings.

The experimental session relies on the dataset described in Gasch et al. (2000). The predictive power of the selected genes is assessed via a *cross-validation* strategy (Hastie et al., 2001) for a conventional *linear* model (D'haeseleer et al., 1999).

2 Supervised Learning Framework for Network Inference

Let us represent a DNA microarray dataset by a $N \times n$ matrix E, where N is the number of samples, n is the number of genes, and $E[C_j, G_i] = expr_{C_j}^{G_i}$ denotes the expression measure of gene G_i in mRNA sample C_j.

Let $\mathcal{T}$ be the set of target genes G_i, $i \in \{1, \ldots, |\mathcal{T}|\}$. Typically, this set is constituted by all genes, and thus $|\mathcal{T}| = n$. Also, let $\mathcal{R}$ be the set of regulatory genes RG_i, $i \in \{1, \ldots, |\mathcal{R}|\}$.

The issue of modeling the statistical dependencies between gene expression levels can be described as a supervised learning problem (see Fig. 1) characterized by the following elements: a data generator (the input), a target operator (the output), a training set and a learning machine (Vapnik, 1998).

The goal of a learning machine is to return a hypothesis with low *prediction error*, i.e. a hypothesis which computes an accurate estimate of the output of the target when the same value is an input to the target and the predictor. The prediction error is also usually called *generalization error* since it measures the capacity of the hypothesis to generalize, that is to return a good prediction of the output for input values not contained in the training set.

A typical way of representing the unknown input/output relation is the *regression plus noise form*[2]: $\mathbf{y} = f(x) + \mathbf{w}$, where $f(\cdot)$ is a deterministic function, also known as the *regression function*, and the term $\mathbf{w}$ represents the noise or random error. It is typically assumed that $\mathbf{w}$ is independent of $\mathbf{x}$ and $E[\mathbf{w}] = 0$.

Concerning the expression levels of the genes, the following dependency is assumed for each gene[3] $G_i \in \mathcal{T}$:

$$\mathbf{expr}^{G_i} = f_i(expr^{RG_1}, \ldots, expr^{RG_{|\mathcal{R}|}}) + \mathbf{w} .$$

The goal of the machine learning is to find a model $h(\cdot)$ which is able to give a good approximation of the unknown function $f(\cdot)$ by minimizing an estimation of the *mean integrated squared error* (MISE), which measures the generalization error in the case of a quadratic cost error.

In this paper, we will consider the *leave-one-out* (LOO) algorithm to return an estimate of the MISE prediction error:

$$\widehat{\mathrm{MISE}}_{\mathrm{LOO}} = \frac{1}{N} \sum_{i=1}^{N} (y_i - h(x_i, \alpha_N^{-i}))^2 ,$$

where α_N^{-i} is the set of parameters returned by the parametric identification performed on the training set with the ith sample set aside.

The parametric identification of the hypothesis is done according to the *empirical risk minimization* (ERM) principle (Vapnik, 1998).

In order to assess more easily the quality of a $\widehat{\mathrm{MISE}}_{\mathrm{LOO}}$ estimate, we will focus on the LOO estimate of the *normalized mean integrated squared error* (NMISE):

$$\widehat{\mathrm{NMISE}}_{\mathrm{LOO}} = \frac{\widehat{\mathrm{MISE}}_{\mathrm{LOO}}}{\mathrm{Var}[\mathbf{y}]} .$$

The $\widehat{\mathrm{NMISE}}_{\mathrm{LOO}}$ of a predictor is by definition positive and the closer it is to zero, the better is the generalization accuracy of the predictor. Note that the simplest predictor of the output variable, i.e. the average:

$$\hat{y}_i = \frac{1}{N} \sum_{j=1}^{N} y_j , \qquad i = 1, \ldots, N,$$

has an $\widehat{\mathrm{NMISE}}_{\mathrm{LOO}}$ of one. It follows that a value of $\widehat{\mathrm{NMISE}}_{\mathrm{LOO}}$ close to one for a supervised learning predictor has to be interpreted as a sign of bad accuracy.

---

[2] Throughout this paper, boldface denotes random variables.

[3] If the target gene G_i to be predicted is a regulatory gene, then it will not appear among the inputs.

3 Feature selection

The supervised learning formulation of the dependency between expression levels leads to a prediction problem where the number of inputs is very large with respect to the number of samples. Due to the very high dimensional input space, conventional supervised learning techniques can perform badly (Guyon and Elisseeff, 2003). A preliminary feature selection step is then required.

Many feature selection algorithms include *variable*[4] *ranking*, i.e. ranking variables according to their individual predictive power, as a principal or auxiliary selection mechanism because of its simplicity, scalability, and good empirical success (Guyon and Elisseeff, 2003). Computationally, it is efficient since it requires only the computation of e scores, where e is the number of input variables, and the sorting of the scores. Statistically, it is robust against over-fitting because it introduces bias but it may have considerably less variance (Hastie et al., 2001). Therefore, although variable ranking is not optimal, it may be preferable to variable subset selection methods because of its computational and statistical scalability.

In this paper we adopt the Gram-Schmidt (GS) orthogonalization (Stoppiglia et al., 2003) ranking procedure. Given a set of e candidate features, there are 2^e possible models. Only e models for selection are considered in this paper: the model with the feature ranked first, the model with the first two features, and so on. The price paid for that complexity reduction is the fact that there is no guarantee that the best model is among the e models generated by the procedure.

4 Method

This section presents the algorithmic procedure adopted to assess the predictive power of a number of regulator genes on their targets and the effectiveness of a scale-free aggregation of the most relevant genes.

First, because of the very high dimensional input space, a subset $\mathcal{V}$ of the set of regulatory genes $\mathcal{R}$ is selected. For this purpose:

1. We use the GS orthogonalization algorithm to obtain a ranking of the regulatory genes for each gene G_i of set $\mathcal{T}$. These rankings are called the *individual rankings* $\mathcal{IR}^i$.
2. We build an *aggregate ranking* of the regulatory genes by averaging the positions of each regulatory gene in the individual rankings.
3. We build a random ranking $\mathcal{RR}$, where the regulatory genes are randomly ranked (this is done for comparison purposes – see Sect. 6).
4. We build the subset $\mathcal{V} = \{RG_{v_1}, \ldots, RG_{v_{|\mathcal{V}|}}\}$ of predictors (i) by fixing the number $|\mathcal{V}|$ of genes to be considered in the subset $\mathcal{V}$ of predictors, (ii) by choosing between the individual rankings $\mathcal{IR}^i$, $i \in \{1, \ldots, |\mathcal{T}|\}$, the aggregated ranking $\mathcal{AR}$ and the random ranking RR, and (iii) by taking the $|\mathcal{V}|$ first genes of the chosen ranking.

Second, the function $h_i(\cdot)$, corresponding to a conventional *linear* model (D'haeseleer et al., 1999), is estimated according to the ERM principle (Vapnik, 1998) for each[5] $G_i \in \mathcal{T}$, where

$$\widehat{expr}^{G_i} = h_i(expr^{RG_{v_1}}, \ldots, expr^{RG_{v_{|\mathcal{V}|}}}) \; .$$

Then, the generalization capacity of each model is assessed by estimating the generalization error with the $\widehat{\text{NMISE}}_{\text{LOO}}$. Finally, the average of the $\widehat{\text{NMISE}}_{\text{LOO}}$ obtained is computed.

For each target gene, we also computed the numbers of genes of its individual ranking to be added to $\mathcal{V}$ before the $\widehat{\text{NMISE}}_{\text{LOO}}$ starts to increase. In other words, for each target gene G_{tg}, we start by taking the first gene G of its individual ranking $\mathcal{IR}^{tg}$, i.e. $\mathcal{V} = \{G\}$. Second, if the $\widehat{\text{NMISE}}_{\text{LOO}}$ obtained with the set of genes $\mathcal{V} \cup \{G'\}$, where G' is the next gene of $\mathcal{IR}^{tg}$, is smaller than the one obtained with the genes of $\mathcal{V}$ solely, then G' is added to $\mathcal{V}$ and we continue by considering the following gene of $\mathcal{IR}^{tg}$. If this is not the case, we stop. We thus obtain sets of regulators of varying sizes. Finally, we count the number of genes "targeted" by each regulator, i.e. the number of occurrences of each regulator in these sets.

5 Materials

We now study the predictive power of the yeast *Saccharomyces cerevisiae* genes by applying our procedure on a DNA microarray data set described in Gasch et al. (2000). We used the list of know and *putative* regulatory genes of the yeast *Saccharomyces cerevisiae* used by Segal et al. (2003).

---

[4] The terms *variable* and *feature* are used interchangeably in this paper. Note, however, that a distinction is sometimes made in the literature (Guyon and Elisseeff, 2003).

[5] If the target gene G_i to be predicted is a regulatory gene, then it will not appear among the inputs.

Table 1: Averages of the $\widehat{\text{NMISE}}_{\text{LOO}}$ in the linear case for different number $|\mathcal{V}|$ of regulatory genes for the individual rankings, the aggregated ranking and the random ranking.

$\lvert\mathcal{V}\rvert$	$\mathcal{IR}$ (ind.)	$\mathcal{AR}$ (agg.)	$\mathcal{RR}$ (rand.)
1	0.606	0.781	1.003
2	0.550	0.768	0.854
3	0.527	0.758	0.839
4	0.515	0.741	0.829
5	0.508	0.740	0.787
6	0.504	0.732	0.785
7	0.501	0.725	0.779
8	0.499	0.710	0.763
9	0.497	0.710	0.762
10	0.496	0.698	0.713
15	0.494	0.665	0.678
20	0.497	0.662	0.666
25	0.502	0.652	0.645
30	0.510	0.644	0.649

6 Results

6.1 Predictive Power

The averages of the predictive $\widehat{\text{NMISE}}_{\text{LOO}}$ for different number $|\mathcal{V}|$ of regulatory genes are presented in Table 1. The table contains the results for the individual rankings, the aggregated ranking and the random ranking (average over 10 realizations). It is worthy to remark that all the differences between rankings reported in the table are statistically significant (p-value=0.01) according to a paired t-test on the error vectors and that the results obtained when permuting the outputs have a poor $\widehat{\text{NMISE}}_{\text{LOO}}$ (> 1.4) and are significantly worse than the non permuted results.

The leave-one-out assessment of the predictive power shows that a small subset of genes (about 4 to 8) can have a significantly better performance than the random case and, although inferior, not too far from the predictive power obtained with individual rankings. However, the improvement of the aggregated ranking with respect to a random selection vanishes for larger number of inputs. This suggests that the variability of most of the genes can be explained by a small subset of regulators composed of 4 to 8 genes.

Concerning the individual rankings, the leave-one-out assessment shows that small subsets of genes (about 4 to 8) can each have an important predictive power. Moreover, no significant improvements in terms of $\widehat{\text{NMISE}}_{\text{LOO}}$ occur when bigger subsets are considered. This suggests that, each gene can be predicted by a small number of regulatory genes.

Another outcome of the feature selection procedure is the histogram illustrating the distribution of the number of target genes regulated by a regulatory gene. As shown in Fig. 2(a), it appears that many regulatory genes regulate a small percentage (less than 1%) of genes, while few regulatory genes regulate around 10% of the genome (i.e. about 600 genes). The log-log plot (Fig. 2(b)) of the histogram suggests an underlying power-law distribution, as confirmed by (i) the fitting of a generalized Pareto distribution to the data (Fig. 2(c)) and (ii) the insufficient evidence to reject the null hypothesis that the underlying distribution is a generalized pareto distribution (p-value=0.49) as returned by the hypothesis testing method discussed in Goldstein et al. (2004).

6.2 Biological validation of results

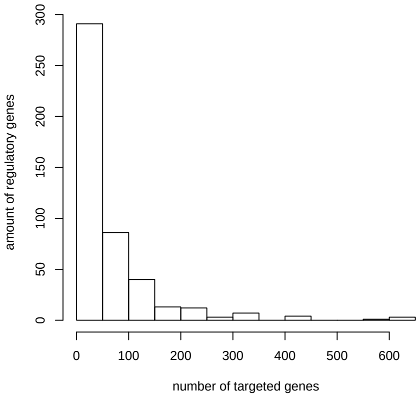
This section aims to show that most of the regulatory genes well ranked in our method's aggregated ranking $\mathcal{AR}$ (i) correspond to regulatory genes identified by Segal et al. (2003) and (ii) form a biologically coherent set of genes. Moreover, (iii) the predictive power of the regulatory genes found by Segal et al. (2003) is comparable to the one of the regulatory genes our procedure identified.

In order to compare with our method, let $\mathcal{AR}^i$ be the set composed of the i first genes of the aggregated ranking $\mathcal{AR}$, let $\mathcal{S}_{all}$ be the set of the 60 regulatory genes identified in Segal et al. (2003), and let $\mathcal{S}_{main}$ be the subset of 22 regulators considered as the "main" regulators of $\mathcal{S}_{all}$.

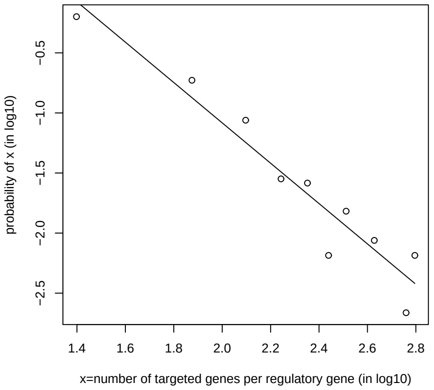
Table 2 shows that 32 to 50% and 60 to 90% of the regulatory genes identified by sets $\mathcal{AR}^i$, $i \in \{10, 15, 20, 25, 30\}$, are also in $\mathcal{S}_{main}$ and $\mathcal{S}_{all}$, respectively. Moreover, these results are highly significant. Indeed, the corresponding p-values of the hypergeometric distribution are all smaller than 1.0×10^{-4}.

The 10 first genes of the aggregated ranking $\mathcal{AR}$ obtained with our procedure are listed in Table 3 (genes in bold belong to set $\mathcal{S}_{main}$). We noted that 5 of these genes, identified by an asterisk (*), form a biologically coherent set of genes as they are involved in starvation, in nutrient limitation or in nutrient control.

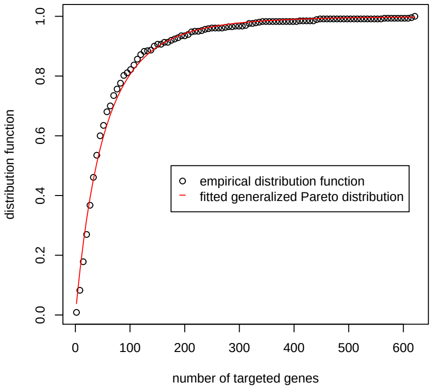
The predictive power of the regulators obtained by Segal et al. (2003) is similar, in terms of $\widehat{\text{NMISE}}_{\text{LOO}}$, to the one obtained with our method: averages of



(a) Histogram of the number of target genes regulated by a regulatory gene.



(b) Log-log plot of the histogram.



(c) Fitting of a generalized Pareto distribution to the data.

Figure 2: Number of target genes regulated by a regulatory gene.

Table 2: Number of regulatory genes of sets $\mathcal{AR}^i$, $i \in \{10, 15, 20, 25, 30\}$, that belong to $\mathcal{S}_{main}$ and $\mathcal{S}_{all}$, respectively.

	$\mathcal{AR}^{10}$	$\mathcal{AR}^{15}$	$\mathcal{AR}^{20}$	$\mathcal{AR}^{25}$	$\mathcal{AR}^{30}$
$\mathcal{S}_{all}$	9	11	12	15	18
$\mathcal{S}_{main}$	5	7	8	8	10

Table 3: The 10 first genes of the aggregated ranking obtained with our procedure. Genes in bold belong to $\mathcal{S}_{main}$. Genes identified by an asterisk (*) are involved in starvation, in nutrient limitation or in nutrient control.

Rank	Systematic name	Standard name
1	**YPL230W***	ORF Uncharacterized
2	**YJL164C**	TPK1
3	**YGL099W**	LSG1
4	**YDR096W***	GIS1
5	YER118C	SHO1
6	YLL019C	KNS1
7	YGL208W*	SIP2
8	YPL203W*	TPK2
9	**YIL101C***	XBP1
10	YJL103C	ORF Uncharacterized

the $\widehat{\text{NMISE}}_{\text{LOO}}$ by using both lists of main predictors $\mathcal{S}_{main}$ and of all predictors $\mathcal{S}_{all}$ are 0.688 and 0.620, respectively. The ranking produced by our method thus seems to yield better results compared to list $\mathcal{S}_{all}$ and less good results compared to $\mathcal{S}_{main}$ in terms of $\widehat{\text{NMISE}}_{\text{LOO}}$.

7 Conclusion

In this paper, we experimentally assessed the predictive power of the scale-free paradigm in a supervised learning framework. The results obtained, although preliminary, tend to validate this paradigm. Indeed, it appears that (i) each gene can be predicted by a small subset of regulatory genes, and (ii) on a global scale, a small subset of regulatory genes, called the hubs, can have a non-negligible predictive power on all the target genes.

Moreover, most of the regulatory genes well ranked in our procedure correspond to regulatory genes found by Segal et al. (2003) and form a biologically coherent set of genes.

Future work will focus on testing other feature selection algorithms and using other learning algorithms. Another interesting direction consists in generating in silico DNA microarray data for given networks by using existing simulation techniques. The effect of network topology on the predictive power that the expression levels of a set of regulator genes have on the expression levels of some target genes could then be more accurately assessed by using different topologies (scale-free, random and small-world network topologies for example).

Acknowledgements

This research was supported by the "Communauté Française de Belgique" through an ARC project (no. 04/09–307) research grant. The project is entitled "Integrating experimental and theoretical approaches to decipher the molecular networks of nitrogen utilisation in yeast".

References

A.-L. Barabási and Z. N. Oltvai. Network biology: Understanding the cell's functional organization. *Nature Reviews*, 5:101–114, 2004.

A.-L. Barabási, Z. N. Oltvai, and S. Wuchty. Characteristics of biological networks. *Lecture Notes in Physics*, 650:443–457, 2004.

P. D'haeseleer, X. Wen, S. Fuhrman, and R. Somogyi. Linear modeling of mRNA expression levels during CNS development and injury. In *Proceedings of the Pacific Symposium on Biocomputing*, 4: 41–52, 1999.

I. Farkas, H. Jeong, T. Vicsek, A.-L. Barabási, and Z. N. Oltvai. The topology of the transcription regulatory network in the yeast, *Saccharomyces cerevisiae*. *Physica A*, 318:601–612, 2003.

A. P. Gasch, P. T. Spellman, C. M. Kao, O. Carmel-Harel, M. B. Eisen, G. Storz, D. Botstein, and P. O. Brown. Genomic expression programs in the response of yeast cells to environmental changes. *Molecular Biology of the Cell*, 11:4241–4257, 2000.

M. L. Goldstein, S. A. Morris, and G. G. Yen. Problems with fitting to the power-law distribution. *The European Physical Journal B*, 41:255–258, 2004.

I. Guyon and A. Elisseeff. An introduction to variable and feature selection. *Journal of Machine Learning Research*, 3:1157–1182, 2003.

T. Hastie, R. Tibshirani, and J. Friedman. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Springer Series in Statistics. Springer, 2001.

H. Jeong, B. Tombor, R. Albert, Z. N. Oltvai, and A.-L. Barabási. The large-scale organization of metabolic networks. *Nature*, 407(6804):651–654, 2000.

E. Segal, M. Shapira, A. Regev, D. Pe'er, D. Botstein, D. Koller, and N. Friedman. Module networks: Identifying regulatory modules and their condition-specific regulators from gene expression data. *Nature Genetics*, 34(2):166–176, 2003.

H. Stoppiglia, G. Dreyfus, R. Dubois, and Y. Oussar. Ranking a random feature for variable and feature selection. *Journal of Machine Learning Research*, 3:1399–1414, 2003.

V. van Noort, B. Snel, and M. A. Huynen. The yeast coexpression network has a small-world, scale-free architecture and can be explained by a simple model. *EMBO reports*, 5(3):280–284, 2004.

V. N. Vapnik. *Statistical Learning Theory*. Wiley Series on Adaptive and Learning Systems for Signal Processing, Communications, and Control. Wiley-Interscience, 1998.

X. Zhou, X. Wang, and E. R. Dougherty. Gene prediction using multinomial probit regression with Bayesian gene selection. *EURASIP Journal of Applied Signal Processing*, 1:115–124, 2004.

Generating networks with realistic properties: The topology of locally evolving random graphs

Andreas Krause*
*School of Management
University of Bath
Bath BA2 7AY
Great Britain `mnsak@bath.ac.uk`

Abstract

We present a model in which a random graph evolves locally by randomly changing edges in the immediate neighborhood of a node. We find that the emerging graph maintains its small diameter and obtains a high clustering coefficient as well as a power-law tail in the degree distribution. The scaling of the diameter and clustering coefficient of the model are also obtained.

Keywords: Evolving graphs, random graphs, power law, local search
PACS numbers: 89.75.Hc, 89.65.-s
Mathematics Subject Classification: 05C80, 91D30
JEL Classification: D85

1 Introduction

Many investigations into the properties of real networks have shown that they can mostly be characterized by three key properties: a small diameter, a high clustering coefficient and a power-law tail of the degree distribution, Albert and Barabási (2002) provide a comprehensive review. Since Watts and Strogatz (1998) introduced small world networks there has been an abundance of models with high clustering coefficients and small diameters as well as models of preferential attachment generating a power-law tail of the degree distribution following the introduction of scale-free networks by Barabási and Albert (1999).

Until the introduction of local edge formation, however, it had been proved difficult to generate graphs exhibiting all desired properties simultaneously. The vast majority of models proposed using local edge formation thus far are models of growing networks, either through adding new links and/or new nodes, (Davidsen et al., 2002; Jost and Joy, 2002; Vazquez, 2003; Csányi and Szendrői, 2003; Li and Chen, 2003; Blanchard and Krüger, 2004; Geng and Li, 2005). While growing networks might be appropriate for some applications, in many cases it would be more adequate to assume that the number of nodes and edges remains constant and a mechanism is required that relies only on re-connecting these edges and thereby generating realistic properties.

Blanchard et al. (2005) developed a model in which after an initial phase of a growing network, existing edges get re-connected without new nodes or edges being added. Their mechanism requires in each time period a single node to re-align one of his edges to another node which has a distance of two, what they call their "my friends are your friends" principle. After sufficient time periods this model shows a high clustering coefficient, small diameter and a fat-tailed degree distribution, but no evidence for a power-law tail as found in many real networks.

In this paper we employ a very similar rule for the evolution of a graph without adding or subtracting any nodes or edges. Our focus is on small networks having only a very limited number of nodes and we assume that the initial network is random rather than regular. As in Blanchard et al. (2005) we find evidence for a small diameter and a high clustering coefficient whose properties we compare with that of a random graph and in contrast to Blanchard et al. (2005) we find strong evidence for a power-law tail of the degree distribution.

2 The graph evolution

The starting point is a random undirected graph with $N > 2$ nodes in which a link exists between any two nodes with probability of $p \in (0; 1]$. We now let this graph evolve in discrete time steps using the following algorithm in each time step:

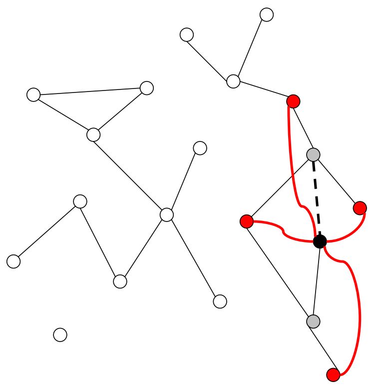


Figure 1: Local evolution of the graph. Suppose the black node is chosen for updating and intends to replace this thick dashed link. He will be able to form a link to any of his neighbor's neighbor, market red. Which link is actually chosen is randomly determined.

1. Select a node x randomly with equal probability for all nodes,
2. Select an edge i of this node randomly with equal probability for all edges,
3. Select randomly with equal probability for all nodes another node y which has a distance of 2 to the already selected node x,
4. If node x has no common edge with node y, remove edge i and replace it with an edge i' connecting nodes x and y,
5. If node x has no edges, there does not exist a node y which has a distance of 2 to node x or the selected node y already has a common edge with node x, no changes to the edges are made.

The number of time periods investigated in the analysis is denoted by T and we generally investigate the resulting properties after $T = 20N$ time periods. This algorithm is illustrated in figure 1.

This algorithm is in essence the same as used by Blanchard et al. (2005) with one important difference: their model starts with a simple circle to which they add links using a similar algorithm as above until they have obtained the desired number of links in the model and then from this point onwards let the graph evolve as described before.

3 The resulting network topology

Using the algorithm described in the previous section we conducted a number of simulations using a variety of parameter constellations, exploring any combination with $N \in \{50, 100, 150, 200, 250\}$ and $pN \in \{2, 4, 6, 8, 10\}$. For each parameter constellation we ran 100 simulations and use the average values for the diameter and clustering coefficient while we aggregate all nodes to obtain the degree distribution. Any analysis is conducted after $T = 20N$ time periods and we do not observe any significant changes when the number of time periods is extended further.

3.1 Network diameter

Most real networks have a diameter which is only slightly larger than that of a random graph. As we can establish from figure 2 this is also true in our model for $pN \geq 6$, i.e. if the average number of links of a node is at least 3. For random graphs the average path length ℓ scales as

$$\ell_{rand} \sim \frac{\ln N}{\ln pN}, \tag{1}$$

which we also find for our model, whose average path length is only about 10% higher than that of a random graph. For smaller values of pN, however, the scaling is approximately linear in N rather than logarithmic. A similar result has also been obtained by Blanchard et al. (2005) who found that an average of at least two links per node or more was required to obtain a small diameter.

3.2 Clustering coefficient

Another characteristic of real networks is a clustering coefficient which is significantly higher than that of a random graph with the same number of nodes and edges. For random graphs we know that

$$C_{rand} = p. \tag{2}$$

As can be seen from figure 3 the clustering coefficient in our model is significantly higher, with again the case of $pN = 2$ standing out slightly. For $pN \geq 4$ we find a different scaling of the clustering coefficient C as follows:

$$C = 20.6646pN^{-0.2880}. \tag{3}$$

We thus observe a high clustering coefficient in our model which is slowly decreasing in the number of

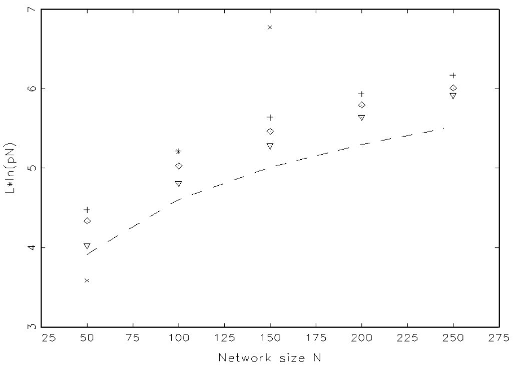


Figure 2: Average path length ℓ of the resulting graph for different sizes of the network N and probabilities of two nodes being connected p: $pN = 4$ ($\times$), 6 ($\bigtriangledown$), 8 ($\diamond$), 10 ($+$). Results are based on averaging over 100 runs for each of the 25 parameter constellations after $20N$ time steps. The dashed line represents the results from a random graph.

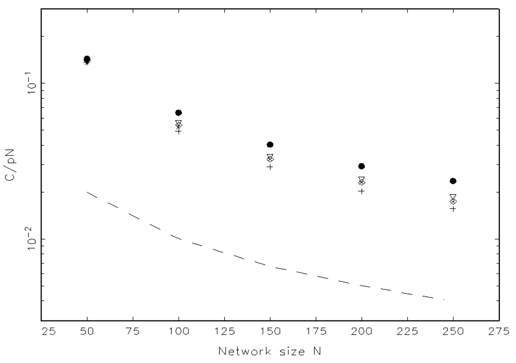


Figure 3: Clustering coefficient C of the resulting graph for different sizes of the network N and probabilities of two nodes being connected p: $pN = 2$ ($\bullet$), 4 ($\times$), 6 ($\bigtriangledown$), 8 ($\diamond$), 10 ($+$). Results are based on averaging over 100 runs for each of the 25 parameter constellations after $20N$ time steps. The dashed line represents the results from a random graph.

nodes, a property for which there is weak empirical evidence in the overview collated in Albert and Barabási (2002), where they mention the clustering to be nearly constant, although the graph they present suggests a small negative relationship.

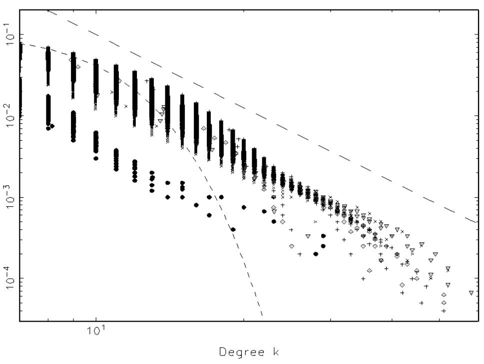


Figure 4: Degree distribution $P(k)$ of the tail from the resulting graph after $20N$ time steps for different sizes of the network N and probabilities of two nodes being connected p: $pN = 2$ ($\bullet$), 4 ($\times$), 6 ($\bigtriangledown$), 8 ($\diamond$), 10 ($+$). The short dashed line represents the results from a random graph with $pN = 10$ and the long dashed lines that of a power law distribution with an exponent of 3 (distribution shifted upwards for clarity). An exponential cut-off can be seen at approximately $k = 20$. The distribution is obtained from 100 simulations of each of the 25 parameter constellations using $pN = 2, 4, 6, 8, 10$ and $N = 50, 100, 150, 200, 250$.

3.3 Degree distribution

The degree distribution as illustrated in figure 4 shows clear evidence of a power-law tail with an exponent of approximately 3. This result is in clear contrast to the very similar model of Blanchard et al. (2005) who find evidence for fat tails but no sign of a power-law for the tail. Given that their algorithm is, apart from the initial phase, nearly identical to ours, this result is very surprising and merits further consideration of the relevance of the initial graph for these results which seems to have a significant influence on the results.

We observe from figure 4 that, again apart from the case of $pN = 2$, the degree distributions scale quite uniformly with a power-law tail. However, closer inspection of the distribution as illustrated in figure 5 shows that we do not observe a perfect power-law tail, but it rather appears to be the combination of two exponential tails. We generally observe an exponential cut-off at approximately $k = 20$. Given that we did not investigate larger graphs it has to be seen whether this observation can be explained with the finite size of the graph or is a more genuine feature of the algorithm used. Evidence from the graphs investigated

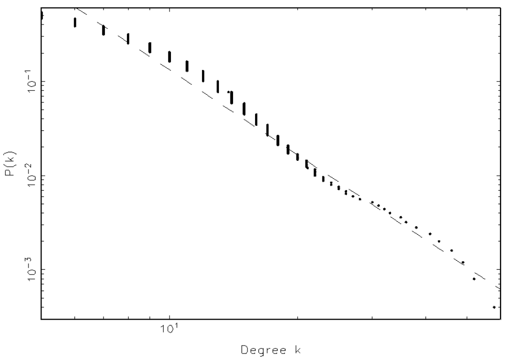


Figure 5: Degree distribution $P(k)$ from the resulting graph after 5,000 time steps for $N = 250$ nodes and $pN = 6$. The long dashed lines that of a power law distribution with an exponent of 3. An exponential cut-off can be seen at approximately $k = 20$. The distribution is obtained from 100 simulations with this parameter constellation.

here suggests that the finite size effect drives this result because the distribution moves closer to a power-law tail as we increase the number of nodes.

3.4 Graph structure

Apart from the properties discussed we see that one other important element of the resulting graph is that nodes with a small degree tend to reduce their degrees even further over time given that they are hardly attracting any new links but loose existing links at the same rate as any other node. This will inevitably result in such nodes becoming isolated or forming small subgraphs which are not connected with the large component or each other. Figure 6 shows an example of the initial random graph and the resulting graph after 2,000 time steps, clearly illustrating this property. We furthermore observe that the large component is usually very well connected and does not show clear evidence of any further distinguishable features, being quite homogeneous in its structure.

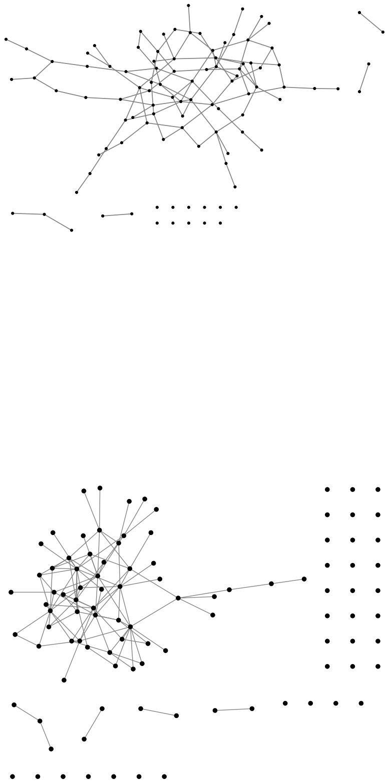


Figure 6: Single realization of an initial random graph with $N = 100$ and $Np = 2$ (top panel) and the evolved graph after $T = 2000$ time steps (bottom panel). It has to be noted that with a larger number of edges the results are qualitatively the same, but the large number of edges makes any representation very difficult to visualize.

3.5 Explanation of findings

The algorithm used in this model generates a high clustering coefficient and a power-law tail of the degree distribution while maintaining the small diameter of the initial random graph. This result can easily be explained from the way new edges are formed. Any node with a large number of neighbors is also likely to have a large number of nodes with a distance of two, i.e. neighbor's neighbors, and thus is quite likely to be chosen as the destination of an edge by a randomly selected node. In contrast, a node with only few neighbors will also only have a small number of neighbor's neighbors and the probability of him being chosen as the destination of a link is relatively small, causing the number of edges connected to this node to fall and in some instances leading to a node becoming isolated. Thus nodes with a high degree tend to attract more links than those with a low degree, giving rise to a similar effect as the preferential attachment of nodes in the Barabási and Albert (1999) model of scale-free graphs which exhibits a power-law tail of the degree distribution, also with an exponent of 3.

The high clustering coefficient emerges as the consequence of the local evolution of the graph where edges are established to a neighbor's neighbor, thus increasing the number of triangles in the graph and thereby increasing the clustering coefficient. The clustering coefficient will only be limited by the number of links available in the graph. The random nature of any connections is maintained, however, thus retaining the small diameter of the graph.

4 Discussion and conclusions

The model presented in this paper allows a graph to evolve randomly in its immediate neighborhood. The resulting graphs had properties that were largely consistent with those of real networks, namely the small diameter, high clustering coefficient and power-law tail of the degree distribution. The algorithm for the evolution of the graph needs to be adjusted, however, to avoid the appearance of a large number of isolated nodes or small components which are not realistic.

Nevertheless, the algorithm can be described as realistic for many social networks where new contacts are often made through existing contacts, the "friend's friend" or "neighbor's neighbor" principle, but it remains unclear at this stage how much the initial network structure affects the graph topology, especially in light of the results obtained by Blanchard et al. (2005) for the degree distribution. Future research needs to clarify the importance of the initial conditions for the results obtained here, in particular evaluating whether similar results can be obtained when starting with a regular graph.

It is furthermore of interest to evaluate the sensitivity of the way new links are determined by considering a wider range of rules, e.g. preferential attachment, as well as to include some random attachment outside the neighborhood to prevent nodes from becoming isolated as in our model.

References

Réka Albert and Albert-László Barabási. Statistical mechanics of complex networks. *Review of Modern Physics*, 74(1):47–97, 2002.

Albert-László Barabási and Réka Albert. Emergence of scaling in random networks. *Science*, 286:509–512, 1999.

Philippe Blanchard, Tyll Krüger, and Andreas Ruschhaupt. Small world graphs by iterated local edge formation. *Physical Review E*, 71(4):046139, 2005.

Phillippe Blanchard and Tyll Krüger. The "cameo principle" and the origin of scale-free graphs in social networks. *Journal of Statistical Physics*, 114 (516):1399–1416, 2004.

Gábor Csányi and Balász Szendrői. Structure of a large social network. cond-math/0305580, 2003.

Jörn Davidsen, Holger Ebel, and Stefan Bornholdt. Emergence of a small world from local interactions: Modelling acquaintance networks. *Physical Review Letters*, 88(12):128701, 2002.

Xianmin Geng and Qiang Li. Random models of scale-free networks. *Physica A*, 356:554–562, 2005.

Jürgen Jost and Maliackal P. Joy. Evolving networks with distance preferences. *Physical Review E*, 66 (3):036126, 2002.

Xiang Li and Guanrong Chen. A local-world evolving network model. *Physica A*, 328:274–286, 2003.

Alexei Vazquez. Growing network with local rules: Prefrential attachement, clustering hierarchy, and degree correlations. *Physical Review E*, 67(5): 056104, 2003.

Duncan J. Watts and Steven H. Strogatz. Collective dynamics of 'small-world' networks. *Nature*, 393: 440–442, 1998.

Evolving Genetic Regulatory Networks Performing as Stochastic Switches

André Leier* Kevin Burrage*
*Advanced Computational Modelling Centre
University of Queensland, Brisbane, QLD 4072 Australia
{leier,kb}@acmc.uq.edu.au

Abstract

Recent studies have shown that small genetic regulatory networks (GRNs) can be evolved *in silico* displaying certain dynamics in the underlying mathematical model. It is expected that evolutionary approaches can help to gain a better understanding of biological design principles and assist in the engineering of genetic networks. To take the stochastic nature of GRNs into account, our evolutionary approach models GRNs as biochemical reaction networks based on simple enzyme kinetics and simulates them by using Gillespie's stochastic simulation algorithm (SSA). We have already demonstrated the relevance of considering intrinsic stochasticity by evolving GRNs that show oscillatory dynamics in the SSA but not in the ODE regime. Here, we present and discuss first results in the evolution of GRNs performing as stochastic switches.

1 Introduction

Genetic regulatory networks (GRNs) are complex systems composed of molecular species including genes, RNAs, transcriptional factors and other proteins, that chemically interact by specific reactions, thereby controlling the expression levels of the genes. GRNs are the fundamental units governing cell activities. Understanding them is essential to gaining understanding of a cell's development, organization, function and, ultimately, control. Revealing the design principles of GRNs is an important step towards this goal.

Studies suggest that GRNs have a modular structure, that is, they are composed of small constituent subnetworks or "modules" representing basic building blocks (Hartwell et al., 1999). Breaking down a GRN into its modules and analysing their individual structure and dynamics as well as their interactions (reverse engineering) can facilitate the identification of the GRN's overall functionality. The inverse approach, synthesis of GRNs instead of decomposition, is an alternative way to acquire new insights. By trying to build GRNs showing specific behaviours we may recognize certain design principles. Engineering (synthetic) genetic circuits either by hand (Elowitz and Leibler, 2000; Gardner et al., 2000; Becskei et al., 2001; Kobayashi et al., 2004) or by using directed evolution in vivo (Yokobayashi et al., 2002) is both time-consuming and expensive. Alternatively, evolutionary approaches *in silico* have been applied to find genetic networks performing as bistable switches or oscillators in the underlying mathematical model (François and Hakim, 2004; Leier et al., 2006). Naturally, in terms of the *in vivo* performance, the resulting networks are only as accurate as the formalisms describing them. Thus, since the mathematical models are only simplifications of the biological processes, the evolved GRNs may not perform *in vivo* as they do *in silico*. Nevertheless, as first results show, it can still be insightful to study the structure and characteristics of the resulting networks, taking the mathematical model into account (François and Hakim, 2004; Leier et al., 2006).

The molecular character of GRNs makes them intrinsically stochastic and noisy (McAdams and Arkin, 1997; Arkin et al., 1998; Elowitz et al., 2002; Hasty and Collins, 2002). The uncertainty of knowing when a reaction occurs and which reaction it might be causes fluctuations that become increasingly noticeable with smaller numbers of interacting molecules. Although noise can adversely affect cell function, it is also considered a source of robustness and stability, signal amplification, and selection of signalling pathways. Intrinsic stochasticity can be modelled by using the stochastic simulation algorithm (SSA), introduced by Gillespie (1977, 2001). The SSA is a statistically exact simulation method assuming that the system is homogeneous and populations are well-mixed within a constant volume.

In Leier et al. (2006) we present a genetic programming (GP) approach for evolving biochemical reaction networks based on simple enzyme kinetics that demonstrate desired dynamics when simulated with the SSA. It is inspired by the work of François and Hakim (2004) that use a similar approach to evolve genetic networks that are modelled by ordinary differential equations (ODEs). However, the dynamic behaviour of biochemical systems in the ODE regime (deterministic, continuous) and the SSA regime (discrete, stochastic) can be very different (Elowitz and Leibler, 2000; Heuett and Qian, 2005). Therefore, it is important to see how certain dynamic behaviour can evolve in the presence of noise. Since one neglects the stochastic nature of molecular interactions when modelling GRNs with ODEs, stochastic simulation can give deeper insights into chemical dynamics when there are only small numbers of molecules in the system.

We used our GP system to evolve GRNs with noisy oscillatory dynamics (Leier et al., 2006). Evolutionary runs breed GRNs that clearly oscillate in the discrete, stochastic regime but not when modelled as ODEs. The outcomes also confirm results from François and Hakim (2004) stating the importance of post-translational modifications for the functioning of the networks.

Switching, in particular bistable switching, is another fundamental dynamics that can be observed in many biological systems. Genetic switches are known to be responsible for controlling developmental processes and responding to environmental and intercellular signals. Biological switches and their underlying mechanisms can be quite different (Wolf and Arkin, 2003): a switch can be mono-stable, also called memory-less (the system moves back into its single stable state once the switching stimulus ends) or multistable (the system switches between two or more stable states by a transient application of a stimulus). Switching can occur randomly or by induction. Mechanisms, identified to allow bistable switching include cross-repressive feedback loops with cooperativity ((Gardner et al., 2000) utilized this mechanism to synthesize a toggle switch in *E. coli*) and positive feedback with cooperativity (based on this mechanism (Becskei et al., 2001) construct a bistable switch in *S. cerevisiae*).

This work presents first results in the evolution of genetic switches under intrinsic noise using the GP approach. For a successful evolution identification of switching behaviour in GRNs is a crucial factor. Although switching is often associated with a stimulus or induction signal, we first evolved GRNs that show periodic, not externally stimulated switching behaviour. However, we also analysed the resulting networks in terms of induced switching by injection of certain key molecules. Since noise can bounce trajectories between quasi-equilibrium states SSA dynamics of switches can be very different from ODE dynamics when there are only small numbers of molecules involved. Therefore, we also compared the SSA trajectories with the solutions of the corresponding ODE models and tested the ODE models for induced switching. Here, we focus on GRNs that show switching behaviour in the SSA but not in the ODE regime.

2 Methods

In the following material we briefly describe the reaction model, the stochastic simulation algorithm and the genetic programming approach. For additional information on the reaction model and the GP system we refer to Leier et al. (2006).

2.1 Reaction Model

Our reaction model describes GRNs as a set of species (genes, mRNAs, proteins and complexes such as gene-protein bindings or protein complexes) and *master reactions* governing their interactions. Master reactions are small sets of elementary and irreversible chemical reactions that correspond to biologically meaningful processes. Each elementary reaction either is a first order reaction, a second order reaction or a homodimer formation determined by the reaction rate constant. The seven master reactions are: gene transcription and translation (including basal transcription), transcriptional regulation of genes provided with two regulatory binding sites operating in a mode of cooperativity (based on the model described in Goutsias (2005)), protein modification, dimerization and three types of degradation (partial, catalytic and partial catalytic). Master reactions and their associated elementary reactions are listed in Table 2.1.

2.2 Stochastic Simulation Algorithm

Two issues led to Gillespie Gillespie (1977) introducing the SSA that exactly simulates the evolution of a discrete, stochastic chemical kinetic process in a well stirred mixture: (1) certain key molecules may be produced in quite low numbers (models based on continuous concentrations miss out on the discrete nature) and (2) the system is intrinsically noisy due to the uncertainty of knowing when a reaction and what

Master Reactions
Basal transcription and translation:
$a \rightarrow a + a_{\text{mRNA}}$
$a_{\text{mRNA}} \rightarrow a_{\text{mRNA}} + A$
$a_{\text{mRNA}} \rightarrow \emptyset$
$A \rightarrow \emptyset$
Transcriptional regulation:
$a + T \rightarrow aT$
$aT \rightarrow a + T$
$aT \rightarrow aT + a_{\text{mRNA}}$
$aT + R \rightarrow aTR$
$aTR \rightarrow aT + R$
Dimerization:
$A + B \rightarrow$ A:B
$A{:}B \rightarrow A + B$
Partial degradation:
$A{:}B \rightarrow A$
Catalytic degradation:
$A + B \rightarrow A$
Partial catalytic degradation:
$A{:}B + C \rightarrow A$

Table 1: The set of master reactions that are the building blocks of the genetic networks. Each gene has two regulatory binding sites (R_1 and R_2) that work in a cooperative manner: binding of a transcription factor at R_2 requires R_1 to be occupied by another factor. Lowercase letters (a,b, etc.) represent genes with unbound regulatory sites. When a transcription factor T is bound at R_1 of a the binding is denoted aT. The regulatory effect (positive or negative regulation) depends on the corresponding reaction rate constant. Binding at R_2 excludes any transcriptional activity and hence, represses transcription. The case of a repressor R bound to aT is denoted as aTR. mRNA is indicated such as in a_{mRNA}. Capitalized letters (A,B, etc.) represent the proteins translated from the associated mRNA. Protein complexes are represented using colons (i.e. a protein complex composed of proteins A and B is represented by $A{:}B$). Each reaction is specified by a reaction rate constant.

reaction takes place (deterministic models ignore stochasticity). Hence, in this context, the use of continuous differential equations methods is debatable.

In the following material we briefly describe the SSA. It is assumed that the biochemical system is well-mixed within a constant volume held at constant temperature. Let there be N molecular species $\{S_1, \ldots, S_N\}$ that chemically interact through M reactions $\{R_1, \ldots, R_M\}$. The system state at time t is described by a vector $X(t) \equiv (X_1(t), \ldots, X_N(t))^T$ where $X_i(t)$ is the number of molecules of species i at time t. Each reaction R_j can be uniquely defined by its *propensity function* a_j, where $a_j(X(t))dt$ is the probability that reaction R_j will occur somewhere in the system within the time interval $(t, t + dt)$, and its assigned *stoichiometric vector* ν_j that specifies the update of the system state when reaction R_j occurred. This is defined by ν_{ji} for $i = 1, \ldots, M$, which is the change in the number of S_i molecules produced by one R_j reaction. Our SSA implementation simulates the time evolution of a system according to the *direct method* Gillespie (2001): two independent samples r_1 and r_2 of the uniform random variable $\mathbf{U}(0, 1)$ are drawn consecutively. The length of the time interval $[t, t + \tau)$ is given by

$$\tau = \frac{1}{a_0(X(t))} \ln(\frac{1}{r_1}) \,,$$

where

$$a_0(X(t)) = \sum_{j=1}^{M} a_j(X(t))$$

is the sum of all propensities. The specific reaction R_j occurring in $[t, t + \tau)$ is determined by the index j satisfying

$$\sum_{j'=1}^{j-1} a_{j'}(X(t)) < r_2 a_0(X(t)) \leq \sum_{j'=1}^{j} a_{j'}(X(t)) \,.$$

The state vector is then updated as

$$X(t + \tau) = X(t) + \nu_j \,.$$

For the elementary reactions (first and second order reactions and homodimer formations) used in our reaction model, the corresponding propensity functions are shown in Table 2.2.

Since the SSA can become very computationally intensive when time steps become very small (due to large numbers of reactions, large reaction rates or large numbers of molecules), we limit our model to small numbers of species.

Reaction	**Propensity Function**
First order reaction $S_k \overset{c_j}{\rightarrow} S_l$	$a_j = c_j * X_k$
Second order reaction $S_k + S_l \overset{c_j}{\rightarrow} S_m$ with $S_k \neq S_l$	$a_j = c_j * X_k * X_l$
Homodimer formation $S_k + S_k \overset{c_j}{\rightarrow} S_l$	$a_j = c_j * X_k * \frac{X_k - 1}{2}$

Table 2: Propensity functions for three elementary types of reactions. c_j is the reaction rate constant of the respective reaction.

2.3 Genetic Programming System

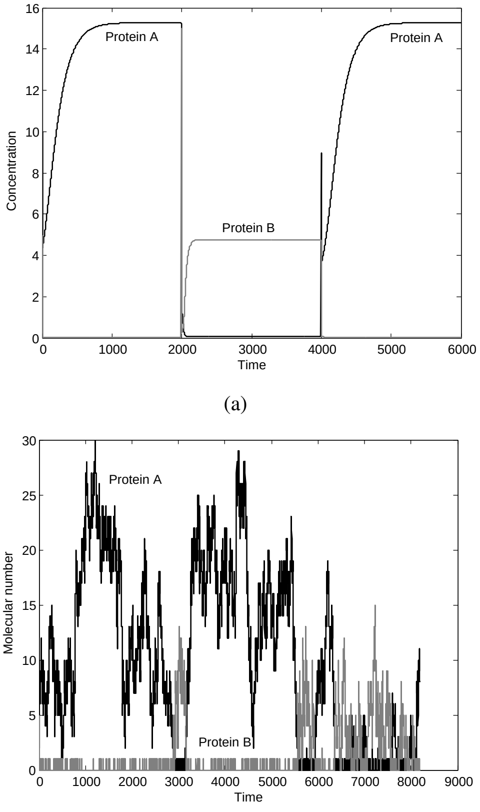
Individuals in the GP population are GRNs according to the reaction model. Each individual is assigned a fitness value describing how well its dynamics meets the prescribed requirements. This value is calculated by simulating the reaction system using the SSA over a predefined time and analysing the resulting trajectory. For our purposes, we define constraints the trajectory must satisfy in order not to be penalized. A penalty is equal to the amount by which the solution exceeds the constraint. Penalties are weighted according to how the constraints are met. Then, penalties are summed up to obtain the fitness value. That is, the lower the fitness the better the individual. To guide the evolution towards GRNs with a switching behaviour in the concentration of a particular protein (e. g. protein A), we define the following constraints (numerical values in parentheses exemplify the thresholds): (i) the molecular number n of the species has to be at "low level" ($n < 20$) for a minimum time period T_1 ($T_1 = 500$), (ii) the molecular number n of the species has to be at "high level" ($150 < n < 200$) for a minimum time period T_2 ($T_2 = 500$), (iii) the time period for a switch between low and high level (and vice versa) is limited by T_3 ($T_3 = 50$). There has to be at least one switch between "low" and "high" concentration levels in the trajectory of the corresponding protein, otherwise the individual's fitness is set to a maximum value.

Evolution is driven by repeated selection and mutation. The selection method is a simple (50+50) evolutionary strategy, that is, 50 individuals produce one offspring each and the best 50 out of 100 individuals build the new population. Offspring are produced by mutation of the parent individuals. The mutation operators involve random modifications of the reaction rate constants and additions and deletions of master reactions. That is, not only rate constants are evolved but also the structure of the GRN. To focus on small regulatory networks we fixed the number of genes at two. This reduced the search space and facilitated evolution. When reactions are added to the GRN the reaction rates are uniformly drawn from $[0, 1]$. A reaction rate is mutated by multiplication with a random number from $[0, 2]$. At the beginning of an evolutionary run, the initial concentrations are randomly chosen from $1, 2, \ldots, 10$ and remain fixed for the entire evolution. Evolution is terminated if the number of generations without fitness improvement exceeds a threshold (100 generations). The evolved networks are simulated several times to verify their dynamic behaviour.

3 Results

We already mentioned in the introduction that ODE and SSA models of the same GRN can display very different dynamics. Figure 3 illustrates this for a bistable switch evolved in the ODE regime (François and Hakim, 2004) (Figure 3A). While the ODE model allows induced switching between two equilibrium states the SSA trajectories are very noisy with many irregular switches. However, protein A and B seem to be in opposite levels, that is, whenever A exists in higher numbers, B does not and vice versa. We note that François and Hakim (2004) also present the GRN with different reaction rates that works as a bistable switch in both regimes where the species in the high concentration level have several hundred molecules. For the rest of this section, we present and discuss evolved GRNs.

The resulting GRNs can be quite different in their dynamics and their structure, i. e. in the composition of the master reactions. Interestingly, although our fitness function is not geared to the evolution of bistable switches, a few GRNs show some sort of bistable switching behaviour, usually with different high level concentrations of protein A (as predefined) and B. Figure 2(a) shows such an evolved GRN where the periodic switching between the two states, either protein A in high and protein B in low numbers or vice versa, is driven by noise (Figure 2(b)). In this GRN, protein A is the transcription factor of gene b. It may bind to the gene's binding sites R'_1 and R'_2. Binding of A at R'_1 activates enhanced transcription of b, additional binding of A at R'_2 represses transcription. In this example the regulatory region of gene a is unused. As expected, the dynamic behaviour of the GRN in the ODE regime is quite different (cf. Figure 2(c)) from the SSA regime. The corresponding ODE model shows a single switch when



(a)

(b)

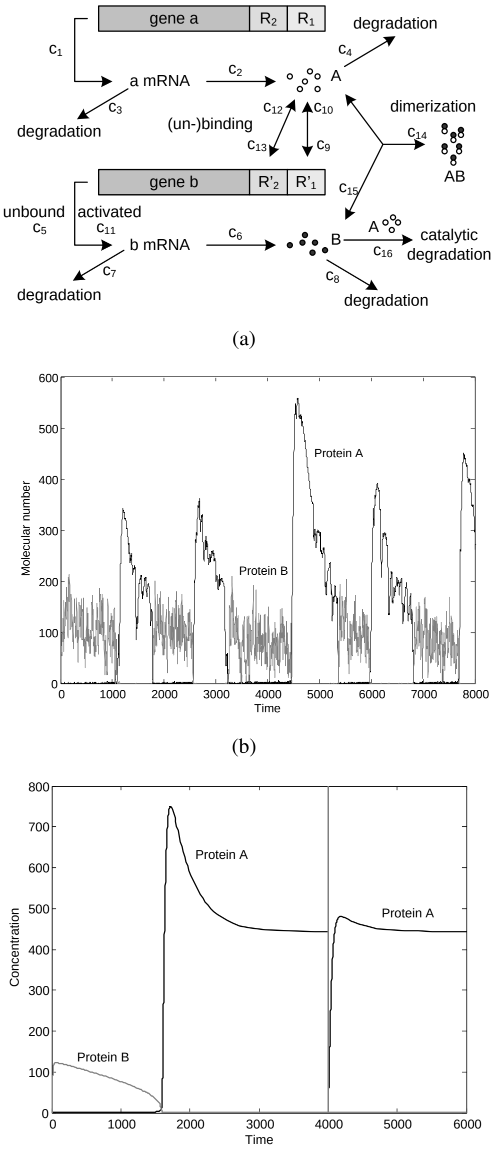
Figure 1: Dynamics of an evolved switch in the ODE regime by François and Hakim (2004). (a) ODE dynamics: the switching is induced by two pulses. At $t = 2000$ and $t = 4000$ we add protein concentrations $[A] = 15$ and $[B] = 9$, respectively. (b) SSA dynamics (without additional pulses): we observe very irregular, noisy behaviour. If one protein is expressed in high numbers the other is not and vice versa. Switching occurs irregularly without prior initiation. For SSA and ODE, the initial conditions are the same ($[A] = 10$, $[B] = 5$, $[A : B] = 0$).

starting with low concentrations $[A]$ and $[B]$ and remains stable at certain concentration levels over the monitored time whereas the SSA trajectories display periodical switches. When the state of the ODE solution is perturbed by injecting protein B the system immediately develops back into the stable state. Only for very large amounts of added protein concentrations (several hundreds or even thousends depending on the systems state at the time of injection), we can observe a short time period where protein B is present in a higher concentration than protein A, similar to the early dynamics shown in Figure 2(c). Thus, the GRN displays no bistable switching in the ODE regime. One-time injection of a sufficient number of protein A molecules during stochastic simulation let the system switch from low protein A level to high protein A level. However, injections of protein B did not necessarily lead to a switch back, irrespective of the number of molecules added to the system.

Figure 3(a) shows an evolved GRN that behaves in the stochastic simulation as a monostable switch with protein A at high level (about 150–250 molecules) and protein B at low level (0 molecules) as the only truely stable state. By injecting protein B molecules the state switches immediately into a state with very low numbers of molecules (< 10) for both A and B. Interestingly, the time the system remains in this state depends on the amount of protein B molecules added to stimulate the switch. Figure 3(b) and 3(c) demonstrate this for injections of 20 and 40 molecules, respectively. In this case the time is roughly twice as long for the second than for the first trajectory. From several simulations we got the impression that the larger the injection the longer the system stays in its state. However, this can only serve as a rule of thumb as large variations were observed as well.

4 Discussions

In this contribution we present two GRNs that our evolutionary approach produced. The resulting GRNs vary highly in their dynamics and not every solution exhibits a switching behaviour. According to the fitness function we search for GRNs where at least one protein dynamics displays a periodic but untriggered switching between a low and a high level (in terms of molecular numbers). The GRN in Figure 2(a) shows this form of switching behaviour for both proteins, that is, their molecular numbers mutually alternate between high and low levels. This does not occur because of any injection process or external control but because of the inherent noise which

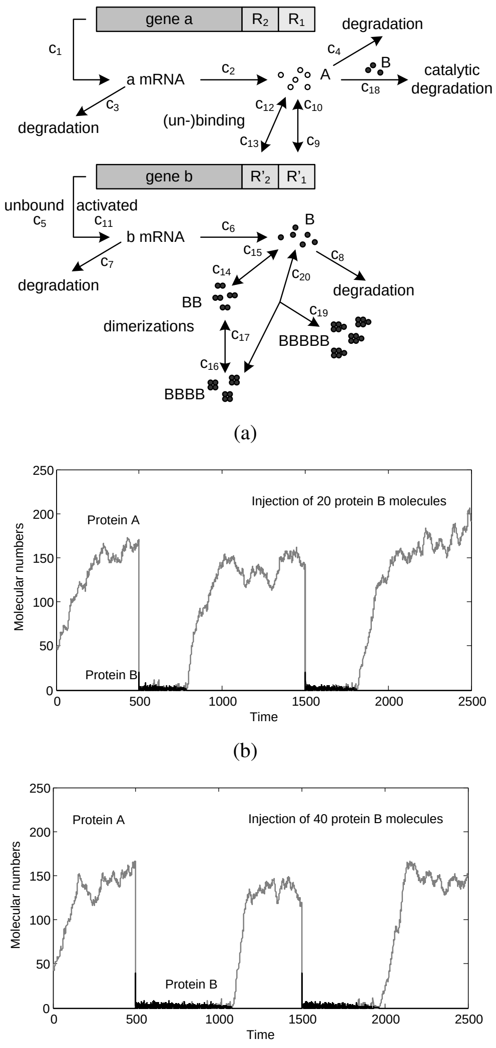


(a)

(b)

(c)

Figure 2: (a) Schematic representation of an evolved GRN. The evolved reaction constants are: $c_1 = 0.445, c_2 = 0.110, c_3 = 0.136, c_4 = 0.003, c_5 = 1.6, c_6 = 3.867, c_7 = 0.122, c_8 = 0.516, c_9 = 1.021, c_{10} = 0, c_{11} = 2.086, c_{12} = 0.013, c_{13} = 0.092, c_{14} = 0.92, c_{15} = 0.089$ and $c_{16} = 0.446$. (b) Simulation results showing the concentration dynamics of protein A and B. The (non-triggered) periodic switching behaviour is evident. (c) Solution of the corresponding ODEs with initial concentrations $[A] = [B] = 5$ and $[AB] = 0$. After injecting a strong concentration of protein B at time $t = 4000$ ($[B] = 800$) the system quickly evolves towards its previous stable state.



(a)

(b)

(c)

Figure 3: (a) Schematic representation of an evolved GRN performing as a stochastic monostable switch. It differs from the GRN in Figure 2(a) in the post-translational modifications and in the reaction rate constants ($c_1 = 0.939, c_2 = 0.208, c_3 = 0.155, c_4 = 0.008, c_5 = 0.047, c_6 = 0.022, c_7 = 0.312, c_8 = 0.044, c_9 = 0.852, c_{10} = 0.169, c_{11} = 0.251, c_{12} = 0.523, c_{13} = 0.013, c_{14} = 0.783, c_{15} = 0.222, c_{16} = 0.063, c_{17} = 0.494, c_{18} = 1.364, c_{19} = 0.121, c_{20} = 1.363$). (b) System dynamics when injecting 20 protein A molecules at time $t_1 = 1000$ and $t_2 = 2000$. Initial concentrations are: $[A] = 50, [B] = 0$ (c) This time, 40 protein A molecules are added to the system at t_1 and t_2. The system remains roughly twice as long in the state with $[A]$ at low level.

drives the dynamics backwards and forwards. Indeed, the corresponding ODE solution has only one equilibrium state and after transient stimulation the system develops back into this state.

The monostable GRN in Figure 3(a) shows interesting properties as well. Here, switches from high level (stable state) to low level (instable) molecule numbers of protein A do not occur randomly. Instead, they have to be induced by protein B injection. The switch back into the stable state is driven by noise, though. Apparently, the time period for which the system is not in its equilibrium depends on the injection.

Experiments with other monostable GRNs showed that often injections need to exceed a certain threshold to become effective. This threshold can depend on the molecular numbers of key proteins in the system. Also, the same dosage of injection can be more efficient, in that it keeps the monostable switch for a longer time in the nonstable state if it is spread over a certain time period. This might be interesting from an experimental perspective and needs to be analysed in more detail.

So far, we did not evolve real bistable switches. This is mainly because of the fitness function which does not put selection pressure on the evolution of such switches. Nevertheless, evolutionary runs with modified fitness functions suggest that it is difficult to find switches which are bistable but work with key molecules in low numbers. When dealing with low numbers of molecules there is a natural fluctuation due to stochasticity and the lower the numbers the larger the possibility of accidental switching (cf. Figure 3).

Whether the evolved GRNs in Figure 2(a) and 3(a) have a structure that occurs in nature remains to be seen. A comparison of this and other evolved solutions with known biological switches might lead to further insights.

This work underpins the necessity of stochastic simulation by evolving GRNs that perform as switches in the SSA regime (but not necessarily in the ODE regime) and might contribute to the finding of functional design principles and a better understanding of regulation in cells.

Acknowledgements

KB would like to thank the Australian Research Council for funding via the Federation Fellowship program.

References

A. Arkin, J. Ross, and H. H. McAdams. Stochastic kinetic analysis of developmental pathway bifurcation in phage λ–infected Escherichia coli cells. *Genetics*, 149(4):1633–1648, Aug 1998.

A. Becskei, B. Séraphin, and L. Serrano. Positive feedback in eukaryotic gene networks: cell differentiation by graded to binary response conversion. *EMBO J*, 20(10):2528–2535, May 2001.

M. B. Elowitz and S. Leibler. A synthetic oscillatory network of transcriptional regulators. *Nature*, 403 (6767):335–338, Jan 2000.

M. B. Elowitz, A. J. Levine, E. D. Siggia, and P. S. Swain. Stochastic gene expression in a single cell. *Science*, 297(5584):1183–1186, Aug 2002.

P. François and V. Hakim. Design of genetic networks with specified functions by evolution *in silico*. *Proc. Natl. Acad. Sci.*, 101(2):580–585, Jan 2004.

T. S. Gardner, C. R. Cantor, and J. J. Collins. Construction of genetic toggle switch in *escherichia coli*. *Nature*, 403(6767):339–342, 2000.

D. T. Gillespie. Approximate accelerated stochastic simulation of chemically reacting systems. *J. Chem. Phys.*, 115(4):1716–1733, 2001.

D. T. Gillespie. Exact stochastic simulation of coupled chemical reactions. *J. Phys. Chem.*, 81(25): 2340–2361, 1977.

J. Goutsias. Quasiequilibrium approximation of fast reaction kinetics in stochastic biochemical systems. *J. Chem. Phys.*, 122(184102):1–15, may 2005.

L. H. Hartwell, J. J. Hopfield, S. Leibler, and A. W. Murray. From molecular to modular cell biology. *Nature*, 402:C47–C52, 1999.

J. Hasty and J. J. Collins. Translating the noise. *Nature Genetics*, 31(1):13–14, May 2002.

W. J. Heuett and H. Qian. A stochastic model of oscillatory blood testosterone levels. To appear in Bulletin for Mathematical Biology, 2005.

H. Kobayashi, M. Kaern, K. Chung, T. S. Gardner, C. R. Cantor, and J. J. Collins. Programmable cells: Interfacing natural and engineered gene networks. *Proc. Natl. Acad. Sci.*, 101(22):8414–8419, 2004.

A. Leier, P. D. Kuo, W. Banzhaf, and K. Burrage. Evolving noisy oscillatory dynamics in genetic regulatory networks. In P. Collet, M. Tomassini, M. Ebner, A. Ekart, and S. Gustafson, editors, *Proceedings of the 9th European Conference on Genetic Programming (EuroGP 2006)*. Springer, 2006. In press.

H. H. McAdams and A. Arkin. Stochastic mechanisms in gene expression. *Proc. Natl. Acad. Sci.*, 94(3):814–819, Feb 1997.

D. M. Wolf and A. P. Arkin. Motifs, modules and games in bacteria. *Curr. Opin. Microbiol.*, 6:125–134, 2003.

Y. Yokobayashi, R. Weiss, and F. H. Arnold. Directed evolution of a genetic circuit. *Proc. Natl. Acad. Sci.*, 99(26):16587–16591, 2002. doi: 10.1073/pnas.252535999.

Network Entropy and Cellular Robustness

Thomas Manke*
*Max Planck Institute for Molecular Genetics
Ihnestr. 73, 14195 Berlin, Germany
manke@molgen.mpg.de

Lloyd Demetrius†
†Dept. of Organismic and Evolutionary Biology
Harvard University, Cambridge MA02138, USA
Second@else.where

Martin Vingron*
‡Max Planck Institute for Molecular Genetics
Ihnestr. 73, 14195 Berlin, Germany
vingron@molgen.mpg.de

Abstract

Here we present a novel approach to the study of networks and their resilience properties. Our work is based on a fundamental relation from dynamical systems theory which states that the macroscopic resilience of a steady state is correlated with the uncertainty in the underlying microscopic processes, a property which can be measured by entropy. Here we apply these ideas to the analysis of biological networks as obtained from large-scale protein interaction screens in yeast and *C. elegans*. In this context we characterize the diversity of possible pathways in terms of network entropy. Our analysis shows that knockouts of proteins with large contribution to network entropy are preferentially lethal. This observation is robust with respect to several possible errors and biases in the experimental data. Our analytical approach goes beyond the phenomenological studies based on local network observables, such as connectivity. It provides a rationale to study proxies of cellular resilience and to rank network elements (proteins) according to their importance within the global network context.

1 Introduction

Recent experimental efforts have highlighted the pervasiveness of molecular networks in biological sciences (Alm and Arkin, 2003). While a large number of molecular interactions and associations have been mapped qualitatively, we have yet to understand the relation between the structure and the function of biological networks which control the information flow and regulation of cellular signals. One particularly important functional characterisation is the resilience of an organism against external and internal changes (Stelling et al., 2004; Kitano, 2004), which, at the molecular level, amounts to perturbations in the network parameters. In recent experiments this resilience has been studied in direct response to gene deletions or RNA interference (Giaever et al., 2002; Kamath et al., 2003). It has been demonstrated that a large number of such network perturbations do not result in any phenotypic variation under a given experimental condition. This observation has led to a simple classification into 'viable' and 'lethal' proteins, according to whether the organism survives the removal of this component or not. In the following we also refer to the latter as "essential" proteins.

If network topology gives rise to behavioral complexity, one may ask if there is any topological correlate for lethality. In this work we present a natural framework to derive macroscopic parameters that characterise the topological and structural resilience of a network against random perturbations. Our analytical framework goes beyond the seminal studies of Albert et al. (2000) and Jeong et al. (2001), which addressed the same problem in terms of phenomenological parameters such as degree. The key idea and underlying assumption of our work is that many biological systems operate at steady state, where characteristic macroscopic observables (the "phenotype") remain constant for relatively long times. This, however, does not imply that the underlying microscopic variables (such as protein activities and concentrations) are static, but rather that their complex and continuous interplay results in a stable phenotype which can be experimentally observed. Indeed, it is the diversity and uncertainty of microscopic processes which determines the resilience of macroscopic steady states against random perturbations. In the context of dynamical systems this uncertainty is quantified by the dynamical entropy. The relationship between entropy and the robustness of macroscopic observables (their rate of return to steady state values) is the content of a fluctuation theorem (Demetrius et al., 2004), which states that changes in entropy are positively correlated with changes in robustness. As a great simplification, and in recognition of our ignorance about the actual molecular processes, we assume that the microscopic processes on the network are Markovian. This leads to the notion of *network entropy* as a global measure of pathway diversity and as a correlate of cellular resilience. In this paper we will demonstrate that this global description leads to a natural ranking of network elements (proteins) according to their contribution to network entropy. In terms of functional perturbation experiments, we will test the hypothesis that proteins with higher entropic contribution to the cellular network more frequently have a lethal phenotype when they are impaired (knock-out/knock-down). Previously this question has been addressed in terms of various notions of network centrality: degree Jeong et al. (2001), shortest path length Yu et al. (2004) and more recently betweenness Hahn and Kern (2005). Here we also provide a rationale for why these adhoc measures are sometimes convenient proxies for network resilience and how they could be extended.

2 Network Entropy and Entropic Ranking

First recall the definition of the dynamical entropy for a Markov process, $P = (p_{ij})$, which is given by (Billingsley, 1965)

$$H = -\sum_{ij} \pi_i p_{ij} \log p_{ij} \quad . \tag{1}$$

Here p_{ij} denotes the transition probabilities and the π_i are the components of the stationary distribution. There are many other ways to investigate complex dynamical systems through microscopic modelling, such as differential equations, but our simple stochastic description of dynamical uncertainty is based on random walks on the network and has a long tradition in the analysis of diffusive systems Berg (1993).

In our context the dynamical entropy of a Markov process characterizes the diversity of possible pathways and is related (through the fluctuation theorem) to the systems response to perturbations. If only the network topology is known, we associate the following process with a given adjacency matrix $A = (a_{ij})$:

$$p_{ij} = \frac{a_{ij} v_j}{\lambda v_i} \quad . \tag{2}$$

It has been shown (Arnold et al., 1994) that this process is the unique solution to a variational principle for the leading eigenvalue, λ, of the adjacency matrix. For irreducible matrices the components of the corresponding leading eigenvector, v_i, are all strictly positive. For Boolean matrices the process matrix of Eq.2 maximizes the entropy and provides the most parsimonious choice of p_{ij}. See (Demetrius and Manke, 2004) for a more detailed discussion.

In the following we utilize the decomposition of network entropy into contributions from all individual proteins

$$H = \sum_i \pi_i H_i \quad , \tag{3}$$

where H_i is the Shannon entropy associated with protein i. This decomposition suggests that network elements with a higher contribution to the overall entropy have a larger effect on the network's resilience and functionality when removed.

3 Biological Networks and Functional Studies

Here we analyse biological networks of protein-protein interactions for a single-cellular organism (budding yeast) and the multi-cellular worm (C.elegans) which we retrieved from public databases (Mewes et al., 2002; Chen et al., 2005). For both organisms this information is supplemented by functional studies of large-scale gene disruption experiments (Giaever et al., 2002; Kamath et al., 2003).

For *S.cerevisiae* we retrieved a bidirected interaction network of 3854 proteins with 11912 yeast-two hybrid interactions and 1170 essential proteins among the total set of 6203 proteins. The protein-protein interaction network of *C.elegans* consists of 2800 proteins and 8740 interactions. Of all the proteins with a recorded interaction 322 are classified as essential, because their inhibition resulted in a lethal phenotype. It should be noted that both interaction data and functional screens have a number of associated errors, resulting from experimental insufficiencies of large-scale studies and their limitation to certain environmental conditions.

4 Proteins with high entropic contribtion tend to be essential

The network data described in the previous section lends itself to a structural analysis, which has conventionally been done in terms of various connectivity measures Jeong et al. (2001); Yu et al. (2004); Hahn and Kern (2005). Here we utilize network entropy as a global characteristic measure and its decomposition according to Eq.3, which provides an alternative measure to rank the importance of proteins within the network. Figure 1 shows that proteins with high rank are more often essential than expected by chance.

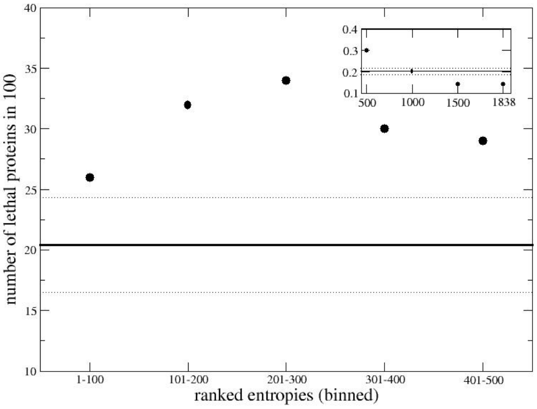


Figure 1: In the main figure we define 5 classes of *C.elegans* proteins according to their rank with respect to entropic contribution: 1-100, 101-200 .... In all these high ranking cases the fraction of essential proteins is significant. The expectation from 100 random proteins is shown as horizontal lines (± one standard deviation). The inset shows the same analysis for taking larger bins of 500 proteins. Again we can see an enrichment for high-ranking proteins, while there is an under-representation of essential proteins for proteins with small entropic contributions (for ranks > 1000)

5 Systematic Errors

As was mentioned above, the current large-scale data has sizable errors. Therefore we now investigate, whether the observed enrichment of essential proteins in top-ranking lists is robust against known sources of systematic errors.

First we extended the analysis of the previous section and evaluate the prevelance of essential proteins more systematically. For a given number, N_1 of top-ranking proteins we observe a certain number, N_{12} of essential proteins. This fraction can be translated into a probability (hypergeometric score) to observe such an overlap by chance, given a total of N proteins of which N_2 are essential.

In figure 2 we plot this probability against the number of top-ranking proteins for several setups. Our initial analysis (full circles) shows a steep decline of hypergeometic p-values and a systematic deviation from p-values obtained from randomized list of proteins (solid line). This reiterates the observation from the previous section. Next we tested the effect of false positive interactions by randomly deleting 50% of all edges (interactions) from the protein interaction network. This gave rise to a new ranking of proteins. Figure 2 illustrates that, despite this drastic change, the correlation is only moderately affected (triangles down). Missing interactions, on the other hand, can be expected to have a more pronounced effect. If, in a similar spirit, we increase the number of all interaction by 50% (random link addition), the entropic ranking will also change and the corresponding lethality assignment will become less and less predictive (triangles up). Notice though, that even with such large assumed error rates, the lethality assignment based on entropy is still significantly better than random (solid line).

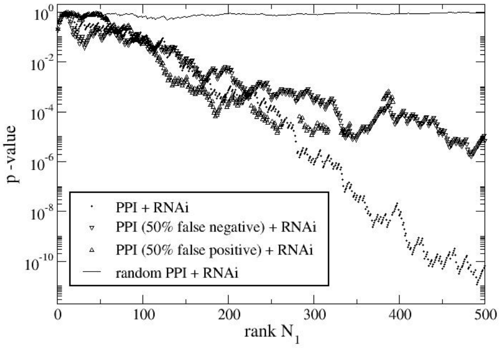


Figure 2: Randomized Networks. Here we analyse the correlation of entropic contribution and lethality assignment in the light of possible experimental errors for *C.elegans*. To this end we have added and removed a sizeable fraction of random interactions to the original data (full circles). While a large fraction of false positive errors (triangles down) does not significantly change the observed correlation, a large number of false negatives would reduce the significance of the observed correlation (triangles up). As expected, completely randomised interaction networks do not show any correlations as signified by the flat behaviour of the solid line.

Different cellular locations are known to influence the results of protein interaction screens. Therefore we have also tested our result against this possible bias by selecting randomized groups of “top-ranking” proteins, while maintaining their distribution with respect to cellular components. We find that the observed correlation is robust against this experimental artefact (data not shown).

Given the predominance of degree-based methods for network analysis, we also compared our novel importance measure, entropic contribution, to protein connectivity. While the two measures show a correlation for large degree, there are also clear differences, see Fig.3. Since entropy is a global measure, entropic contribution also takes into account the overall position of a protein within the network. This has the effect that proteins with highly connected interaction partners make a higher contribution to network entropy than proteins (with the same connectivity), but less connected neighbours.

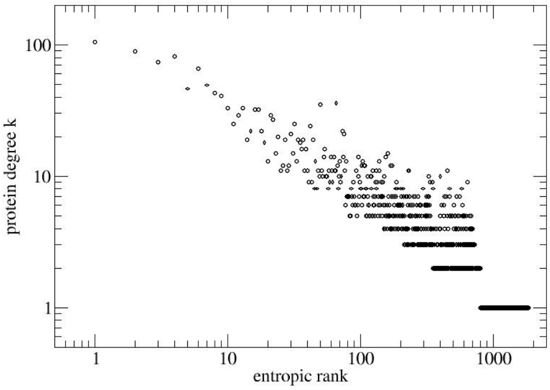


Figure 3: Here we show (for *C.elegans*) that connectivity and entropic contribution are correlated, but distinct from one another. For the process defined in Equation 2, proteins with high degree tend to have high entropic contribution. On the other hand there are also lowly connected proteins with high contribution to network entropy and hence robustness.

6 Conclusions

In summary, we have shown that the entropic characterisation of protein interaction networks can account for a significant fraction of proteins whose removal results in a lethal phenotype.

In our framework proteins are ranked according to their contribution to network entropy, which is a measure of microscopic uncertainty (pathway diversity) and is correlated with the macroscopic robustness of a dynamical system defined on the network.

We introduced a systematic method to assess the correlations between the entropic ranking scheme and phenotypic lethality data, and we have carefully tested the observed correlations against a number of possible errors. Our new conceptual framework provides a rationale to understand macroscopic resilience in the light of microscopic uncertainty, as characterized by entropy, rather than structural network observables. From this perspective, the observed enrichment of essential proteins in ranked lists of proteins has a natural and clear interpretation: proteins with higher contribution to cellular resilience are more often essential. Heuristic constructs, such as node degree, emerge as effective descriptors of dynamical properties, but our work also illustrates where one can go beyond such structural measures. Moreover, and in contrast to degree based-methods, our approach is extendable to networks where more quantitative data is available.

In the following we want to point to possible limitations of our approach. First, the phenotypic assessment of a gene disruption is usually done for one given condition and the observed correlation is strictly with respect to this single condition. It has been remarked that so-called viable proteins may actually play a significant role in untested environments and their disruption could cause lethal phenotypes. An exhaustive study of all possible conditions is clearly beyond experimental capabilities. Therefore we take the present lethality data as representative for other conditions and implicitly assume that the classification of lethal and viable proteins is at least robust against environmental changes.

A related problem concerns the static representation of interaction data which discards all dynamical dependencies. Just as many genes are expressed only under specific conditions, we also should think of different network realisations of an underlying blueprint which experimental interaction screens try to establish. Since the concept of entropy is based on the notion of dynamical diversity of the microscopic processes underlying the cellular states, we believe that this approach will ultimately be more fruitful than network characterisations which are solely based on topology. We should, however, stress that in the present application we relied exclusively on structural information of only a part of the complete cellular network - namely protein-protein interactions. Furthermore, we characterized the microscopic diversity through a Markov process that maximizes the entropy based on a Boolean adjacency matrix, rather than quantitative information about transition rates. Needless to say that actual processes may be different from this representative one.

To the extent that real processes resemble the one defined in this work, we can now better understand the importance of structural network observables as correlates of dynamical properties. We expect that structural properties will become less useful concepts for processes that deviate from the one with maximal entropy. Our approach is a first attempt to bridge these two domains and to address structural and dynamical questions in a single framework.

This situation can be likened to thermodynamics, where some properties of large systems can be effectively described by a number of macroscopic parameters, regardless of our ignorance about the microscopic processes. For equilibrium systems, this simplification is made explicit through relations between the Gibbs distribution over microstates and various macroscopic properties that can be derived from it Gibbs (1901). Formally, our work builds on an extension of the Gibbs formalism, which also applies to non-equilibrium systems at steady state Ruelle (2004). We implicitly assumed that the cellular processes on protein interaction networks fall into this larger class. If these assumptions hold, our approach should also apply to other complex networks, and there is hope that some systemic properties can be elucidated without having to resort to microscopic details.

Acknowledgements

This work has been supported by a grant from the German National Genome Research Network (NGFN).

References

Albert, Jeong, and Barabasi. Error and attack tolerance of complex networks. *Nature*, 406(6794): 378–82, Jul 2000.

Eric Alm and Adam P Arkin. Biological networks. *Curr Opin Struct Biol*, 13(2):193–202, Apr 2003.

L. Arnold, V. Gundlach, and L. Demetrius. Evolutionary formalism for products of positive random matrices. *Annals of Probab.*, 4(3):859–901, 1994.

Howard C. Berg. *Random Walks in Biology*. Princeton University Press, 1993.

P. Billingsley. *Ergodic Theory and Information*. Wiley, New York, 1965.

Chen *et al.* WormBase: a comprehensive data resource for Caenorhabditis biology and genomics. *Nucleic Acids Res*, 33(Database issue):D383–9, Jan 2005.

Lloyd Demetrius, Volker Matthias Gundlach, and Gunter Ochs. Complexity and demographic stability in population models. *Theor Popul Biol*, 65 (3):211–25, May 2004.

Lloyd Demetrius and Thomas Manke. Robustness and network evolution – an entropic principle. *Physica A*, 346(3-4):682–696, 2004.

Giaever *et al.* Functional profiling of the Saccharomyces cerevisiae genome. *Nature*, 418(6896): 387–91, Jul 2002.

J Willard Gibbs. *Elementary Principles in Statistical Mechanics*. Dover, New York, 1901.

Matthew W Hahn and Andrew D Kern. Comparative genomics of centrality and essentiality in three eukaryotic protein-interaction networks. *Mol Biol Evol*, 22(4):803–6, Apr 2005.

H Jeong *et al.* Lethality and centrality in protein networks. *Nature*, 411(6833):41–2, May 2001.

Kamath *et al* Systematic functional analysis of the Caenorhabditis elegans genome using RNAi. *Nature*, 421(6920):231–7, Jan 2003.

Hiroaki Kitano. Biological robustness. *Nat Rev Genet*, 5(11):826–37, Nov 2004.

Mewes *et al.* MIPS: a database for genomes and protein sequences. *Nucleic Acids Res*, 30(1):31–4, Jan 2002.

David Ruelle. *Thermodynamic Formalism*. Cambridge Mathematical Library, 2004.

Stelling *et al.* Robustness of cellular functions. *Cell*, 118(6):675–85, Sep 2004.

Yu *et al.* Genomic analysis of essentiality within protein networks. *Trends Genet*, 20(6):227–31, Jun 2004.

Effects of Dimensionality over Cooperation Dynamics

Ivette C. Martínez⋆
⋆Grupo de Inteligencia Artificial
Universidad Simón Bolívar
martinez@ldc.usb.ve

Klaus Jaffe†
†Laboratorio de Comportamiento
Universidad Simón Bolívar
kjaffe@usb.ve

Abstract

An Agent Based Model was used to explore the effects of simulating one and two dimensional grids over the dynamics of cooperation, under scenarios of biological evolution (BE) and cultural evolution (CE). Our results show that the way space is simulated does affect the dynamics of evolution. Interestingly, biological evolution was more susceptible to this effect that cultural evolution.

1 Introduction

Biologists, economists, computer scientists and physicists have all worked to further our understanding of human and animal cooperation. Yet different premises underlay these efforts. The main difference among them is the assumption that social behavior arrived through biological evolution among animals, and that culture and rational decision making is a principal driver of the evolution of cooperation and sociality among humans (Richardson et al., 2004). Human cooperation seems to be molded by both, cultural and biological forces (Kurzban and Houser, 2005).

Important differences between the dynamics of cultural evolution (Richardson et al., 2004; Ehrlich and Levin, 2005) and biological evolution (Nowak and Sigmund, 2004) exist. One important feature differentiating systems driven by biological and cultural evolution, is that transmission of information in BE is vertical (heredity), and that in CE is horizontal (imitation of the behaviour of the majority). This feature affects the speed information is transmitted, and is sufficient to explain important differences in the dynamics between both types of evolution (Jaffe and Cipriani, 2006). Here we want to explore the effect of the dimensionality of space on the different dynamics reported. To do so, we modify a one-dimensional spatial model (Cipriani and Jaffe, 2005) to study the differences between the dynamics of cooperative, group-forming individuals subject to a selective pressure (in this case predation). We based our model on the well known 'selfish herd' concept (Hamilton, 1971) and assume that cultural and biological dynamics is driven by natural selection on the phenotypes (i.e., roles) of individuals: cooperators and non-cooperators.

2 The Model and Experiments

Based on a cellular automata model that represents a population of interacting individuals with different social roles proposed by (Cipriani and Jaffe, 2005), we construct an agent-based model that incorporated environments with different spatial structures. Our initial implementation, made in Python, is for one-dimensional and two-dimensional toroidal grids environments of size 10000.

The majority rule implemented here to simulate cultural evolution (CE) assumed that individuals had a given probability of imitating the behavior of their neighbours. Behavioural traits were transmitted 'horizontally' via learning by imitation. We contrast this mechanism with biological evolution (BE) where learning does not take place and information was transmitted to offspring via hereditary rules. The summation of both mechanisms is a metaphor of species driven by both cultural and biological evolution.

We studied five scenarios characterized in our model by the way agents were allowed to interact. In CE, the rate T determined the probability an agent would imitate the behaviour (cooperate or not) of the majority of its neighbors. Production of new agents for CE was uniform ($50/50$) and T took three values: $\{0, 0.5, 1\}$. In the BE scenario, empty cells were replenish in proportional to the current number of individuals of each kind (cooperators and noncooperators) and $T = 0$. In the BE+CE scenario $T = 1$ and proportional replenish were applied..

Our experiments consisted in 5 series of simulations, corresponding to the described scenarios. In each series (101 simulations) we varied the cost of cooperation in steps of 0.01. Each simulation was run for 400 time-steps. For all simulations the "fitness differential" was 0.6 (the difference between the predation rate of isolated individuals (0.8) and that for cooperators being part of a group of cooperators (0.2)).

3 Results

The results of the different experiments are shown in figures 1 and 2 where each figure summarizes the results from simulations with environments of either a 1-dimensional grid 1 or a two-dimensional grid 2. Interestingly, the differences between the dynamics of the various scenarious explored were larger when we simulated a 1-dimentional grid than when using the 2-dimentional grid. The basic morphology of the resulting dynamics was not affected by the dimension of the simulated space. That is, BE has very sharp thresholds compared to CE and CE+BE was the strategy favouring most cooperation.

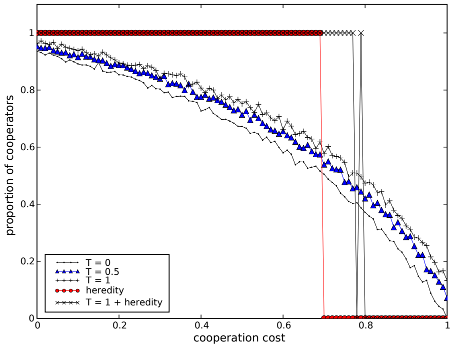


Figure 1: Influence of the cooperation's cost over the proportion of cooperators at the end of simulations. 1-dimensional grid.

4 Discussion

Our experiments confirmed the result obtained by (Jaffe and Cipriani, 2006). That is, we showed that the dynamics of CE and BE differed in very basic aspects. The fact that we used our own model implementation, confirms that this observed effects is not an artefact of the specific model implemented.

We also showed that the dimensionality of the simulated space, and thus also probably also the topology of the space (i.e. grids, networks, small world, scale free worlds, etc) will affect the dynamics under study, making this a complex subject to study.

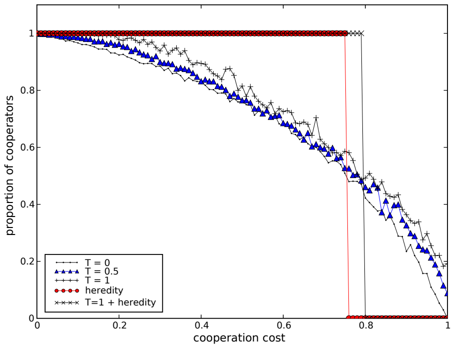


Figure 2: Influence of the cooperation's cost over the proportion of cooperators at the end of simulations. 2-dimensional grid.

Our results strongly suggest that simulations reporting on the evolution and/or dynamics of cooperation (or probably of anything) should specify if it is simulating BE or CE, avoiding future confusions when comparing results of simulations from different authors.

References

R. Cipriani and K. Jaffe. On the dynamics of grouping. In *Proceedings of the Fifth IASTED International Conference on Modelling, Simulation and Optimization*, pages 56–60, 2005.

P.R. Ehrlich and S.A. Levin. The evolution of norms. *PLoS Biology*, 3(6):e194, 2005.

W.D. Hamilton. Geometry for the selfish herd. *Journal of Theoretical Biology*, 31:295–311, 1971.

K. Jaffe and R. Cipriani. Nature and nurture, two different routes leading to sustainable cooperation. Submited, 2006.

R. Kurzban and D. Houser. Experiments investigating cooperative types in humans: A complement to evolutionary theory and simulations. *PNAS*, 102: 1803–1807, 2005.

M.A. Nowak and K. Sigmund. Evolutionary dynamics of biological games. *Science*, 303:793–799, 2004.

P.J. Richardson, J.E. Strassmann, and C.R. Hughes. *Not by Genes Alone: How Culture Transformed Human Evolution*. Chicago Univ. Press, 2004.

Community structure in group living animals

David Mawdsley[⋆]
[⋆]Department of Physics
University of Bath, BA2 7AY
`d.mawdsley@bath.ac.uk`

Richard James[†]
[†]Department of Physics
University of Bath, BA2 7AY
`r.james@bath.ac.uk`

Abstract

We present a technique using networks to detect intermediate level community structure within animal fission-fusion societies. The technique uses simulated annealing to optimise the quality of a proposed division of the network into communities. We also present a method that allows the statistical significance of the communities to be determined. We illustrate this technique by the detection of communities in systems of wild guppies and Galápagos sea lions. In each case, we show that this technique allows new levels of statistically significant structure to be revealed. In both cases, this allows new insights into the structure of the system under investigation.

1 Introduction

The social structure of animals in group living populations is likely to exert a profound effect of many aspects of each animal's life, as well as that of the whole population. For example, an animal's breeding and reproductive success, its foraging behaviour, the spread of disease and of information through a population will all be affected by with whom, and how frequently the individuals interact (e.g. Barnard (2004)). In many group living animals such contacts are made and broken frequently (fission-fusion groups). Despite the high frequency with which these individual associations occur, there is increasing evidence that robust non-random structures can exist (Whitehead et al., 2005). Detecting such structures give us new insights into the social and structural organisation of such systems, as well as underlying assortative tendencies that may drive them.

We present a method using networks which allows us to both detect structures at an intermediate level between that of the dyad and that of the population within systems of group living animals, and evaluate the statistical significance of such structures.

The results for wild populations of guppies and Galápagos sea lions are presented. Both systems are fission-fusion systems, where individuals are exchanged between groups with a timescale much less than the frequency with which group populations are observed.

2 Method

Traditionally, intermediate level structure in group living animals has been studied using techniques such as cluster analysis, which agglomerate animals based on some given similarity measure (e.g. Kaufman and Rousseeuw (1990)). We adopt an alternative, but complementary, approach and simply consider the structure of (repeated) interactions over an extended period of time. From this data, we construct a network. The presence of an edge in this network (representing an interaction between two animals) is determined *solely* from the observations of individual animals, without appealing to any external parameters.

Such a network is constructed from repeated censuses of the population. Animals observed in the same group on a given census are considered socially connected, and an edge is made between them. We combine many such censuses, each sufficiently separated in time from the others that they may be considered independent. We apply filters to the network to keep only strong interactions between dyads, and to remove animals that are only weakly associated with the population.

We use the technique of community detection to look for intermediate level structure. The idea of a community within a network is simply put; it is a region of the network that has a greater density of connections within it than to other parts of the network. The detection of communities within a network is, however, non-trivial, though several techniques have been proposed in recent years to do so (see Newman

(2004) for a review).

The size of systems that we have studied contain several hundred nodes. They are thus (by modern network analysis standards) quite small. This does, however, allow more sensitive, but computationally intensive techniques to be used than would be possible with larger networks. This is especially important in fission-fusion systems, where the structures that we seek may be subtle and difficult to detect. This, in turn, motivates the need to test for *significance* in the resulting structures that we find.

We maximise a measure, Q, that was originally proposed as a stopping criterion for earlier community detection algorithms (Newman and Girvan, 2004) that quantifies the quality of a set of communities.

Q is maximised via the well known optimisation technique of simulated annealing (Kirkpatrick et al., 1983). We impose an initially random division of community upon the network, and, via a series of trial moves attempt to find the global maximum of the function. The partition of communities that result thus represent our best effort at their detection. We have performed tests which show the accuracy and sensitivity of this technique outperforms existing community detection algorithms.

Significance testing

Almost all community detection algorithms, including the simulated annealing technique, will find some division of community, regardless of whether such a division is meaningful. Although large values of Q suggest a strong community structure, as noted by Guimerà et al. (2004), this isn't a sufficient condition for a *meaningful* community structure. We thus need to test the significance of the communities we find, to determine whether they are real, or simply artefacts of the process. We do this via a randomisation test (Manly, 1997).

We perform a simulation of the census rounds, "observing" the animals that were seen in each round at random, but preserving the group sizes that were observed. We thus assume random interaction between the animals. By comparing the true value of Q with the ones that result from repeated runs with the randomised network, we are able to determine the significance of the community structure we find.

The communities that we find in the guppy system are significantly assorted by body-length and the median depth of water that each animal in the community was observed. In previous studies, phenotypic assortment has been found at the level of the shoal, but our result suggests that this may also occur at a higher level of organisation.

Analysis of the sea lion data shows that geographically separated communities exist. If we apply our algorithm to each community as a separate entity, we can seek a further layer of sub-community structure, whose presence can not be explained by simple category assortment or space use, suggesting that there is evidence of genuine sociality in this system, revealed by our method.

Acknowledgements

We thank J. Krause and D. P. Croft of the University of Leeds for providing us with the guppy data, and J. B. W. Wolf and F. Trillmich from Universität Bielefeld for the sea lion data. We thank them all for stimulating and fruitful discussions.

References

C J Barnard. *Animal behaviour: mechanism, development, function, and evolution*. Prentice Hall, 2004.

R Guimerà, M Sales-Pardo, and L A Nunes Amaral. Modularity from fluctuations in random graphs and complex networks. *Phys. Rev. E*, 70:025101, 2004.

L Kaufman and P Rousseeuw. *Finding groups in data: an introduction to cluster analysis*. Wiley Interscience, 1990.

S Kirkpatrick, C D Gelatt, and M P Vecchi. Optimization by simulated annealing. *Science*, 220(4598): 671–680, 1983.

B F J Manly. *Randomization, Bootstrap and Monte Carlo Methods in Biology*. Chapman and Hall, London, 2nd edition, 1997.

M E J Newman. Detecting community structure in networks. *Eur. Phys. J. B*, 38:321–330, 2004.

M E J Newman and M Girvan. Finding and evaluating community structure in networks. *Phys. Rev. E*, 69:026113, 2004.

H Whitehead, L Bejder, and C A Ottensmeyer. Testing association patterns: issues arising and extensions. *Anim. Behav.*, 69:e1–e6, 2005.

Noise R Us: From Gene Regulatory Networks to WWW

Margaritis Voliotis*
School of Computing
University of Leeds, UK
mar@comp.leeds.ac.uk

Carmen Molina Parìs
Department of Applied Mathematics
University of Leeds, UK
carmen@maths.leeds.ac.uk

Liverpool B. Tanniemola
Department of Applied Mathematics
University of Leeds, UK
tannie@maths.leeds.ac.uk

Netta Cohen
School of Computing
University of Leeds, UK
netta@comp.leeds.ac.uk

Abstract

Gene regulatory systems are complex biological systems accounting for gene expression. Like many complex systems, they are subject to inherent noise as well as external perturbations posing a threat to their robust functionality. We review different types and sources of noise in the context of gene regulatory systems as well as different mechanisms that such systems have adopted to effectively deal with this deficiency. In so doing we consider whether there might be some link between how certain structural and architectural properties might have evolved due to a system's requirement to function under noise.

1 Introduction

Today's world is dominated by complex engineered systems of interacting components, ranging from electrical power grids to the World Wide Web (WWW) and the internet. Many of these complex systems are prone to random failures, noise or even deliberate attacks that can have devastating and far reaching implications. Indeed, two massive power failures in the summer of 2003 (in Italy and across North America) left many tens of millions of people without electricity and caused damages estimated in many billions of dollars. Such large scale events are typically due to cascading series of failures, reflecting the vulnerability of many of these systems even to local failure. Other relevant examples could be denial-of-service attacks that render specific parts of the WWW unreachable or even the spread of computer viruses through the internet.

In principle, our world is also in abundance of naturally occurring (as distinct form traditional engineered) complex systems. Such systems appear across the whole spectrum of life. A cell can be thought as a complex system of interacting biomolecules. Cells, in turn, combine to form tissues, organs and neural networks. Moving up one level, multicellular organisms can be regarded as systems consisting of a multitude of complex subsystems such as the metabolic and nervous systems. Finally, living organisms can be regarded as the fundamental components of ecosystems, forming prey-predator relationships, complex food webs and interactions with the environment.

An interesting property of biological or naturally occurring systems, that might be used as a basis for distinction from classical engineered systems, is that of adaptability. Many such systems have the ability to respond (or adapt) to environmental or internal perturbations and therefore can achieve greater robustness with regard to failures, attacks and noise.

In order to better understand the ability of biological systems to withstand or even exploit noise, it is important to pin down the sources and types of noise and their effects within specific biological contexts. This paper focuses on gene regulation networks. In particular, we present a review of a selection of recent work which provides an overview of noise in gene regulatory systems, manifestations of noise therein and implications for gene expression dynamics and long-term evolutionary processes. In doing so, this paper does not offer a comprehensive coverage of the above but rather attempts to demonstrate how certain design and organisational principles can effectively be used as noise barricades and may have evolved to do so

The remainder of the paper is organised as follows. Firstly, a brief introduction to the process of gene regulation is given. We then set out to review different

---

*Corresponding author

sources and types of noise that dominate gene regulatory systems as well as some of the mechanisms that are related to the ability of such systems to preserve robust and adaptive functionality. From an evolutionary perspective, such a discussion can serve as a basis for interesting questions concerning the role of noise in the evolutionary process. In this context, we propose a simple toy model, based on the work of (Kashtan and Alon, 2005), that may provide insight into the evolution of gene regulatory networks. Finally, we look at some examples of classical engineered systems and discuss how noise might be affecting their structure and topology.

2 Gene Regulatory Systems

Progress in molecular biology and experimental methods has paved the way for a more comprehensive, system-level understanding of cell function (Kitano, 2001). Indeed, many basic biological processes occurring within a cell have been extensively studied over the past decades and one can state that their basic functionality has been mastered, at least to some degree. However, addressing how biological processes are managed and synchronised, so as to produce robust functionality requires a more integrative, system-level approach.

One major and rather complex cellular system is that which regulates gene expression and determines the protein profile of the cell. Based upon interactions between DNA, RNA and protein molecules, the gene regulatory system effectively switches on and off the expression of genes to accommodate various intra-cellular needs and changing environmental conditions.

2.1 Gene Expression

Gene expression refers to the set of biomolecular processes that result in the production of proteins from their corresponding genes. Despite its rather complex nature, gene expression is a relatively well understood process (Orphanides and Reinberg, 2002) and can effectively be divided into two main steps: transcription and translation (Fig. 1).

During transcription genes (or sets of genes) are copied into intermediary (mRNA) molecules. This step usually involves the utilisation of specific regulatory proteins, known as transcription factors (TF), which bind to the DNA, either inducing (activating) or restraining (inhibiting) transcription. On the subsequent step of translation, the mRNA molecules are used as templates for the synthesis of proteins or other amino-acid chains. It should be noted that gene expression comprises of multiple interacting processes, of which transcription and translation form the basic universal core. Hence, modelling transcription and translation, simplifies the overall picture of gene expression but nonetheless captures its essence

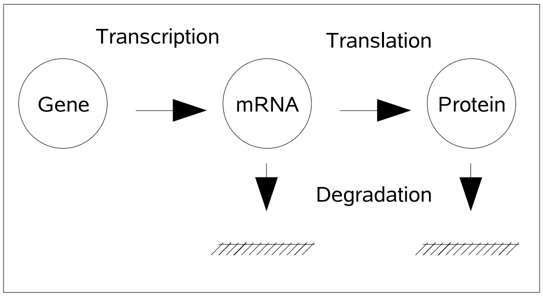


Figure 1: Simple model of the gene expression process. Intermediary (mRNA) molecules are produced from genes (transcription) and are then used as templates for the production of proteins (translation). Proteins and mRNA molecules are subject to degradation.

2.2 Transcriptional Regulation

It is known that the process of gene expression can be regulated at different levels by different means (Orphanides and Reinberg, 2002). Nonetheless at the simplest level, one often focuses only on regulation accomplished via TFs. This type of regulation, dubbed as transcriptional regulation, gives rise to transcriptional regulatory systems, which essentially constitute a part of gene regulatory systems[1]. This simplification is based on our notion that transcriptional regulation is the dominant type of gene regulation, at least in prokaryotes that are widely used as model organisms.

Nevertheless, transcriptional regulatory systems demonstrate a high degree of complexity. This inherent complexity partially arises from the relatively large size of such systems, which usually account for the regulation of a few hundred to a few thousand genes. Additional complexity stems from the fact that interactions between genes are not trivial. In particular, a single TF can regulate a number of different genes and not necessarily all in the same manner or with the same strength. On the other hand, it is also possible that the same gene is regulated by multiple

---

[1] In the remainder of this paper, the terms gene regulation and transcriptional regulation will be used interchangeably.

TFs, which can act either cooperatively or antagonistically.

2.3 From Network Representation to Dynamics

A network (or graph, consisting of nodes and edges) is often a useful abstraction for modelling complex systems. In the case of transcriptional regulatory systems, a network can be constructed by representing distinct genes as nodes, and denoting transcriptional regulation by directed edges between nodes. Examined under this perspective, it was shown that the transcriptional regulatory network of *Saccharomyces cerevisiae* (yeast) and *Escherichia coli* revealed certain architectural (macroscopic) properties, such as small world compactness and modularity (Shen-Orr et al., 2002; Maslov and Sneppen, 2002). On a finer (mesoscopic) level, it was also shown that the these networks were in significant abundance of simple building blocks, sometimes referred to as network motifs (Shen-Orr et al., 2002; Lee et al., 2002).

In addition to mere structural and topological observations, the network abstraction can effectively be used to examine how specific interconnectivity influences the dynamics of the system. In the simplest instance, one may interpret nodes as basic dynamical systems and edges as coupling between nodes. Thus, in the context of regulatory networks, nodes shall denote the actual gene expression process of different genes while edges encode how and with what strength the protein product (TF) of one gene regulates the expression of another.

3 Noise in Gene Regulatory Systems

In the example of the power failure in North America and Italy, reports suggested that possible causes could be local fluctuations in the demand load, as well as external factors such as storms damaging the power lines. In a similar way, robustness of transcriptional regulatory systems is threatened by external perturbations as well as inherent noise. Internal and external sources of noise may sometimes act on different timescales, however they both can have significant effects on the overall functionality of the system.

3.1 Noise in Gene Expression

Perhaps the most intuitive explanation behind noise at the microscopic level, is the fact that gene expression is essentially driven by biochemical reactions that are inherently stochastic processes. In our simplistic model of gene expression (Fig. 1) the rates at which transcription, translation and degradation proceed are not fixed, mainly due to stochastic fluctuations. However when the reactants, in our case DNA, RNA and protein molecules, are in great abundance, fluctuations are insignificant and the behaviour of individual steps can be modelled, with a great degree of accuracy, in a deterministic manner. Unfortunately, this is not the case in a typical intra-cellular environment where DNA, RNA, and protein molecules are usually present only in relatively small numbers. In such an environment, the effect of stochastic fluctuations becomes a significant factor and thus a stochastic framework is needed to fully capture the dynamics of gene expression.

Over the past years, various probabilistic models of gene expression have been proposed in literature (McAdams and Arkin, 1997; Thattai and van Oudernaarden, 2001). Summing up this theoretical work, origins of noise in gene expression can be effectively modelled by the following probabilistic events:

- gene activation and deactivation
- transcription initiation
- translation initiation
- decay of the mRNA and protein molecules.

The random activation and deactivation of a gene is usually attributed to random TF binding on DNA as well as to other events (e.g. chromatin remodelling). This switching in gene activity between *on* and *off* states leads to production of mRNA molecules in bursts of random sizes. Transcriptional bursting in turn leads to considerable fluctuations in the protein product, especially when transition rates between *on* and *off* states are slow (Blake et al., 2003; Raser and O'Shea, 2004).

Similarly to transcription, translation also occurs in random size bursts. This is due to the random lifetime of mRNA molecules during which several protein copies can be produced. According to the translational bursting mechanism a gene with high translational efficiency (number of proteins produced per mRNA molecule) is predicted to show a wide distribution of protein abundance, especially when mRNA molecules exist in low numbers (Thattai and van Oudernaarden, 2001). The above theoretical speculation for the translation mechanism was indeed verified by experimental work on *Bacillus subtilis* strains (Ozbudak et al., 2002).

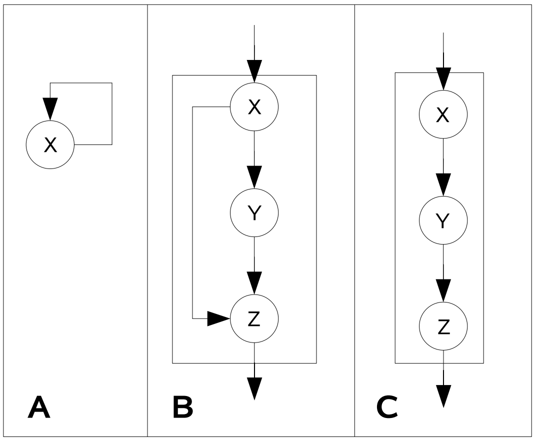


Figure 2: Examples of network motifs in gene regulatory networks (Lee et al., 2002; Shen-Orr et al., 2002). **A** Autoregulation. **B** Feed-forward Loop. **C** Regulatory Chain.

3.2 Noise in Network motifs

It has been proposed that transcriptional regulatory networks, both in prokaryotes and eukaryotes, are essentially assembled from basic structural units known as network motifs (Lee et al., 2002; Shen-Orr et al., 2002). From a dynamical point of view, network motifs can be thought of as simple signal transducing and/or controlling mechanisms. They are usually subject to some input regulatory signals (TF) and produce a corresponding output in the form of a protein product. Quite similar to logic gates in digital circuits, the functionality and dynamical behaviour of network motifs is dictated by their internal structure.

At this level of organisation, notions of intrinsic and extrinsic noise are of particular relevance. Focusing on a simple structural unit of the system, such as a network motif, the former type reflects noise produced internally by its components (as discussed above). This inherent noise propagates through the network motif leading to a noisy output. On the other hand, the term extrinsic noise accounts fluctuations in the output signal that originate from a noisy regulatory cue. Of particular importance is the notion of extrinsic noise and how such noise is propagated through the network motif. As discussed in section 4.2 the strong nonlinearity of the control architecture implies that in some cases external noise will be suppressed, whereas in other cases, it will be significantly amplified.

3.3 Network noise

Gene regulation networks attempt to capture the entire set of regulatory interactions within a cell. As such, they comprise an intricate web of network motifs, that are organised into larger structures (or modules) and form the global gene network of the cell. These cell networks can have characteristic topologies and statistics. At the network level, one often focuses on noise produced by fluctuating environmental and intra-cellular conditions that affect the overall stability and robustness of the regulatory system.

Environment is a basic factor compromising the stability of gene regulatory systems. Environmental conditions (e.g. temperature, pH) provide cues that can trigger the system, which in turn responds by modifying its expression pattern, or switching between alternate gene expression profiles. Therefore, environmental fluctuations are effectively transformed into noise in the system. Intra-cellular sources of noise can be ascribed to a wide variety of factors directly or indirectly affecting the process of gene expression. Such factors can be fluctuations in metabolite concentrations and variability in the activity of utility macromolecules (e.g. ribosomes and polymerases). It has also been known that cell specific characteristics such as cell size, cell age and the stage of the cell cycle can alter the gene expression profile (Kaern et al., 2005).

4 Noise Related Mechanisms

From the above discussion, it is perhaps striking that transcriptional regulatory systems are not only functioning under fluctuating environments, but are also comprised of unreliable, inherently noisy components. However, these systems demonstrate remarkably robust and adaptive functionality that sustains life. To accomplish that, gene regulatory systems utilise certain mechanisms, at their different organisational levels. Such mechanisms not only barricade the system against detrimental effects of noise but also have the ability to exploit noise in advantageous ways when this is possible.

4.1 Gene Level

At the component level, various distinct strategies can be adopted by a gene so as to achieve the same levels of protein expression (Fraser et al., 2004). These strategies essentially differ in the average rates in which transcription and translation are proceeding. One such strategy, for example, might yield high transcription rates while imposing low translation rates. Simply put, this strategy produces high numbers of mRNA molecules each one producing in turn low proteins numbers. On the other extreme a gene can accomplish the same protein numbers by producing limited mRNA molecules (low transcription rate) but each one yielding a high number of proteins (high translation rate). Finally, intermediate strategies can also be realised where both transcription and translation proceed at intermediate rates.

Following our discussion in section 3.1, the least noisy strategy is the one that maximises transcriptional efficiency while minimising the rate of translation, since such a combination minimises the effects of translational bursting. Indeed, Fraser et al. (2004) in their bioinformatics study discovered that the most essential genes of yeast showed a strong bias towards utilising this most uniform expression pattern. Moreover, the fact that not all genes follow the same strategy can be reasoned under the perspective that such a noise reducing mechanism is energetically expensive (more mRNA molecules have to be produced) and it should thus be adopted only by vital genes whose fluctuations might lead to deleterious effects (Fraser et al., 2004).

4.2 Network Motifs

Among the network motifs identified in the transcriptional regulatory network of yeast and *E. coli*, are those of autoregulation, the feedforward loop and the regulator chain (Lee et al., 2002; Shen-Orr et al., 2002). The properties of such structures have been recently studied both in theory and experimentally, using synthetically engineered gene circuits, providing us with deeper understanding of how noise is managed in gene regulatory systems.

Autoregulation can be thought of as an elementary form of control mechanism where the output of the gene expression process is fed back as a regulatory input (Fig. 2A). Autoregulation can either be negative or positive, depending on how the protein-product regulates its corresponding gene. Becskei and Serrano (2000) engineered regulatory circuits in *E. coli* cells to assess the importance of negative autoregulation with regard to noise. The results showed that the amount of protein produced from autoregulation-free circuits showed great variability among the cell population as opposed to the protein produced by the autoregulated circuit, which demonstrated significant stability. A similar study by Isaacs et al. (2003) focused on positive autoregulation, and demonstrated that the amounts of protein expressed under such a mechanism follow a bimodal distribution as a result of the inherent noise. In other words positive autoregulation amplifies noises to the point that two distinct phenotypes arise.

Feed-forward loops consist of a gene regulating another in a both direct and indirect manner, through a third gene (Fig. 2B). There are basically two types of Feedforward Loops: coherent ones where the sign of both regulation paths is the same, and incoherent ones where the signs of regulation are opposite. While both types were found in the studied networks the coherent type appears to be far more abundant (Shen-Orr et al., 2002). Summarising the theoretical work of Mangan and Alon (2003) coherent feed-forward loops may act as low pass filters for extrinsic noise, responding only to persistent input stimuli.

In the regulator chain motif, a gene regulates a second gene which in turn regulates a third one and so forth (Fig. 2C). The regulatory cascades can be of varying size and it has been suggested that they account for series of transcriptional events that happen sequentially (Lee et al., 2002). A recent experimental study by Hoosangi et al. (2005) dealt with noise in transcriptional regulatory cascades as a function of their length. In this study transcriptional cascades of several repressing steps were engineered in *E. coli* cells. The experimental results were consistent with the theoretical predictions that long cascades essentially act as extrinsic-noise filters, just as in the case of coherent feed-forward. However, intrinsic noise accumulates as the cascade length is increased.

In contrast to the noise filtering properties that some of the network motifs demonstrate, the noise amplification accomplished by others seems rather counter intuitive. Remarkably enough, however, the latter mechanisms can effectively be used to produce phenotypic diversification out of noise. This is particularly beneficial, especially in the case of bacterial populations, that can exploit diversity to survive and adapt under fluctuating environments. It has also be speculated that in a similar way cell differentiation is accomplished, in the developmental stages in multicellular organisms, form initially homogeneous cell populations (Kaern et al., 2005). It therefore appears that proper interaction between different genes is essential not only for the system to be shielded against noise but also for stochasticity to be exploited.

4.3 Architecture

Little is known about the actual architectural design of gene regulatory systems and much research is still in progress. Such enormous biological systems, consisting in general of thousands of components require vast experimental and theoretical work to be explored in their entirety. Even in the case of well studied model organisms, such as *E. coli* and yeast, their fully detailed regulatory networks are yet to be completed[2]. Circumventing this limitation, studies, focusing on subsets of the actual networks, reveal a rather modular design (Shen-Orr et al., 2002; Lee et al., 2002). However, for one to generalise these findings to the statement that gene regulatory systems are indeed modular, an implicit assumption is often made posing modularity as a design principle underpinning such biological systems.

Nonetheless, modularity is an interesting property with regard to noise. In particular, it can effectively be used by gene regulatory networks to isolate inherent noise into constrained subnetworks, thus minimising the risk of overall failures. Finally, even in extreme cases of deleterious failures, the functional decoupling that modularity provides, might under some conditions, give rise to graceful degradation.

Another interesting architectural property that was observed in the model regulatory network of yeast was that of small-world compactness (Maslov and Sneppen, 2002). This property, reflects the shortness of regulatory pathways, in the sense that only a few regulation steps are usually involved in the expression of a given gene. This property has perhaps an intuitive role if one takes into account the fact that noise propagates through the network. Therefore, minimal number of regulatory steps could prove to be an essential way of controlling the accumulation of noise.

5 Evolution

From an evolutionary perspective, the above discussion can serve as a basis for interesting questions concerning whether and how the notions of adaptation and evolution are linked to noise at every level of organisation. For example at the gene level, Fraser et al. (2004) provided strong support that certain vital genes have evolved towards utilising certain expression strategies that effectively reduce inherent noise. One might also consider alternative ways of gene regulation, other than transcriptional, and examine their noise properties. In doing so, deeper insight can be gained on whether mechanisms controlling inherent noise are subject to evolutionary pressure.

Focusing on the mesoscopic and macroscopic organisational levels one can go even further seeking ways in which evolution may have affected the topology of gene regulatory networks. Although such an issue is still open to discussion, one can examine the different viewpoints and draw some general conclusions. For instance, one may ask whether modularity and network motifs could evolve purely by mutational drift under neutral evolution. This would imply that such traits provide no selective advantages for the organism. Alternatively, such traits could have been subject to selection if, indeed, they offer some evolutionary advantage. In the latter case noise might prove quite significant, since as discussed above, there are strong indications that modularity and internal structures, such as network motifs, provide a framework that can effectively deal with and/or reduce noise.

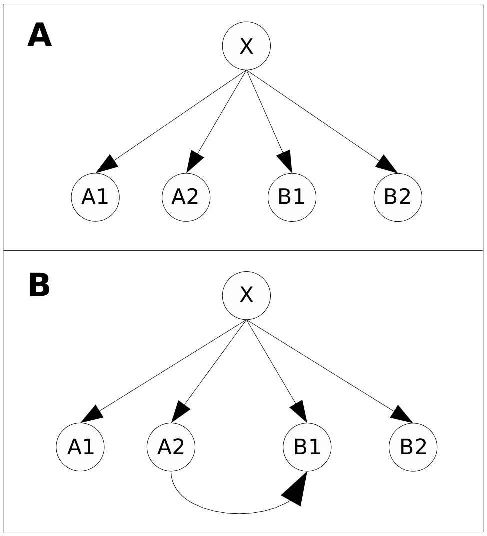


Figure 3: Simple toy models of regulatory networks with one TF. **A** Gene X activates the four genes. **B** Regulated genes can also interact with each other.

In an attempt to model and gain insight into such evolutionary processes, it has been proposed that modularity and network motifs might have spontaneously evolved as a result of an ever-changing environment (Lipson et al., 2002; Kashtan and Alon, 2005). In the example of Kashtan and Alon (2005), the environment defines modular goals, consisting of basic subgoals. As these goals are varied it was shown that modularity was evolved to make the system more adaptable to these changes. One may ask whether such a proposition holds for the case of transcriptional regulatory networks.

Simple toy models of a gene regulatory network and *in silico* evolutionary simulations might be used to demonstrate the effectiveness of different network

---

[2] RegulonDB (http://regulondb.ccg.unam.mx), an online database of transcriptional regulation in *E. coli*, currently includes 139 experimentally verified TFs.

structures in performing modular tasks. Let us consider a simple regulatory network where gene X encodes for a TF, which is actively regulating four genes, namely A1, A2, B1 and B2 (Fig. 3A). The biological function of these genes is not independent but assumed to be coupled in some way. For example, genes A1 and A2 could cooperatively metabolise substance A whereas genes B1 and B2 could metabolise substance B. On a more complex network, regulatory interaction can also exist between the metabolising genes (Fig. 3B) and indirect feedback can reach the TF X (not shown). The initial regulatory network does not demonstrate high modularity and one can readily think of a more modular version, where for example genes B1 and B2 are regulated separately from genes A1 and A2, through a second TF (Fig. 4A-B). Not all network structures achieve the same efficiency under similar conditions, since they cannot reproduce exactly the same expression profiles. For example, in the case where the metabolites A and B act independently, the network in Fig. 4A can be more efficient than that in Fig. 3A, since it can independently regulate the production of different subsets of genes.

Following the example of Kashtan and Alon (2005), mutations in our simple model will not account for changes in the biological functionality of genes (i.e. metabolism) but affect the structure of the network by removing or adding regulatory relationships between genes. Taking the model one step further, one can also assume that such mutations might as well affect the dynamics of the system by altering the rates which govern gene expression. In addition, rare events of gene duplication can also be incorporated. Such events affect the topology of the network by duplicating nodes while preserving the relationships between them.

Now suppose that under a typical steady environment, substances A and B are present at fixed (possibly different) concentrations and the evolutionary process can optimise the parameters of the system (rates of gene expression, and interactions) for the cell to be energetically satisfied. However, if our environment is not constant, in the sense that concentrations of substances can fluctuate, a different network topology may be better suited. This notion of a changing environment can be thought of as a form of noise acting at different timescales, even at an evolutionary one. Under a noisy environment, the mere optimisation of expression rates may be insufficient, and evolution may act on the topology of the network, possibly giving rise to more modular topologies

More generally, genes A1, A2, B1 and B2 of our

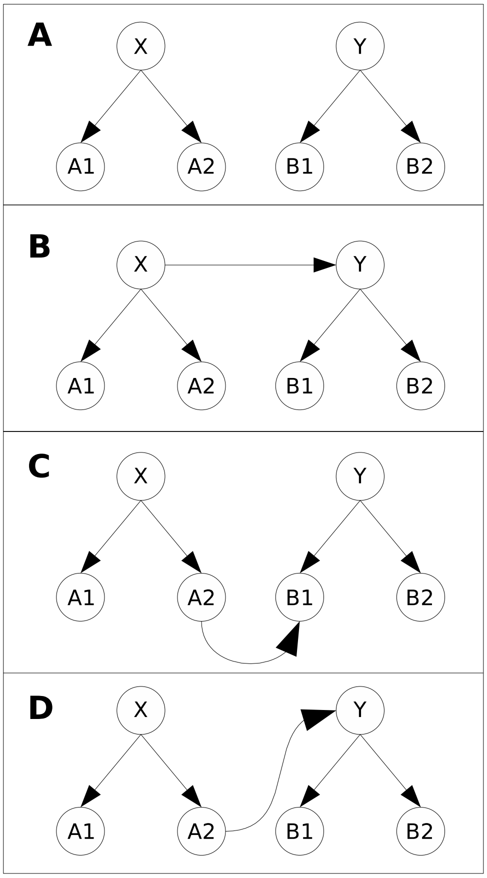


Figure 4: Simple toy models of regulatory networks with two TFs (A-B) and up to 6 TFs (C-D). **A** Two independent subnetworks where TFs X and Y activate genes (A1/A2) and (B1/B2) respectively. **B** Example demonstrating a possible regulatory interaction between the TFs. **C** Example demonstrating a possible transcriptional interaction between regulated genes. **C** Example demonstrating a possible feedback interaction between the regulated genes and their regulators. Combinations of the possible kinds of interaction **B-D** are also possible.

model may take on any role, and may even be TFs themselves. In that case, we may also envision regulatory interactions as illulstrated in Fig. 4C-D. In all these cases, the same principles apply: changes in the environment may provide selective pressures that eventually manifest themselves in the topology of the network.

Comparing evolutionary simulation with and without environmental noise can lead to some general insights on whether the evolutionary design of such toy models is affected by noise. In fact, one can go even further examining a more complicated toy model that also accounts for inherent noise and determining how this source of noise might affect the topology of the network in the long run.

6 The WWW and Power Grids

Similar to gene regulatory systems, classical engineered systems are also prone to noise. Surprisingly enough, some of these systems also demonstrate basic structural and organisational commonalities with biological systems, such as scale-free architectures, small world interconnectivity, modularity and abundance of network motifs (Dorogovtsev and Mendes, 2003; Milo et al., 2002). Therefore, it is perhaps interesting to try and generalise the above discussion on gene regulatory networks and examine whether noise, across biological and engineered systems, might have any relevance to the emergence of such structural properties. We pick the WWW and electric grids as two examples.

The WWW is essentially a vast information platform that enables efficient storage, retrieval and exchange of information across a network of nodes (web pages) and edges (hyper links). Fluctuations in the demand load of specific information can compromise its accessibility, as they might result in information retrieval latency or even unavailability due to limitations of the underlying communication network (the internet). To secure the WWW against this type of noise, certain measures have been adopted including mirror links and cached information on different (independent) web servers[3]. These measures are effectively altering the network topology since they involve the creation of new nodes and links and thus may give rise to certain structural properties. In fact, the creation of mirror sites can be likened to gene duplications in evolution. However, perhaps unlike gene regulatory networks, the WWW is also under a rapid evolutionary process, with information being published and/or withdrawn and links being created and/or removed on at least as fast a time scale. This process is has a dramatic effect on network topology and is likely to mask the topological effects due to mirroring and caching. Thus in general the question of whether noise might be related to the emergence of certain structural network properties seems somewhat obscure in the case of the WWW.

A more intuitive and pronounced example might be that of power grids, where growth takes place on a much slower time scale, so as to meet the needs of growing demand, population centres and industrial development. Even for a relatively static network, power grids are usually subject to load demand fluctuations as a consequence of various external factors (e.g. seasonal temperature fluctuations). Since noise in this case cannot be effectively controlled one intuitive way of minimising its effects is by adopting a modular grid design so that probable blackouts can be localised. Indeed, studies of power grids have revealed a highly clustered design (Dorogovtsev and Mendes, 2003), which has emerged during their evolution, in part due to our need to secure these systems to intrinsic and extrinsic sources of noise, as well as to direct insults.

7 Conclusions

Understanding how complex biological systems, such as gene regulatory systems, control noise, and how they evolved to accomplish that, is a vital step towards their structural and functional understanding. However, it can also provide valuable lessons for the design of complex engineered systems. Most engineered systems were specifically designed to utilise relatively reliable components. Nonetheless, even rare events may, under certain circumstances, lead to cascading failures, whether such failures are due to an internal component or external effects. In order to shield a system against such rare events, lessons learnt and design principles gleaned from biological systems may prove useful. In particular, the prevalence of modularity and small scale control structures in a range of engineered complex systems, could be suggestive of possible relevance of biological networks. It is hoped, therefore, that further research into the structure and function of biological systems, as compared with artificial or engineered ones, and specifically such systems' ability to effectively handle noise, will lead to practical applications in the design and regulation of complex systems.

---

[3] Note, of course, that the creation of mirror sites cached pages etc. is motivated by a combination of facts of which network noise is only one.

Acknowledgements

NC was funded by an EPSRC grant (EP/C011953/1).

References

A. Becskei and L. Serrano. Engineering stability in gene networks by autoregulation. *Nature*, 405: 590–593, 2000.

W. J. Blake, M. Kaern, C. R. Cantor, and J. J. Collins. Noise in eukaryotic gene expression. *Nature*, 422: 633–637, 2003.

S. N. Dorogovtsev and J. F. F. Mendes, editors. *Evolution of Networks. From Biological to the Internet and WWW.* Oxford University Press, 2003.

H. B. Fraser, A. E. Hirsh, G. Giaver, J. Kumm, and M. B. Eisen. Noise minimization in eukaryotic gene expression. *PLoS Biology*, 2(6):834–838, 2004.

S. Hoosangi, S. Thiberge, and R. Weiss. Noise characteristics of feed forward loops. *Proc. Natl. Acad. Sci. USA*, 102(10):3581–3686, 2005.

F. J. Isaacs, J. Hasty, C. R. Cantor, and J. J. Collins. Prediction and measurment of an autoregulatory genetic module. *Proc. Natl. Acad. Sci. USA*, 100 (13):7714–7719, 2003.

M. Kaern, T. C. Elston, W. J. Blake, and J. J. Collins. Stochasticity in gene expression: from theories to phenotypes. *Nature Reviews Genetics*, 6:451–464, 2005.

N. Kashtan and U. Alon. Spontaneous evolution of modularity and network motifs. *Proc. Natl. Acad. Sci. USA*, 102(39):13773–13778, 2005.

H. Kitano, editor. *Foundations of Systems Biology*. The MIT Press, 2001.

T. I. Lee, N. J. Rinaldi, F. Robert, D. T. Odom, Z. Bar-Joseph, G. K. Gerber, N. M. Hannett, C. T. Harbison, C. M. Thompson, I. Simon, J. Zeitlinger, E. G. Jennings, H. L. Murray, D. B. Gordon, B. Ren, J. J. Wyrick, J. Tagne, T. L. Volkert, E. Fraenkel, D. K. Gifford, and R. A. Young. Transcriptional regulatory networks in saccharomyces cerevisiae. *Science*, 298:799–804, 2002.

H. B. Lipson, J. B. Pollack, and N. P. Suh. On the origins of modular variation. *Evolution*, 56(8):1549–1556, 2002.

S. Mangan and U. Alon. Structure and function of the feed-forward loop network motif. *Proc. Natl. Acad. Sci. USA*, 100(21):11980–11985, 2003.

S. Maslov and K. Sneppen. Specificity and stability in topology of protein nwtworks. *Science*, 296:910–913, 2002.

H. H. McAdams and A. Arkin. Stochastic mechanisms in gene expression. *Proc. Natl. Acad. Sci. USA*, 94:814–819, 1997.

R. Milo, S. Shen-Orr, N. Itzkovitz, N. Kashtan, D. Chklovskii, and U. Alon. Network motifs: Simple building blocks of complex networks. *Science*, 298:824–827, 2002.

G. Orphanides and D. Reinberg. A unified theory of gene expression. *Cell*, 108:439–451, 2002.

E. M. Ozbudak, M. Thattai, I. Kurtser, A. D. Grossman, and A. van Ourdanaarden. Regulation of noise in the expression of a single gene. *Nature genetics*, pages 69–73, 2002.

J. M. Raser and E. K. O'Shea. Control of stochasticity in eukaryotic gene expression. *Science*, 304:1811–1814, 2004.

S. S. Shen-Orr, R. Milo, S. Mangan, and U. Alon. Network motifs in the transcriptional regulation network of escherichia coli. *Nature genetics*, 31: 65–68, 2002.

M. Thattai and A. van Oudernaarden. Intrinsic noise in gene regulatory networks. *Proc. Natl. Acad. Sci. USA*, 98(15):8614–8619, 2001.

Statistical Analysis of Dynamic Graphs

Xiaomeng Wan
*Faculty of Computer Science
Dalhousie University
Halifax, Canada
`xwan@cs.dal.ca`

Jeannette Janssen
†Dept. of Mathematics and Statistics
Dalhousie University
Halifax, Canada
`janssen@mathstat.dal.ca`

Nauzer Kalyaniwalla
‡Faculty of Computer Science
Dalhousie University
Halifax, Canada
`nauzerk@cs.dal.ca`

Evangelos Milios
§Faculty of Computer Science
Dalhousie University
Halifax, Canada
`eem@cs.dal.ca`

Abstract

Communications between large numbers of individuals can be modeled as a dynamic graph. The graph is the integrated effect of the individuals acting autonomously. To identify and analyze communication patterns, we study dynamic graphs by examining the global behaviour of local, vertex-specific measures. In this paper, we introduce novel vertex-specific measures, and apply scan statistics to examine the global extremes of these measures. We apply our methods to a set of email data, and show that the different measures offer complementary views of the data.

1 Introduction

A dynamic graph is a graph whose edges and vertices may appear and disappear over time. Examples are phone call graphs, email communication graphs and graphs representing visits of web pages by users. In these graphs, vertices represent entities and edges represent communication transactions. The variation of dynamic graphs over time can be used to profile what normal behaviour is, which is the basis for detecting anomaly and predicting future behaviours. Cortes et al. (2003) study phone call graphs, and present ideas on how to predict behaviour based on historic data. Priebe et al. (2005) introduce the use of scan statistics to study dynamic graphs. They study an e-mail communication graph using a density-based scan statistic, with the aim of detecting anomalies.

The dynamic graphs that are of interest to us can be considered as social networks. In most studies of such graphs, the dynamic nature of the graphs is summarized into the formation or weighting of a graph representing a general notion of "connectedness" or "contact" between vertices over the time period considered. This summarized graph can then be studied with the tools of social networks.

Our approach is, instead, to consider the dynamic graph as a time series of graphs, and to study it by focusing our attention on a number of locality measures derived from the links present in the neighbourhood of a specific vertex. We present a number of promising measures targetting different features of communication patterns. Local statistics are generated for these measures and their extremes are identified and analysed with scan statistics. Those nodes generating extremes which deviate from the general trend can be considered as anomalies and are worth detailed investigation. We applied our methods to a large collection of email data; results are presented and discussed in Section 4.

2 Locality measures

The dynamic graphs we consider are all derived from communications between individuals. Hence the graph is the global result of a large number of individual actions. It therefore stands to reason that we can model global behaviour by analyzing and integrating the local behaviour of each vertex. For each vertex, a variety of time-dependent locality statistics can be defined. A statistical summary of the behaviour of these statistics over time can be used to create a vertex-specific signal. The set of all signals can be used to model normal behaviour, and thus to classify vertices and filter out noise.

Specifically, a dynamic graph is considered as a time series of static graphs. Typically, the time interval in which communications have been observed is divided into shorter intervals. The static graph corresponding to each interval will have a link between vertices if an interaction took place between those vertices in that interval. The locality measures considered are all defined with respect to the edges present in the neighbourhood of a vertex. The *k-th order neighbourhood* of a vertex v consists of all vertices that are at most k “hops” away from v. We distinguish the *dynamic neighbourhood*, which consists of all vertices that received a link from v in a specific time interval for $k = 1$, and the *permanent neighbourhood*, which consists of all vertices that received at least one link from v during the whole time period under consideration for $k = 1$.

A time dependent locality measure based on the density of links in the dynamic neighbourhood of a vertex was introduced by Priebe et al. (2005). For $k = 1, 2$, their measure is the number of edges present in the k-th order dynamic neighbourhood of a vertex v. For $k = 0$, the measure is defined as the number of edges originating from v. Interesting results can be obtained if this measure is compared with a density measure derived from the permanent neighbourhood. For $k = 0$, permanent and dynamic neighbourhood density measures are equal, but for $k \geq 1$ they can catch different aspects of a vertex’s behaviour.

Sometimes, a shift in communication patterns may occur without a change in activity level. To catch such shifts, we introduce a *novelty measure*. This measures the number of “new” links in a neighbourhood, i.e. links that have not been observed in a fixed number of previous intervals. Note that the novelty measure must be defined with respect to the past τ-week neighbourhood.

To capture the behaviour of locality measures over time, the running mean and standard deviation can be computed for each measure, and used to standardize the signal. More specifically, let $\Psi_{k,t}(v)$ be a time-dependent locality measure defined with respect to the k-th neighbourhood. Then, the mean of locality statistic based on the recent history of the previous τ time intervals is defined by (Priebe et al., 2005):

$$\hat{\mu}_{k,t,\tau}(v) = \frac{1}{\tau} \sum_{t'=t-\tau}^{t-1} \Psi_{k,t'}(v) \tag{1}$$

The variance of locality statistic based on the recent history of the previous τ time intervals is defined by:

$$\hat{\sigma}^2_{k,t,\tau}(v) = \frac{1}{\tau - 1} \sum_{t'=t-\tau}^{t-1} (\Psi_{k,t'}(v) - \hat{\mu}_{t,\tau}(v))^2 \tag{2}$$

The vertex-standardized locality statistic is defined by (Priebe et al., 2005):

$$\hat{\Psi}_{k,t}(v) = \frac{\Psi_{k,t}(v) - \hat{\mu}_{k,t,\tau}(v)}{\max(\hat{\sigma}_{k,t,\tau}(v), 1)} \tag{3}$$

3 Scan statistics

A common technique to detect local anomalies in behaviour by a global analysis is the use of scan statistics. Scan statistics are commonly used in signal analysis, and in the detection of anomalies in localized health data (Glaz et al., 2001). Priebe et al. (2005) first applied scan statistics techniques to the local density measures described in the previous sections. The idea behind a scan statistics approach is to study a large number of local measures by studying the extremities of its values over all localities.

In our case, it makes sense to study both the maximum and the minimum of the standardized locality measures. More precisely, the statistic studied is the maximum (minimum) of the vertex-standardized locality measure which is defined by (Priebe et al., 2005):

$$\tilde{M}_{k,t} = \max_{v} \tilde{\Psi}_{k,t}(v) \tag{4}$$

The minimum of vertex-standardized locality measure is defined by:

$$\tilde{M}'_{k,t} = \min_{v} \tilde{\Psi}_{k,t}(v) \tag{5}$$

Note that the maximum of the standardized locality measure represents a sudden increase in activity, while a minimum represents a sudden drop. To determine whether or not certain values of this maximum (minimum) represent an anomaly, this statistic is itself temporally normalized as follows (Priebe et al., 2005).

$$S_{k,t} = \frac{\tilde{M}_{k,t} - \hat{\mu}_{k,t,\ell}}{\max_{k,t,\ell}(\hat{\sigma}_{k,t,\ell}, 1)} \tag{6}$$

Where $\hat{\mu}_{k,t,\ell}$ and $\hat{\sigma}_{k,t,\ell}$ are the running mean and variance of $\tilde{M}_{k,t}$ defined by:

$$\hat{\mu}_{k,t,\ell} = \frac{1}{\ell} \sum_{t'=t-\ell}^{t-1} \tilde{M}_{k,t'} \tag{7}$$

$$\hat{\sigma}^2_{k,t,\ell} = \frac{1}{\ell - 1} \sum_{t'=t-\ell}^{t-1} (\tilde{M}_{k,t'} - \hat{\mu}_{k,t,\ell})^2 \tag{8}$$

4 Experimental setup and results

We apply the methods described above to a large collection of email data from the Faculty of Computer Science at Dalhousie University. The data are derived from the log files of the email server of Faculty of Computer Science covering the period from May 2004 to September 2005. There are 16,580 email addresses involved with 1,500 active accounts (defined as sending emails to more than five distinct users during the period under study). The email addresses are anonymized before being used in the study, but the categories of email addresses (faculty, student, staff, mailing lists, ...) are preserved.

We divide the data into disjoint, one-week intervals. The locality statistics are calculated for the vertices in each window. Using the number of edges for each vertex as the locality statistic (for k = 0, 1, 2), we compute scan statistics based on *(i)* the original number of edges and *(ii)* the novelty measure, as described in Section 2. Figure 1 below shows the vertex-standardized maximum results for the novelty measure.

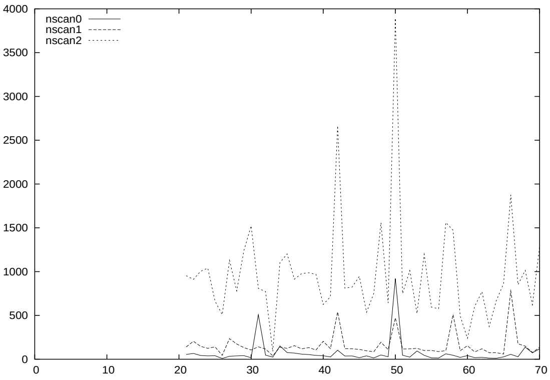


Figure 1: Time series of standardized scan statistics and max degrees for k = 0,1,2 with locality statistics as new links (Novelty) on CS data

The peaks in Figure 1 correspond to dormant or low activity vertices that suddenly come alive. The peaks in scan0 (degree of the vertex), are due to mailing lists that suddenly start transmitting after not sending any messages in the previous τ weeks. On the other hand, the peaks in scan2 are, in general, caused by low activity vertices communicating with a very high volume vertex *viz.* an un-moderated mailing list. The situation for scan1 is more complex, since the first neighbourhood of a very active node can mask the signal for a node that ultimately shows a prominent peak in scan2.

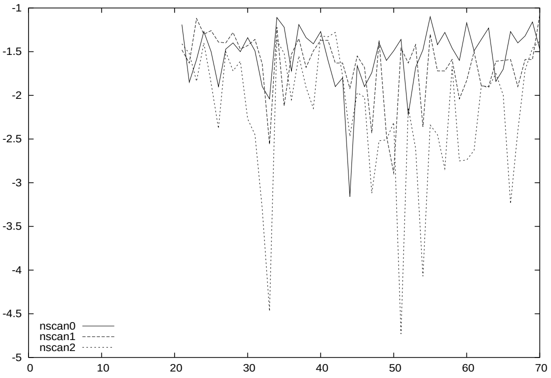


Figure 2: Time series of standardized scan statistics and min degrees for k = 0,1,2 with locality statistics as new links on CS data

The minima of the novelty measure are shown in Figure 2. In this plot, minima in scan0 simply show the sudden drop in activity of a normally active node, the effect is purely individual, but may be of interest. Since scan2 captures a large group of nodes, the minima in scan2 correspond to a general drop in activity. The pronounced minima in scan2 correspond to Christmas (week 32), the first week of May (week 52) which falls in the interval between the winter and summer terms, and the end of August (week 67) all of which correspond to general drop in activity around the university.

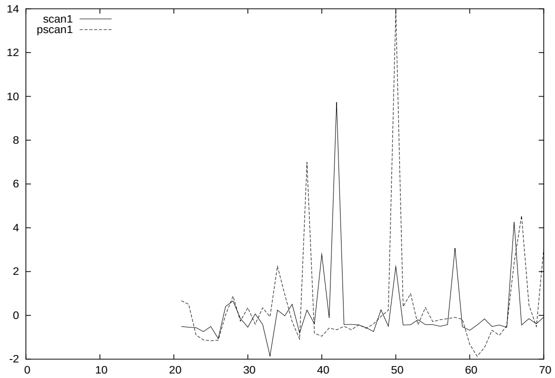


Figure 3: Time series of temporally-normalized standardized scan statistics and max degrees for k = 1 with locality statistics as links for dynamic and permanent windows on CS data

Figure 3 compares the temporally-normalized standardized maximum results of the dynamic and permanent windows. The two measures show different peaks. On closer examination of the data we notice that every peak for the moving window (the scan statistic) captures mailing lists, which are principally bursty vertices. The peaks of the permanent window however, all correspond to individual users. The effect is clearly due to the scan statistic's susceptibility to picking up on bursty behaviour. Consequently, bursty nodes mask the underlying changes in communication patterns of individual users which might be of interest. This effect points to separating the nodes into homogeneous groups and then using the machinery of scan statistics to look for anomalies within the groups.

These are preliminary results, which, however, indicate that locality measures are indeed a promising tool to analyze dynamic graphs. Currently, more experiments are being performed to study the effect of dynamic versus permanent neighbourhoods, and to compare different varieties of the locality measures. In the future, we hope to aggregate all locality measures with the aim of defining a vertex-specific signal, which can be used to categorize vertices according to their behaviour.

5 Conclusions and future work

Our experiments show that the application of scan statistics to local, vertex-specific measures is successful in identifying anomalous behaviour. Different vertex-specific measures appear to identify different types of anomalies. Moreover, the relative behaviour of scan statistics derived from different vertex measures, or from different levels of locality can indicate abnormal patterns of special interest. To interpret the anomalies identified by scan statistics, the vertices and time periods where peaks occur should be studied in detail by other methods.

The analysis of our results shows that the peaks of almost all scan statistics are achieved by vertices corresponding to mailing lists. This is no surprise, since scan statistics are especially sensitive to their bursty behaviour (long periods of inactivity followed by mass mailings). Changes in communication patterns between individual email correspondents are easily masked by the presence of such bursty nodes. In future work, we will develop methods to filter out "noisy" nodes, by characterizing nodes into groups that exhibit fairly homogeneous behaviour. Application of scan statistics to such groups should lead to the discovery of a wider variety of anomalies.

Acknowledgements

The authors gratefully acknowledge the financial support from the MITACS Network of Centres of Excellence, IT Interactive Services Inc., and the Natural Sciences and Engineering Research Council of Canada.

References

C. Cortes, D. Pregibon, and C. Volinsky. Computational Methods for Dynamic Graphs. *Journal of Computational and Graphical Statistics*, 12(4): 950–970, 2003.

J. Glaz, J. Naus, and S. Wallenstein. *Scan Statistics*. Springer-Verlag, New York, 2001.

C. Priebe, J. Conroy, D. Marchette, and Y. Park. Scan Statistics on Enron Graphs. *Computational and Mathematical Organization Theory*, 11(3): 229–247, 2005.

Reconstruction of Flexible Gene-Protein Interaction Networks using Piecewise Linear Modeling and Robust Regression

Ronald L. Westra*
*Dept. Mathematics, Maastricht University
Maastricht, The Netherlands
westra@math.unimaas.nl

Ralf L.M. Peeters†
†Dept. Mathematics, Maastricht University
Maastricht, The Netherlands
ralf.peeters@math.unimaas.nl

Goele Hollanders‡
‡Dept. Computer Science, Hasselt University
Hasselt, Belgium
goele.hollanders@uhasselt.ac.be

Karl Tuyls§
§Dept. Computer Science, Maastricht University
Maastricht, The Netherlands
karl.tuyls@cs.unimaas.nl

Abstract

In this study we will focus on piece-wise linear state space models for gene-protein interaction networks. We will follow the dynamical systems approach with special interest for partitioned state spaces. From the observation that the dynamics in natural systems tends to punctuated equilibria, we will focus on piecewise linear models and sparse and hierarchic interactions, as for instance described by Glass, Kauffman, and de Jong. Next, the paper is concerned with the identification (a.k.a. reverse engineering and reconstruction) of dynamic genetic networks from microarray data. We will describe exact and robust methods for computing the interaction matrix in the special case of piecewise linear models with sparse and hierarchic interactions from partial observations. Finally, we will analyze and evaluate this approach with regard to its performance and robustness towards intrinsic and extrinsic noise.

Keywords: piecewise linear, robust identification, hierarchical networks, gene expression data, gene regulatory networks.

1 Introduction

This paper is concerned with the identification of dynamic gene-protein interaction networks with intrinsic and extrinsic noise from empirical data, such as a set of microarray time series.

Prerequisite for the successful reconstruction of these networks is the way in which the dynamics of their interactions is modeled. The formal mathematical modeling of these interactions is an emerging field where an array of approaches are being attempted, all with their own problems and short-comings. The underlying physical and chemical processes involved are multifarious and hugely complex. This condition contrasts sharply with the modeling of inanimate Nature by physics. While in physics huge quantities of but a small amount of basic types of elementary particles interact in a uniform and deterministic way provided by the fundamental laws of nature, the situation in gene-protein interactions deals with tens of thousands of genes and possibly some million proteins. The quantities thereby involved in the actual interactions are normally very small, as one single protein may be able to (in)activate a specific gene, and thereby change the global state of the system. For this reason, gene regulatory systems are much more prone to stochastic fluctuations than the interactions involved in normal anorganic reactions. Moreover, each of these interactions is different and involves its own peculiar geometrical and electrostatic details. There are different processes involved like transcription, translation and subsequent folding. Therefore, the emergent complexity resulting from gene regulatory networks is much more difficult to comprehend.

In the past few decades a number of different formalisms for modeling the interactions amongst genes and proteins have been presented. Some authors focus on specific detailed processes such as the circadian rhythms in *Drosophila* and *Neurospora* (10), (11), or the cell cycle in *Schizosaccharomyces* (Fission yeast) (14). Others try to provide a general platform for modeling the interactions between genes and proteins. For a thorough overview consult de Jong (2002) in (2), Bower (2001) in (1), and others (6), (13).

We will focus on dynamical models, and not discuss static models where the relations between genes are considered fixed in time. In discrete event simulation models the detailed biochemical interactions are studied. Considering a large number of constituents, the approach aims to derive macroscopic quantities. More information on discrete event modeling can be found in (1).

2 Modeling gene-protein interactions as a piecewise linear system

The traditional approach to modeling the dynamical interactions amongst genes and proteins is by considering them as biochemical reactions, and thus representing them as 'rate equations'. The concept of chemical rate equations consists of a set of differential equations, expressing the time derivative of the concentration of each constituent of the reaction as some rational function of the concentrations of all the constituents involved. Though the truth of the underlying biochemical interactions between the constituents is generally accepted, a rate equation is not a fundamental law of Nature, but a statistical average over the entire ensemble of molecular collisions that contribute to an actual chemical reaction (22). So, rate equations are statistical approximations that – under certain conditions – predict the average number of reactive collisions. The actual observed number will fluctuate around this number, depending on the details of the microscopic processes involved. In case of biochemical interactions between genes and proteins the applicability of the concept of rate equations is valid only for genes with sufficient high transcription rates. This is confirmed by recent experimental findings by Swain and Elowitz (5), (16), (18), (19).

From the above, we may conclude that modeling can only be successful for genes with sufficiently high transcription rates. Even in the optimal case, we would obtain a high-dimensional (reflecting the number of genes, RNAs, and proteins involved – so tens of thousands), non-linear, differential equation, that is subject to substantial stochastic fluctuations. Much more problematic is the fact that the precise details of most reactions are unknown, and therefore cannot be modeled as rate equation. This could be compensated by a well-defined parametrized generic form of the interactions, such that the parameters could be estimated from sufficient empirical data. A generic form based on rational positive functions is proposed by J. van Schuppen (23). However, in the few cases where parts of such interaction networks have been described from experimental analysis, like the circadian rhythms in certain amoeba (10), or the cell cycle in fission yeast (14), it is clear that such forms have a too extensive syntax to be of any practical use.

Let us for the moment forsake these problems, and consider the dynamics of gene-RNA-protein networks. When we assume a stochastic differential equation as model for the dynamics of the interaction network, the relation can be expressed as:

$$\dot{x} = f(x, u|\theta) + \xi(t) \qquad (1)$$

Here $x(t)$, called the state-vector, denotes the N gene expressions and M RNA/protein densities at time t – possibly involving higher order time derivatives. $u(t)$ denotes the P controlled inputs to the system, such as the timing and concentrations of toxic agents administered to the system observed. $\xi(t)$ denotes a stochastic Gaussian white noise term. This expression involves a parameter vector θ, that contains the coupling constants between gene expressions and protein densities. We can consider this system as being represented by the state vector $x(t)$ that wanders through the (at least) $(N + M)$-dimensional space of all possible configurations. In the formalism of dynamic systems theory, eventually x will enter an area of attraction, and become subject to the influence of an attractor. An attractor here can be an uniform convergent attractor, a limit cycle, or a 'strange attractor'. We can understand the entire space as being partitioned into cells, where such attractors – or their antagonists so-called repellers – reign. Thus, the behavior of x can be described by motion through this collection of cells, swiftly moving through cells of repellers, until they enter the basin of attraction of an attractor. Under the effects of external agents via the vector $u(t)$ or by stochastic fluctuations via $\xi(t)$ they can leave this cell, and start wandering again, thereby repeating the process. Now, a vital assumption is that in each cell the behavior is governed by specific (un)stable equilibrium points, and therefore it is possible to make a linear approximation of equation 1 in the cell with index l as:

$$\dot{x}(t) = F_l x(t) + G_l u(t) \qquad (2)$$

In case of a uniform attractor the largest eigen-value of F_l will be negative, and in case of a uniform repeller the smallest eigen-value will be positive. We can now formalize the qualitative behavioral dynamics of gene-protein interactions as predominantly linear behavior near the stable equilibria – called the steady states, interrupted by abrupt transitions where the system quickly relaxes to a new steady state, either externally induced or by process noise.

In biology such behavior is frequently observed, as for instance in embryonic growth where the organism develops by transitions through a number of well-defined 'check points'. Within each such checkpoint the system is in relative equilibrium. There is an ongoing debate on mathematical modeling of cell division as *checkpoint mechanisms* versus *limit-cycle oscillators*, see (20). We will follow the view of *piecewise linear behavior* (PWL, also known more appropriately as piecewise *affine* behavior). This approach corresponds to the piecewise linear models introduced by Glass and Kauffman (9), and the qualitative piecewise linear models described by de Jong et al. (2), (3).

3 The identification of *piecewise linear networks* by L_1-minimization

Next, we will be concerned with the identification (a.k.a. *reverse engineering* or *reconstruction*) of piecewise linear gene regulatory systems from microarray data. The nature of our problem – few microarray experiments and lots of genes – implies that we are dealing with *poor data* (as opposed to *rich data*), where the number of measurements is *a priori* insufficient to identify all parameters of the system. One standard approach to circumvent this problem is by dimension reduction through the clustering of related genes. We consider the case where time series of genome-wide expression data is available. The case of the identification of a *simple* linear system is discussed in Peeters and Westra (15), (26), and Yeung et al. in (27). In the following, we will be concerned with the identification of *piecewise* linear systems. Our aim is to obtain the gene-gene interaction matrix. This matrix can be interpreted as a connectivity matrix, and so directly relates to the graph of the gene regulatory network. With this network we are able to make statements like: 'the expression of this gene causes that and that cluster of genes to alter their expression in this and this way'.

Let us in the following assume a dynamical input-output system Σ that switches irregularly between K linear time-invariant subsystems $\{\Sigma_1, \Sigma_2, \ldots, \Sigma_K\}$. Let $S = \{s_1, s_2, \ldots, s_{K-1}\}$ denote the set of – possibly unknown – switching times, i.e. the time instants $t = s_l$ that the system switches from subsystem Σ_l to Σ_{l+1}. Similarly as with the simple linear networks, we assume *Hankel matrices* $X = (x[1], x[2], \ldots, x[M])$, and $U = (u[1], u[2], \ldots, u[M])$ at M sampling times $T = \{t_1, t_2, \ldots, t_M\}$, representing full observations of the N states and P inputs. The interval between two sample instants is denoted as $\tau_k = t_{k+1} - t_k$. In first instance we assume that the system is sampled on regular time intervals, i.e. that the sample intervals are equal to τ. Within one subsystem Σ_l the relation between the inputs $u(t)$ and outputs $y(t)$ is represented as a state-space system of first-order differential (for continuous time systems) or difference equations (for discrete time systems), using an auxiliary vector $x(t)$ spanning the so-called subspace.

Continuous time:

$$\dot{x}(t) = F_l x(t) + G_l u(t), \tag{3}$$

$$y(t) = H_l x(t) + J_l u(t). \tag{4}$$

Discrete time:

$$x[k+1] = A_l x[k] + B_l u[k], \tag{5}$$

$$y[k] = C_l x[k] + D_l u[k]. \tag{6}$$

The relation between these is given by:

$$A_l = e^{\tau F_l}, \tag{7}$$

$$B_l = e^{\tau F_l} G_l. \tag{8}$$

with $x[k] = x(t_k)$.

3.1 Determination of the new state equilibrium points

Moreover, in each new state the new equilibrium point $\mu_l \in \mathbb{R}^N$ has also to be established. The linearization near μ_l can be written as:

$$\frac{\partial}{\partial t}(\mu_l + (x - \mu_l)) = F_l(x - \mu_l) + G_l u + \mathcal{O}(\|x - \mu_l\|^2) \tag{9}$$

which can be rewritten as: $\dot{x} = F_l x + \tilde{G}_l \tilde{u}$, with:

$$\tilde{G}_l = (G_l | - F_l \mu_l), \tag{10}$$

$$\tilde{u} = \begin{pmatrix} u \\ 1 \end{pmatrix}. \tag{11}$$

The reasoning is similar in the discrete case, and we obtain: $x[k+1] = A_l x[k] + \tilde{B}_l \tilde{u}[k]$. Therefore, we can follow the original formulation and, using $\tilde{u}$ rather than u as input, estimate A_l and $\tilde{B}_l$, and using:

$$\tilde{B}_l = (B_l | - A_l \mu_l)\,, \tag{12}$$

to compute μ_l and B. We will follow this approach, and from here on drop the *tilde*, and simple write B_l for $(B_l| - A_l\mu_l)$, and $u[k]$ for $\begin{pmatrix} u[k] \\ 1 \end{pmatrix}$.

3.2 General dynamics of switching subsystems

In the context of piecewise linear systems of gene regulatory systems, the dynamics is slightly different to the case of simple linear systems as in (15). In our context we assume that we observe *all* N genes, and that there is no direct through-put. This means that $C_l = I$ and $D_l = 0$ for all l. Therefore, we can suffice with equation 5 corrected for the equilibrium point:

$$x[k+1] = A_l x[k] + B_l u[k]. \tag{13}$$

We furthermore assume that the system matrices in these equations are constant during intervals $[s_l, s_{l+1} >$, and abruptly change at the transition between the intervals at $t = s_{l+1}$. We assume that on the time scale τ the system has relaxed to its new state. This means that we do not observe *mixed states*, which would severely complicate the problem of identification.

Finally, we define the *weights* w_{kl}, as the membership functions of observation k to subsystem Σ_l; if observation $\{x[k], u[k]\}$ belongs to system Σ_l then $w_{kl} = 1$, if $\{x[k], u[k]\}$ does not belong to Σ_l then $w_{kl} = 0$. This definition allows for a *fuzzy* definition of weight, such that $w_{kl} \in [0,1]$. *A priori*, we thus can state two constraints on w:

$$\forall_{k,l} w_{kl} \in [0,1], \tag{14}$$

$$\forall_l \sum_l w_{kl} = 1. \tag{15}$$

The challenge in system identification is to estimate the relevant model parameters in piecewise linear dynamics from empirical observations. The success of this approach depends on the amounts of empirical data available – *rich* or *poor*, the validity of the mathematical model, the levels of process noise and measuring noise, and the nature of the sampling process. In case of regular sampling the discrete model 5 can be applied which leads to more straightforward techniques than the continuous model 3 that should be used in case of irregular sampling. In the following sections we will study a number of these conditions in more detail.

3.3 Identification of PWL models with *unknown* switching and *regular* sampling from *poor* data

The assumption that the switching times between the linear subsystems are completely known suits various experimental conditions, as for instance when toxic agents are administered. In many biological situations, however, the exact timing between subsystems is not known, as during embryonic growth and in many metabolical processes.

3.3.1 As an extension to the simple linear systems in case state derivatives are available

When a sufficiently accurate record of estimates of the state derivatives $\dot{X} = \{\dot{x}[1], \dot{x}[2], \ldots, \dot{x}[M]\}$ is available, we can simply rewrite this problem as a special case of the method described in the case of a simple linear problem as in (15). In fact, by exploiting the data $\mathcal{D} = \{X, U, \dot{X}\}$, the problem can be stated as a linear equation in terms of new matrices H_1 and H_2 as:

$$\dot{X} = H_1 X + H_2 U. \tag{16}$$

In this equation the matrices H_1 and H_2 relate to the – unknown – system matrices $\{A_1, B_1, \ldots, A_K, B_K\}$ and ditto unknown weights $\{w_{kl}\}$ as:

$$\textsf{vec}\,(H_1) = W \cdot \textsf{vec}\,(A), \tag{17}$$

$$\textsf{vec}\,(H_2) = W \cdot \textsf{vec}\,(B). \tag{18}$$

The matrices A, B, and W are composed as follows:

$$A = \begin{pmatrix} A_1 \\ \ldots \\ A_K \end{pmatrix}, \quad B = \begin{pmatrix} B_1 \\ \ldots \\ B_K \end{pmatrix}, \tag{19}$$

$$W = w \otimes I_{N^2} = \begin{pmatrix} w_{1,1} I_{N^2} & \ldots & w_{1,K} I_{N^2} \\ \ldots & \ldots & \ldots \\ w_{M,1} I_{N^2} & \ldots & w_{M,K} I_{N^2} \end{pmatrix} \tag{20}$$

where $\otimes$ is the Kronecker-product, and I_{N^2} is the $N^2 \times N^2$ identity matrix. Note that equation 16 is not anymore a linear problem, as the unknown matrices A, B, and W appear in a non-linear way in the equation. This equation is exactly of the type of simple linear networks as in (15). Therefore, its solution method is fully applicable, so that an efficient and accurate algorithm is available for solving this problem in terms of H_1 and H_2. However, now the problem has shifted to solving two additional non-linear equations:

$$W \diamond A = H_1, \tag{21}$$

$$W \diamond B = H_2. \tag{22}$$

where A, B, and W have to be solved from the known – i.e. computed – matrices H_1 and H_2. The operation $\diamond$ makes the relations in equations 21 and 18 explicit. This is an underdetermined system that can only be solved by additional information, such as assuming sparsity for A, and a block structure for W, such as the two constraints in equations 14 and 15.

This non-linear problem can thus be solved in terms of H_1 and H_2, but not in terms of A, B, and W. It is a bilinear problem in terms of A and B for fixed W, otherwise it is a quadratic problem. As a quadratic programming problem this is not a a well-posed problem, i.e. it has a nonsingular Jacobian at optimality and is ill-conditioned as the iterates approach optimality. Therefore, we follow a different approach and split the problem in two LP-problems that are well-posed. The approach is as follows: (i) initialize A, B, and W, (ii) perform the iteration:

1. Compute H_1 and H_2, using the approach from Peeters and Westra (15) on equation 16,
2. Using fixed values for the weights W, compute A and B using equations 21, and 22,
3. Using fixed values for matrices A and B, compute the weights W using equations 14, 15, 21, and 22,

until: (iii) a cumulative weighted error criterion $\mathcal{E}$ has converged sufficiently – or a maximum number of iterations has passed. A proper choice for the criterion function is:

$$\mathcal{E}(A,B,W|\mathcal{D}) = \sum_{k,l} w_{kl} \|A_l x[k] + B_l u[k] - \dot{x}[k]\|_2^2 \tag{23}$$

This problem can be solved by minimizing the quadratic L_2-criterion subject to mentioned constraints, for instance by a gradient descent method. We can, however, formulate a different approach for solving this problem by defining an alternative criterion function $\mathcal{E}$, namely as a linear L_1-criterion:

$$\mathcal{E}_1(A,B,W|\mathcal{D}) = \sum_{k,l} w_{kl} \|A_l x[k] + B_l u[k] - \dot{x}[k]\|_1 \tag{24}$$

This expression allows for an LP-formulation of the problem, in which $\mathcal{E}_1$ serves as the objective function. Thus, we can split the non-linear optimization problem as two separate LP-formulations that are successively applied in the iteration; (i) an LP-problem LP_1 for obtaining the system matrices A and B from minimizing objective function $\mathcal{E}_1$ with given weights w, subject to the constraints in equations 21 and 22; and (ii) an LP-problem LP_2 for obtaining the weights w from minimizing objective function $\mathcal{E}_1$ with given system matrices A and B , subject to the constraints in equations 14, 15, 21, and 22.

We will revisit this philosophy in the next Section, when reviewing the more realistic case when the state derivatives of the gene expressions are *not* available.

4 Numerical experiments and performance of the approach.

This approach resulted in an efficient and fast algorithm that is able to accurately estimate the gene-gene coupling matrix for tens of thousands of genes based on only several hundred genome wide measurements, and that is robust towards measurement noise. With increasing measurement noise or decreasing number of measurements the approach retains the strongest gene-gene coupling links - i.e. the largest modal value of the coupling matrix A - longest, see Figure 1. A basic assumption in the approach is the sparsity of the underlying gene-gene coupling matrix, represented by the number of non-zero entries per row. If this number grows above a certain threshold the performance of the approach is severely affected, see Figure 2*b*. A number of numerical experiments were performed with this approach. These controlled experiments consist of the comparison of reconstructed network with the - known - original network structure. They were all performed on a PC with an PIV dual XEON processor of 3.2 GHz and 4096 MB RAM memory under Linux fedora core 3, using Matlab 6.5 release 13 including the optimization toolbox. The Matlab routine `linprog` was used to solve LP problems; its default solution method is a primal-dual interior point method, but an active set method can optionally be used too. *For larger problems it turned out to be essential for obtaining reasonable computation times, that the LP problems were solved by application of the active set method on the dual problem formulation. Therefore this method was adopted throughout all the experiments.* In line with the definitions above, we use the parameters N, M, K to quantify the size and complexity of the input. In addition, the sparsity of the interaction matrix A is measured by the number of nonzero entries per row and denoted by k (which should be much less than N). To quantify the quality of the resulting approximation A_{est} of A^* two performance measures are introduced: the number of errors N_e and the CPU-time T_c as clocked on the same platform.

1. *The number of errors* N_e.
   Errors in the reconstruction are generated by the

failure of the algorithm to identify the true nonzero elements of the original sparse vector x_0. These errors stem from false positives and false negatives in the reconstructed vector x_d. Their numbers are added up to produce the total number of errors N_e.

2. *The CPU-time T_c.*
   Using internal clocking, the time T_c required to perform the full computation was measured. As all numerical experiments are executed on the same platform under similar conditions, this provides a measure to compare problem instances.

The numerical experiments clearly demonstrate the range where the approach is effective. For relatively moderate noise levels and a high degree of sparsity i.e., a small number k of nonzero elements in the rows of matrix A - and not too many external stimuli p and switching times K, the approach allows one to reconstruct a sparse matrix with great accuracy from a relative small number of observations $M \ll N$. For example, a row of A with 30,000 components of which all but 10 are equal to zero, can be efficiently reconstructed from just 150 independent measurements, see Figure 4a. The sparsity property of A fits in nicely with the technique of L_1-minimization, which automatically will always set many entries of the solution A^* to zero, whereas L_2-regression would spread out the error over all components, thus creating many small components. Reconstruction of large networks from this approach is straightforward: each of the rows of the gene-gene interaction matrix can be computed independently from the same set of microarray experiments.

Acknowledgements

The authors express their gratitude to Gert-Jan Bex and Marc Gyssens from the University of Hasselt for their valuable contribution.

References

[1] Bower J.M., Bolouri H.(Editors), Computational Modeling of Genetic and Biochemical Networks, *MIT Press*, 2001. bibitemDavidson1999Davidson E.H. (1999), A View from the Genome: Spatial Control of Transcription in Sea Urchin Development, *Current Opinions in Genetics and Development*, **9**, pp. 530 – 541.

[2] de Jong H., Modeling and Simulation of Genetic RegulatorySystems: A Literature Review, Journal of Computational Biology, 2002, Volume 9, Number 1, pp. 67–103

[3] de Jong H., Gouze J.L., Hernandez C., Page M., Sari T., Geiselmann J., Qualitative simulation of genetic regulatory networks usingpiecewise-linear models, Bull Math Biol. 2004 Mar;66(2): pp 301–40.

[4] D'haeseleer P., Liang S., Somogyi R., Genetic Network Inference: From Co-Expression Clustering to Reverse Engineering, *Bioinformatics*, vol. **16**, no. 8, 2000, pp. 707–726.

[5] Elowitz M.B., Levine A.J., Siggia E.D., Swain P.S., Stochastic gene expression in a single cell, *Science*, vol.**297**, August 16, 2002, pp.1183–1186.

[6] Endy, D, Brent, R. (2001) Modeling Cellular Behavior, *Nature* 2001 Jan 18; 409(6818):391-5.

[7] Fuchs J.J. (2003), More on sparse representations in arbitrary bases, in: Proc. 13th IFAC Symp. on System Identification, Sysid 2003, Rotterdam, The Netherlands, August 27-29, 2003, pp. 1357–1362.

[8] Fuchs J.J. (2004), On sparse representations in arbitrary redundant bases, IEEE Trans. on IT, June 2004.

[9] Glass L., Kauffman S.A. (1973), The Logical Analysis of Continuous Non-linear Biochemical Control Networks, *J.Theor.Biol.*, 1973 Vol. 39(1), pp. 103–129

[10] Goldbeter A (2002) Computational approaches to cellular rhythms. Nature 420, 238-45

[11] Gonze D, Halloy J, and Goldbeter A (2004) Stochastic models for circadian oscillations : Emergence of a biological rhythm. *Int J Quantum Chem* **98**, pp 228–238.

[12] Guthke R., Möller U., Hoffmann M., Thies F., Töpfer S., 2004, Dynamic network reconstruction from gene expression data applied to immune response, *Bioinformatics*, 2004, pp 2261

[13] Hasty J., McMillen D., Isaacs F., Collins J. J., (2001), Computational studies of gene regulatory networks: in numero molecular biology,*Nature Reviews Genetics*, vol. 2, no. 4, pp. 268– 279, 2001.

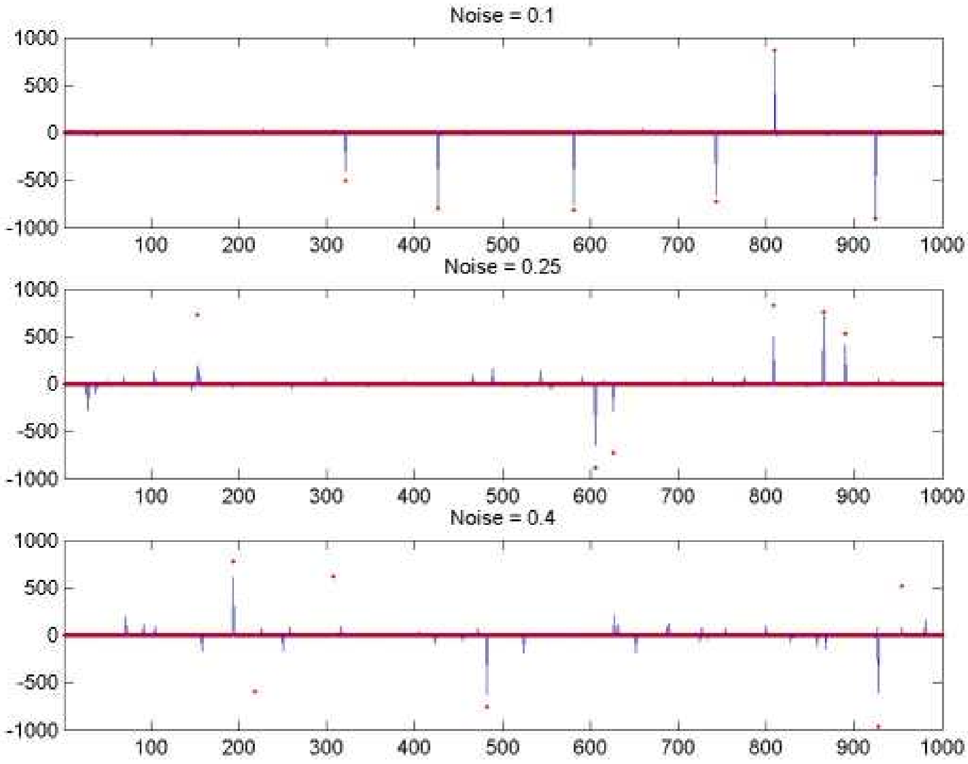


Figure 1: The influence of increasing intrinsic noise on the identifiability. The plot shows the corresponding values of the gene-gene matrix $a \equiv vec(A)$, and increasing zero-mean Gaussian noise added to A. The red dots indicate the true value of a, and the blue line the reconstructed values a^*. For low noise levels, like 0.1, the non-zero values of a are recovered without exception. At noise level 0.4 only the largest modulus maxima values have a chance to be found.

[14] Novak B, Tyson JJ (1997) Modeling the control of DNA replication in fission yeast, PNAS, USA, Vol. 94, pp. 9147-9152, August 1997.

[15] Peeters R.L.M., Westra R.L., On the identification of sparse gene regulatory networks, *Proc. of the 16th Intern. Symp. on Mathematical Theory of Networks and Systems* (MTNS2004) Leuven, Belgium July 5-9, 2004

[16] Rosenfeld N, Young JW, Alon U, Swain PS, Elowitz MB, Gene regulation at the single-cell level, *Science* 307 (2005) pp 1962.

[17] Somogyi R., Fuhrman S., Askenazi M., Wuensche A. (1997). The Gene Expression Matrix: Towards the Extraction of Genetic Network Architectures. Nonlinear Analysis, *Proc. of Second World Cong. of Nonlinear Analysis* (WCNA96) 30(3) pp 1815–1824.

[18] Swain P.S., Efficient attenuation of stochasticity in gene expression through post-transcriptional control, J Mol Biol 344 (2004) pp 965.

[19] Swain P.S., Elowitz MB, Siggia ED, Intrinsic and extrinsic contributions to stochasticity in gene expression, *PNAS* 99 (2002) pp 12795.

[20] Steuer R. (2004), Effects of stochasticity in models of the cell cycle:from quantized cycle times to noise-induced oscillations, Journal of Theoretical Biology 228 (2004) 293-301.

[21] Tegnér J., Yeung M.K.S., Hasty J., Collins J.J., Reverse engineering gene networks: Integrating genetic perturbations with dynamical modeling, *Proc. Nat. Acad. Science*, vol. **100**, no. 10, 2003, pp. 5944–5949.

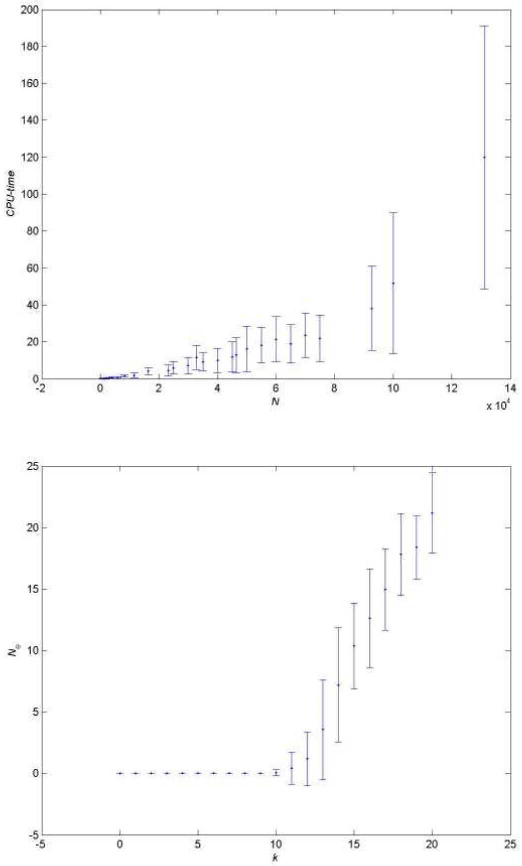


Figure 2: *a*: CPU-time T_c as a function of the problem size N, *b*: Number of errors as a function of the number of nonzero entries k in x_0, for $M = 150$, $m = 5$, $N = 50000$.

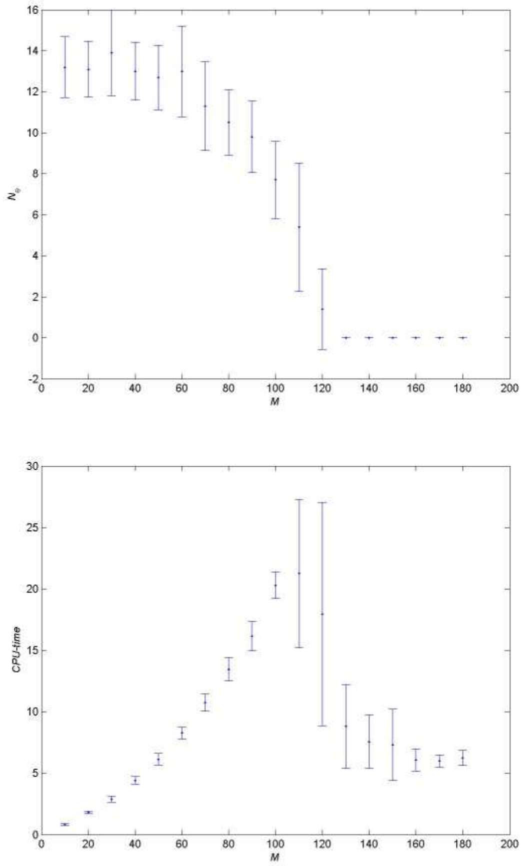


Figure 3: *a*: Number of errors as a function of M for $N = 50000$, $k = 10$, $m = 0$, *b*: Computation time as a function of M, for $N = 50000$, $k = 10$, $m = 0$.

[22] van Kampen N. G. (1992), Stochastic Processes in Physics and Chemistry, Elsevier ScienceB. V., Amsterdam, (1992).

[23] van Schuppen J.H. (2004), System theory of rational positive systems for cell reaction networks, CWI Report MAS-E0421, December 2004, ISSN 1386-3703

[24] Verdult V., Verhaegen M., Subspace Identification of Piecewise Linear Systems, In *Proc. 43rd IEEE Conference on Decision and Control (CDC)*, pp 3838–3843, Atlantis, Paradise Island, Bahamas, December 2004.

[25] Westra R.L., Peeters R.L.M. (2004), Modelling and identification of dynamical gene interactions: presentation, Workshop Intelligent Technologies for Gene Expression Based Individualized Medicine, 14th May 2004, Jena/Germany

[26] Westra R.L.,(2005*a*), Piecewise Linear Dynamic Modeling and Identification of Gene-Protein Interaction Networks, Nisis/JCB Workshop reverse engineering, Jena, June 10, 2005.

[27] Yeung M.K.S., Tegnér J., Collins J.J., Reverse engineering gene networks using singular value decomposition and robust regression, *Proc. Nat. Acad. Science*, vol. **99**, no. 9, 2002, pp. 6163–6168.

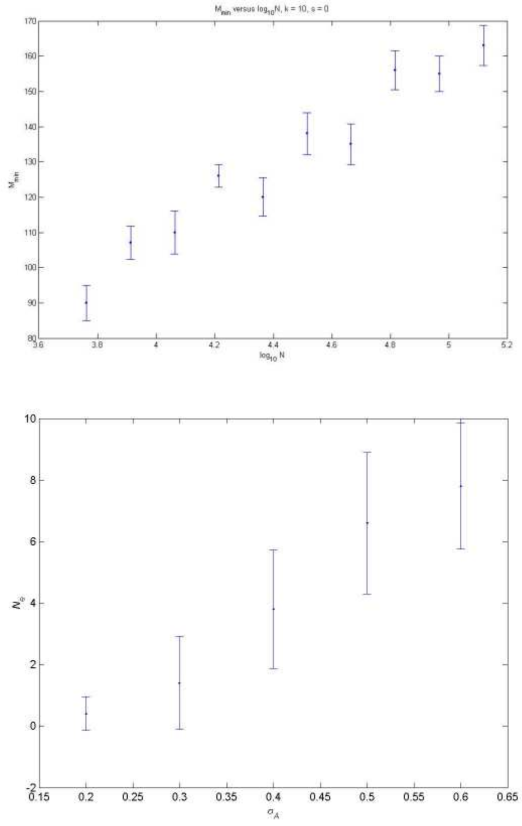


Figure 4: *a*: Dependency of the critical value M_{min} required to compute the matrix free of error versus the problem size N, *b*: Number of errors as a function of the intrinsic noise level σ_A, for $N = 10000$, $k = 10$, $m = 5$, with $M = 150$ and measuring noise $\sigma_B = 0$.

General Classification of Networks

Thomas Wilhelm*
*Theoretical Systems Biology
Leibniz Institute for Age Research - Fritz Lipmann Institute
Beutenbergstr.11
D-07745 Jena, Germany
`wilhelm@fli-leibniz.de`

Jens Hollunder*
`hollund@fli-leibniz.de`

Jong-Kwang Kim*
`gensdei@fli-leibniz.de`

Andreas Beyer*
`beyer@fli-leibniz.de`

Abstract

We present a new general classification scheme of networks, which is appropriate for any type: weighted or unweighted, and directed or undirected networks. We show that all networks can be grouped into one of two general classes: democracy or dictatorship networks. In democracy networks nodes tend to play equal roles in the network, whereas in dictatorship networks some nodes are more prominent for the network function. In other words, in democracy networks there is more cycling of information (or mass, or energy), while dictatorship networks are characterized by a straight through-flow from sources to sinks. The classification is based on information theoretic measures. If the *redundancy* of a given network is smaller than in a randomized version ($R < R_r$), we call it democracy network. In dictatorship networks it is $R > R_r$. We use the measure *Medium Articulation* to quantify the complexity of networks. Complex networks ($MA > MA_r$) are always between the pure democracy and the pure dictatorship networks. Taken together, we distinguish four different network types: pure democracy networks ($R < R_r, MA < MA_r$), complex democracy networks ($R < R_r, MA > MA_r$), complex dictatorship networks ($R > R_r, MA > MA_r$), and pure dictatorship networks ($R > R_r, MA < MA_r$). Finally, different real networks (weighted and unweighted, directed and undirected) are classified according to our proposed scheme.

1 Introduction

Starting with two pioneering works (2; 18), the last years have seen a surge of papers dealing with networks on fields as diverse as social networks (9; 10; 19), food webs (6; 21; 22), communication networks (4; 23), transportation networks (7), and sub-cellular networks, such as metabolic (13), protein interaction (5; 24), and genetic networks (14). Many interesting network properties have been found, providing us insights into both, the dynamics on networks (17) and (on another time scale) the evolutionary processes leading to such networks (1; 2). However, nearly all of these papers deal with simple unweighted networks, where links are either present or absent. Here we go one step further and present a characterization of the most general form of networks: directed weighted (non-binary) networks. Here n nodes are connected by maximally n^2 directed weighted edges (links) t_{ij} from node i to node j. Note that both, undirected and unweighted networks can also be analysed in this framework: in unweighted networks the edges have only two different weights (0-no edge, 1-edge), and undirected networks can be understood as directed with all links pointing in both directions (with the appropriate weight).

Note that many networks could better be described by their weighted form. For instance, in acquaintance networks one can quantify the acquaintances, or in coauthorship networks one can count the number of jointly written papers. Especially for food webs a weighted network description seems to be inevitable. Very small fluxes of mass or energy between nodes (usually trophic species) are probably not observed in reality, or may simply be neglected if deemed irrelevant. However, the decision whether to take small fluxes into account affects the statistics of the corresponding unweigted network (22). Up to now there is much ambiguousness in food web theory. More than ten years ago the leading "students of food webs" urgently demanded a common food web standard (3).

However, up to now such a standard has not been established, but it seems to be generally accepted that food webs should contain weighted links (3; 22).

It was shown that weighted networks are substantially more informative than unweighted networks (21; 22). The latter can simply be deduced from the former (21; 22): a special *cut*-value is defined, if the weight is larger *cut* the link is set, if not it is neglected. We have shown that such a deduced unweighted network may not be distinguishable from a random (Erdős-Renyi-) network, although the original weighted network was clearly structured (21). A simple *Gedankenexperiment* shows the potential ambiguity of the characterization of weighted networks with measures developed to describe unweighted networks: the highly weighted links can show a scale-free property (2) (power-law distribution of node degrees, taking into account only strong links), but a different degree distribution may be obtained if also links with small weights are considered. This problem is of importance especially for food webs where no standard exists: it depends on the personal choice of the ecologist if weak links are also counted. For instance, lions are usually considered as top predators, but gnats also bit them, so there is a small material flux also from lions to gnats which is usually not considered in corresponding food webs.

In either case, it is important to develop a deeper understanding of weighted networks. Appropriate statistical measures to characterise such networks are therefore needed. Theoretical ecologists developed different information theoretic measures for the description of directed and weighted food webs (11; 12; 16; 20; 21; 22), which can serve as a starting point for a corresponding general network theory. Recently, we proposed *Medium Articulation* as the first complexity measure for networks (20; 21).

In the first part of this paper we shortly review different appropriate measures to characterize weighted and unweighted, directed and undirected networks in a unifying manner. In the second part we show that an analysis of these measures allows to classify a given network into one of the four classes: pure democracy networks, complex democracy networks, complex dictatorship networks, and pure dictatorship networks. In the last part we discuss the corresponding class-membership of different real networks.

2 Information theoretic measures for the characterisation of networks

In the following T_{ij} exclusively denotes the normalized link from i to j: $T_{ij} = t_{ij}/T$ with the total sum of links $T = \sum_i \sum_j t_{ij}$ (t_{ij} is the non-normalized value). The most important measures we need are the *joint entropy* H, the *redundancy* of the network R and the *mutual information* I which are defined as follows (cf.(11)):

$$H = -\sum_i \sum_j T_{ij} \log T_{ij}, \tag{1}$$

$$R = -\sum_i \sum_j T_{ij} \log \frac{T_{ij}^2}{\sum_k T_{kj} \sum_k T_{ik}}, \tag{2}$$

$$I = H - R = \sum_i \sum_j T_{ij} \log \frac{T_{ij}}{\sum_k T_{kj} \sum_k T_{ik}} \tag{3}$$

From (3) one directly recognizes $I = I_{min} = 0$ if $T_{ij} = \sum_k T_{kj} \sum_k T_{ik}\ \forall t_{i,j} \neq 0$. Such a network is shown in Fig.1a. Furthermore: $H = H_{max} = 2\log n$ if $T_{ij} = 1/n^2\ \forall i,j$ (Fig.1a); $H = H_{min} = 0$ if $T_{ij} = 1$ for any i, j and the remaining links equal zero (eq. (1)); $R = R_{max} = 2\log n$ if $T_{ij} = 1/n^2\ \forall i,j$ (Fig.1a) and $R = R_{min} = 0$ if $T_{ij}^2 = \sum_k T_{kj} \sum_k T_{ik}\ \forall T_{ij} \neq 0$ (eq. (2)) (Fig.1c). $I = I_{max} = \log n$ if H is as large as possible ($H_{I_{max}} = \log n$) under the condition $R = R_{min} = 0$ (Fig.1c). Note that all networks in the extreme cases $H = H_{max} = R_{max}$ and $I = I_{max}$ belong to the class of Kirchhoff-networks where $\sum_i T_{ij} = \sum_i T_{ji}$ $\forall j$. Summarizing, highly connected networks are characterized by high H and high R, but low I-values. Sparsely connected networks, i.e. highly “articulated” ones, have low H- and R-values, but a high mutual information I.

Recently, we introduced *Medium Articulation* as the first measure for the complexity of networks(20):

$$MA = I \cdot R. \tag{4}$$

MA is a typical complexity measure in the sense that it is zero in the extreme cases (here: if either $I = 0$ or $R = 0$), but maximum in between (8; 15). Thus, $MA = 0$ for the networks given in Fig.1a,c. We have shown previously (20) that $MA = MA_{max} = (\log n)^2/2$ for the network in Fig.1b (link weights should all be equal in the extreme cases).

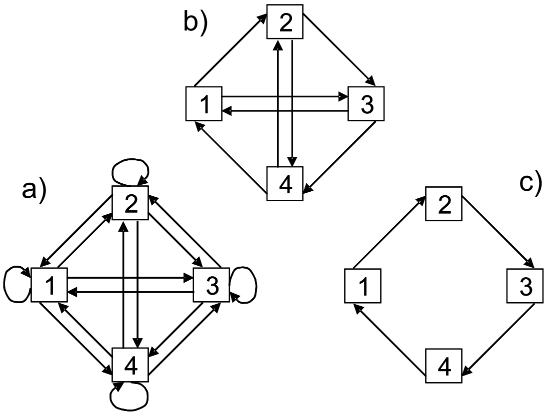


Figure 1: Three different 4-node-networks. a) maximally connected, $H = H_{max}$, $R = R_{max}$, $I = 0$, $MA = 0$; b) moderately connected, i.e. moderately articulated, $R = R_{max}/2$, $I = I_{max}/2$, $MA = MA_{max}$; and c) minimally connected, i.e. maximally articulated, $R = 0$, $I = I_{max}$, $MA = 0$.

3 Complex and non-complex democracy and dictatorship networks

Here we show that the measures described above can be used to classify all networks into one of four different classes. For that purpose, the redundancy R and the medium articulation MA of a given network are compared with the mean R_r and MA_r of correspondingly randomized networks (edges are randomly rewired). Note that the joint entropy H of a given network does not depend on the network's topology, but only on the number and weights of the edges. It follows $H = H_r = R + I = R_r + I_r$. If $R < R_r$ it follows $I > I_r$ and vice versa, thus I does not carry any additional information. Fig.1 shows that $R = 0$ for minimally connected networks with a ring structure. A corresponding analysis shows that minimally connected networks can also have a vanishing mutual information (i.e. maximum redundancy for the given edges), namely star-shaped networks: $I = 0$ if all links are going out from one single node or if all links are pointing to it. Obviously, ring-networks with $R = 0$ have a lower redundancy than their random counterparts $R < R_r$. Because in such networks the nodes play equal roles we call them *democracy networks*. Dictatorship networks, in contrast, have $R > R_r$ (i.e. $I < I_r$). The network complexity measure medium articulation serves to subdivide the two major groups: complex networks with $MA > MA_r$ lie between pure democracy and pure dictatorship networks. Summarizing, pure democracy networks have $R < R_r, MA < MA_r$, complex democracy networks $R < R_r, MA > MA_r$, complex dictatorship networks $R > R_r, MA > MA_r$, and pure dictatorship networks $R > R_r, MA < MA_r$.

In the following part of this section we exclusively deal with directed unweighted networks, i.e. $T_{ij} = 1/L\,(\forall\, T_{ij} > 0)$, where L denotes the number of links (directed edges). Each directed unweighted network with L links has a joint entropy $H(n, L) = log(L)$, for any number of nodes.

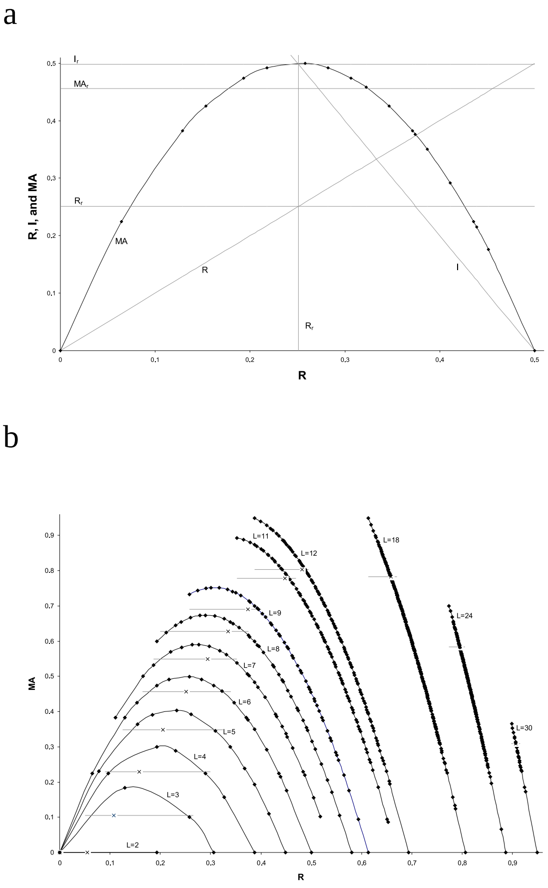


Figure 2: Classification of all directed unweighted 6-node-networks (normalized R, R_r, I, I_r, MA, MA_r). All networks above the horizontal MA_r line are complex networks (below are non-complex (pure) networks), all networks left of R_r are democracy networks (right are dictatorship networks). a) all networks with $L = 6$ edges b) all networks with $L = 2, 3, \ldots$ edges. x indicates the corresponding exact arithmetic mean values (R_r, MA_r) of the randomized networks.

Fig.2a shows the redundancy R, mutual information I, and medium articulation MA for all networks with $n = 6$ nodes and $L = 6$ directed unweighted edges. It can be seen that all four different network types can be found within the class of directed $n = 6, L = 6$ networks. Fig.2b shows for some selected L the corresponding R and MA for all $n = 6$ networks, as well as the corresponding random network values R_r and MA_r. It can be seen that complex dictatorship networks only exist for small L. Analysis shows that for a given n there are many more complex democracy, than complex dictatorship networks. Pure democracy networks only exist for even smaller L. In other words, most democracy networks are complex, whereas most dictatorship networks are non-complex.

Our classification scheme also allows to extract a special information about a given network: if $R < R_r$ (democracy), the information (or mass, or energy) tends to cycle in the network, if $R > R_r$ (dictatorship) there is a tendency for straight information through-flow from sources to sinks. Thus, democracy networks are cycling networks, and dictatorship networks could also be named source-sink networks.

4 Classification of real networks

In contrast to the well-known network classifications "small-world"(18) and "scale-free"(2) our classification scheme is of maximum generality. It is applicable to all four network types: directed and undirected, and weighted and unweighted networks. Table 1 shows the classification for some real networks of each of these types. The analysed food webs are always dictatorship networks. This seems plausible, because of the underlying trophic hierarchy. It is well-known that predators are mostly controlling different prey one trophic level below themselves. Four of the five directed unweighted food webs are non-complex, whereas 11 of 12 directed weighted food webs are complex. For a first corresponding comparison we have taken the largest weighted network ($n = 66$) as unweighted (i.e. all fluxes above the $cut = 0$ are 1 ($T_{ij} = 1/66$), the others are 0) and obtained also a pure dictatorship network ($R = 0.68 > R_r = 0.60, MA = 0.65 < MA_r = 0.94$). With other cut-values we again obtained complex dictatorship networks ($cut = 1/10000, 1/100, 1/10 \cdot t_{ij,max}$). In future we will study the dependency of the classification on cut-values more in detail.

In the analysed two neural networks the nodes are neurons and the weights correspond to the synaptic strength between the neurons (male adult worm (jsh), hermaphrodite worm (n2u)). Interestingly, both networks are pure dictatorship networks, that means there is a tendency to straight information through-flow from sources to sinks. This feature is even more pronounced in the neural network of the male adult worm.

In the undirected weighted railway network of the German federal state Brandenburg nodes are stations and weights correspond to spatial distances. It is a pure democracy network. The cycling property can easily be understood, because the whole railway network has the form of a cycle: it is circled around Berlin (Fig.3). It will be interesting to compare this result to other transportation networks.

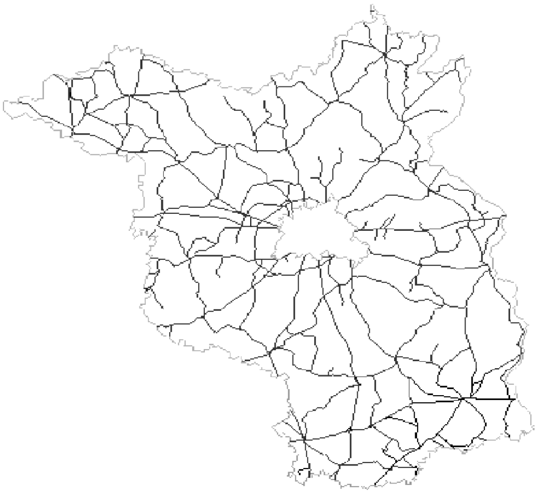


Figure 3: The Brandenburg railway network.

The analysed protein-protein interaction network is of the pure democracy type, which indicates that, on average, proteins play similar roles in the corresponding networks and the information is cycling. It seems possible to extract biologically important information from a classification analysis of different protein networks. A lower complexity could, for instance, indicate perturbation or disease, but future studies are needed.

5 Conclusion

Using information theoretic measures to characterize networks, we have introduced the four network classes pure democracy networks, complex democracy networks, complex dictatorship networks, and pure dictatorship networks. This general classification scheme holds for all types of networks, weighted and unweighted, as well as directed and undirected

Table 1: Classification of real networks. Data from: the directed unweighted networks (www.cosin.org/extra/data/foodwebs/web.html), the directed weighted food webs (www.cbl.umces.edu/ ulan/ntwk/network.html), the directed weighted neural networks (elegans.swmed.edu/parts/neurodata.txt), the Brandenburg railway network (www.bahnstrecken.de/strecken.htm), and the *E.coli* protein-protein interaction network (www.cosin.org/extra/data/proteins).

Networks	n	L	R	R_r	MA	MA_r
directed and unweighted (food webs):						
Grassland	88	137	0.232	0.18	0.589	0.532
Little Rock Lake	183	2494	0.613	0.508	0.675	0.986
Silwood Park	154	370	0.396	0.219	0.605	0.645
St. Martin Island	45	224	0.524	0.446	0.784	0.945
Ythan Estuary	135	601	0.452	0.328	0.724	0.851
directed and weighted: (food webs)						
fw1	21	82	0.11	0.037	0.09	0.044
fw2	21	61	0.124	0.036	0.097	0.047
fw3	36	122	0.186	0.104	0.319	0.243
fw4	36	172	0.295	0.084	0.121	0.173
fw5	21	55	0.303	0.249	0.598	0.595
fw6	66	791	0.159	0.022	0.132	0.041
fw7	43	348	0.233	0.057	0.107	0.104
fw8	32	158	0.263	0.099	0.247	0.22
fw9	51	270	0.269	0.133	0.365	0.323
fw10	34	158	0.259	0.164	0.451	0.407
fw11	34	149	0.245	0.139	0.385	0.333
fw12	34	115	0.284	0.177	0.462	0.438
(neural networks)						
C. elegans (jsh)	190	4336	0.451	0.399	0.868	0.933
C. elegans (n2u)	202	3963	0.446	0.403	0.883	0.936
undirected and weighted:						
Railway network	213	332	0.108	0.115	0.339	0.352
undirected and unweighted:						
E. coli prot-prot interaction	270	1432	0.403	0.43	0.793	0.966

networks. Our first analyses show that special types of real networks belong to special classes. Unweighted food webs, for instance, are pure dictatorship networks, whereas the analysed weighted food webs are complex dictatorship networks.

R. Albert, A.-L. Barabasi, Statistical mechanics of complex networks. Rev. Mod. Phys. 74,47-97, 2002.

A.-L. Barabasi, R. Albert, Emergence of scaling in random networks. Science 286,509-512, 1999.

J.E. Cohen, et al., Improving food webs. Ecology 74, 252-258, 1993.

R. Cohen, K. Erez, D. Ben-Avraham, S. Havlin, Resilience of the internet to random breakdowns Phys.Rev.Lett. 85,4626-4628, 2000.

H. Jeong, S.P. Mason, A.-L. Barabasi, Z.N. Oltvai, Lethality and centrality in protein networks. Nature 411,41-42, 2001.

A.E. Krause, K.A. Frank, D.M. Mason, R.E. Ulanowicz, W.W. Taylor, Compartments revealed in food-web structure. Nature 426,282-285, 2003.

V. Latora, M. Marchiori, Is the Boston subway a small-world network? Physica A 314,109-113, 2002.

R. Lopez-Ruiz, H.L. Mancini, X. Calbet, A statistical measure of complexity. Phys.Lett.A 209,321-326, 1995.

M.E.J. Newman, The structure of scientific collaboration networks. Proc.Natl.Acad.Sci.USA 98,404-409, 2001.

M.E.J. Newman, D.J. Watts, S.H. Strogatz, Random graph models of social networks. Proc.Natl.Acad.Sci.USA 99,2566-2572, 2002.

C. Pahl-Wostl, The Dynamic Nature of Ecosystems. Wiley, New York, 1995.

R.W. Rutledge, B.L. Basore, R.J. Mulholland, Ecological stability: an information theory viewpoint. J.Theor.Biol. 57,355-371, 1976.

S. Schuster, D.A. Fell, T. Dandekar, A general definition of metabolic pathways useful for systematic organization and analysis of complex metabolic networks. Nature Biotech. 18,326-332, 2000.

S. Shen-Orr, R. Milo, S. Mangan, U. Alon, Network motifs in the transkriptional regulation network of E. coli. Nature Genetics 31,64-68, 2002.

J.S. Shiner, M. Davison, P.T. Landsberg, Simple measure for complexity. Phys.Rev.E 59,1459-1464, 1999.

R.E. Ulanowicz, Ecology, the Ascendent Perspective. Columbia Univ. Press, 1997.

D.J. Watts, Small Worlds. Princeton Univ. Press, Princeton, NJ, 1999.

D.J. Watts, S.H. Strogatz, Collective dynamics of ‘small-world’ networks. Nature 393,440-442, 1998.

D.J. Watts, P.S. Dodds, M.E.J. Newman, Identity and Search in Social Networks. Science 296,1302-1305, 2002.

T. Wilhelm, R. Brüggemann, Information theoretic measures for the maturity of ecosystems. in: Integrative Systems Approaches to Natural and Social Sciences - Systems Science 2000, M. Matthies, H. Malchow and J. Kriz (eds.) (Springer, Berlin), 2001, pp. 263-273.

T. Wilhelm, An elementary dynamic model for non-binary food-webs. Ecol.Model. 168,145-152, 2003.

T. Wilhelm, A study of equal node models for food webs. Ecol.Model. 181,215-227, 2005.

T. Wilhelm, P. Hänggi, Power-law distributions resulting from finite resources. Physica A, 329(3-4),499-508, 2003.

T. Wilhelm, H.-P. Nasheuer, S. Huang, Physical and functional modularity of the protein network in yeast. Mol.Cell.Prot. 2.5,292-298, 2003.

Network Analyses to Understand the Structure of Wikipedia

Takeshi Yamada[⋆] Kazumi Saito[⋆] Kazuhiro Kazama
[⋆]NTT Communication Science Laboratories
Kyoto, JAPAN
yamada,saito @cslab.kecl.ntt.co.jp
NTT Network Innovation Laboratories
Tokyo, JAPAN
kazama@ingrid.org

Abstract

We investigated the structure of Wikipedia, the well known Web-based encyclopedia, as a very large-scale hyperlink network by using network analysis methods. We analyzed the time evolution of the number of articles and links, applied core-extraction methods to identify highly interconnected subnetworks, and compared different centrality measures to understand its characteristics.

1 Introduction

According to its website[1], Wikipedia is a multilingual Web-based, free-content encyclopedia. It is written collaboratively by volunteers, allowing articles to be changed by anyone with access to a Web browser. The project began on January 15, 2001 as a complement to the expert-written Nupedia, and is now operated by the non-profit Wikimedia Foundation. The English-language version of Wikipedia has more than 880,000 articles as of November 2005.

It is an interesting challenge to apply network analyses to such a large-scale network like Wikipedia to unveil its basic structure and extract useful information. There are a number of reasons why the Wikipedia network is so attractive. First, the size of the network is enormous: there are more than 880,000 articles and more than 27,000,000 links as of November 2005. Sophisticated but time-consuming methods, such as clique-based community extraction approaches, are not applicable: even calculating the graph diameter of the entire network takes a lot of time. We, therefore, need to invent fast and efficient algorithms for existing methods or develop alternative approximation methods. Second, it is a rapidly growing network. The revision records show that there were only about 1,000 articles and 9,000 links in April 2001. We can study growth patterns by using the network as a typical example of asynchronously updated and collaboratively built network. Third, it is a relatively well-organized and comprehensible network. The semantic information inherent in the Wikipedia network as an encyclopedia, such as article titles, article contents, and category hierarchies can be used to verify and evaluate the correctness and usefulness of a method for network analysis.

This paper is organized as follows. In Section 2, we analyze the growth pattern of the Wikipedia network, and show that it follows the densification power law. In Section 3, we apply community extraction methods to an undirected network generated from Wikipedia and investigate its community structure. In Section 4, we calculate and compare centrality measures for the directed Wikipedia category network and discuss their different features. Section 5 concludes the paper.

2 Time evolution

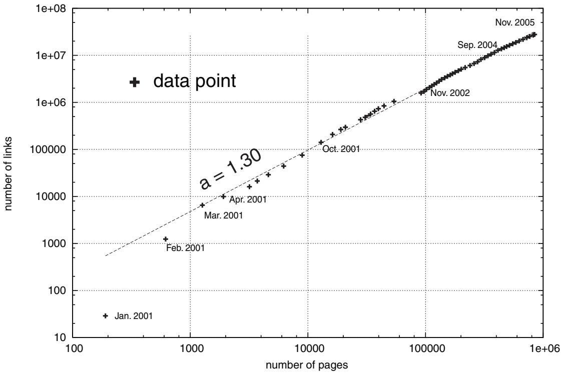


Figure 1: Number of articles versus number of links

The articles in Wikipedia are divided into several namespaces including the main namespace, consisting of all regular articles, and the category namespace, consisting of articles that index regular articles. We first focus on the main namespace. Each Wikipedia article has number of references to other articles, expressed as hyperlinks. We treat a reference

---

[1] http://en.wikipedia.org/wiki/Wikipedia

from an article to another as a link.

Figure 1 shows the time evolution of the number of articles (x-axis) and the number of links (y-axis) on a logarithmic scale. Each point in the graph corresponds to a monthly snapshot of the Wikipedia network in the period from January 2001 to November 2005. This graph illustrates the power law relationship ($R^2{=}0\ 98$), and is called a densification power law plot (Leskovec et al., 2005). The fact that the slope of the plot is $a{=}1\ 30 > 1$ indicates that the network is becoming denser over time, with the number of links growing super-linearly with the number of articles. The value is lower than that of the citation network reported in (Leskovec et al., 2005), but still clearly exhibits super-linear growth. It is clear, then, how rapidly the Wikipedia network has been growing and densifiying the relationship between articles in a scale-free fashion.

3 Community extractions

In this section, we focus on an undirected network by considering that two articles are linked if both articles refer to each other, and taking the maximal connected component, resulting in an undirected network with 536,724 nodes and 1,337,902 links This undirected network, as well as the directed one discussed in the next section, is obtained from the Wikipedia snapshot created on 24th September 2005.

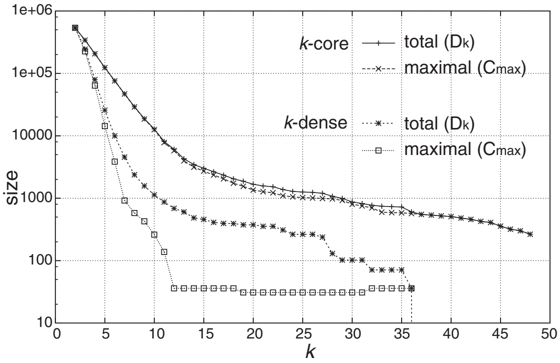


Figure 2: Maximal and total community sizes for k-core and k-dens methods for each k

To understand the structural and functional properties of a large-scale network, it is crucial to identify subnetworks (communities) in which the nodes are more highly interconnected than to the rest of the network. There are several such community extraction methods published in the literature. For example, Palla et al. (2005) proposed a method called "Clique Finder" based on the notion of k-clique (complete subgraphs of size k) [2]. They define a k-clique community as a union of all k-cliques that can be reached from each other through a series of adjacent k-clique (where two k-cliques are called adjacent when they share $k-1$ nodes). Unfortunately, their method is not suitable for a large-scale network, because finding all k-cliques is NP-hard and algorithmically intractable.

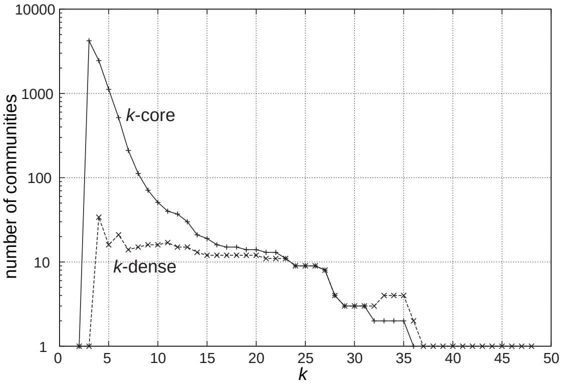


Figure 3: Number of communities obtained by the k-core and k-dens method for each k

Another well known method is called k-core community extraction or k-core decomposition. The notion of k-core was first introduced by Seidman (1983). A k-core community is defined as a maximal subgraph in which each node is adjacent to at least $k - 1$ nodes in the subgraph.

Saito et al. (2006) proposed the k-dense community method that extends the concept of k-core and approximates the k-clique method. The k-dense community is defined as a maximal subgraph in which each two-clique (i.e., pair of adjacent nodes that are connected by a link) has at least $k-2$ adjacent nodes in common that connect to both of the nodes in the clique, in the subgraph. The k-dense method is more computationally efficient than the k-clique method and as simple as the k-core method. It is obvious that a k-clique is included in a k-dense component, which is included in a k-core component.

We applied k-core and k-dense extraction methods to the Wikipedia network for all possibly k values. For each k, the network is divided and pruned, and a set of N communities $\{C_i^k\}_{1 \leq i \leq N}$ is obtained. Let C_{max}^k be the largest community in $\{C_i^k\}$ and D^k be the total number of nodes (articles) in $\{C_i^k\}$. Figure 2 illustrates how D^k and C_{max}^k change as k changes, for both k-core and k-dense methods. Figure 3 shows the number of communities for each k.

---

[2] We use the term k-clique as a clique consisting of k nodes.

From these figures, we can see that the k-core results consist of one dominant community and possibly other much smaller ones. The dominant community is either too large as a single community (for small ks) or is the only community extracted (for large ks), which makes the extraction results not sufficiently informative. On the other hand, the k-dense results consist of an appropriate number of smaller communities which are comparable in sizes for appropriate choices of ks and thus they are more informative.

For example, the k-dense method for $k = 12$ extracts total 37 communities. They include a community of baseball events, airplanes, airports, authors, and German cities. Twenty-two communities, including these five communities, belong to one same community if the k-core method for $k = 12$ is used.

Figure 4 shows a two-dimensional layout of the articles, calculated by using the spring method (Kamada and Kawai, 1989) (here, links are not shown). Because it is hard to plot all the 536,724 pages at once, articles not included even in low-degree communities are omitted from the spring-model calculation for computational efficiency. Thus, 9,773 pages are plotted in total including gray dots corresponding to articles in low-degree (6-dense) communities, and black dots (amount to 687 pages) in high-degree (12-dense) communities. The graph distance between articles are calculated from the original network. We can observe that the high-degree communities exist across the base network as clusters and form a characteristic structure.

4 Centralities

Each regular article in Wikipedia belongs to one or more categories, and each category has its own article (in the category namespace) to index corresponding regular articles. They form category hierarchies, but ones that are loose and loopy. In this section, we consider a directed network with 71,993 articles and 117,426 links in which a directed link corresponds to a reference, with its direction from a sub-category to a super-category.

We applied well known centrality measures such as PageRank (Brin and Page, 1998) and HITS (Kleinberg, 1999) to the Wikipedia category network. Because of the inherent nature of the category structure, we would expect that some fundamental concepts in the human knowledge should appear at the top of the rankings. Table 4 shows a portion of the category article rankings. PageRank ranking and HITS authority ranking, as well as the number of links directed to the article (in-link degree) and the number of articles that are reachable by iteratively following the in-links backward, are respectively shown in the columns labeled PRNK, HITS, INDEG and NRCH.

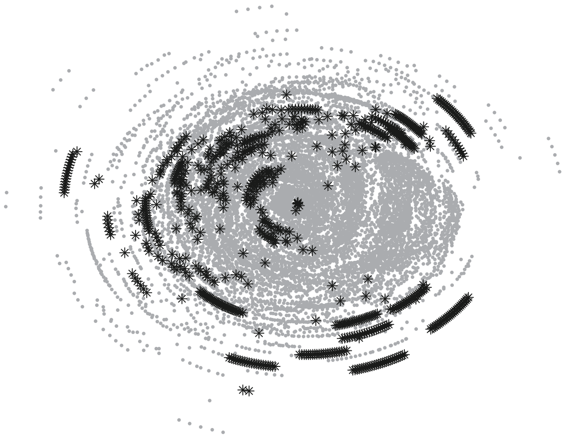


Figure 4: The Wikipedia network structure

The top five rows correspond to the top five articles ranked by PageRank. As expected, very fundamental concepts such as *Categories*, *Fundamental*, *Humans*, and *Cultures* as well as *Wikiportals*, which is Wikipedia specific, are listed. This suggests that PageRank most effectively reveals the basic Wikipedia category structure. PageRank importance is determined by "votes" in the form of links from other articles, and the importance of a vote from any source should be tempered (or scaled) by the number of articles for which the source is voting (Langville and Meyer, 2005). In the Wikipedia category network, the votes from the articles located in the lower level of the category hierarchy are accumulated to the articles in a higher level.

If we look for a ranking measure that gives those fundamental concepts high rankings, the number of reachable articles given in the NRCH column (the NRCH value) would be a straightforward alternative. Articles corresponding to fundamental concepts located near the top of the category hierarchy must have a large NRCH value. This is because iteratively following in-links backward corresponds to moving all the way down the hierarchy. In fact, the top five articles have all large NRCH values. However, it should be noted that the converse is not always true. For example, the article titled *Albert Einstein* has a large NRCH value but a lower PageRank ranking.

This gap can be attributed by the distribution of the number of steps to reach the rest of the articles. Figure 5 illustrates the relationship between the number

Table 1: Category article rankings

No.	Title	PRNK	HITS	INDG	NRCH
1	Categories	1	11662	5	62018
2	Fundamental	2	6586	9	62015
3	Humans	3	5468	20	61966
4	Culture	4	1749	62	61966
5	Wikiportals	5	5472	8	61966
6	Geography	13	1970	48	41991
7	Information	744	3806	4	61968
8	Albert Einstein	1834	12735	1	61967
9	Albums by artist	76	1	1387	1436
10	American albums	885	2	373	158
11	Canadian albums	1276	3	313	99
12	Alternative rock albums	923	4	255	135
13	British albums	1991	5	235	55

of steps to reach other articles (x-axis) and the number of articles at those steps' reach (y-axis). The number in parentheses for each article title is the PageRank ranking. The article *Categories* has a more concentrated and steeper distribution than *Albert Einstein*, which would explain the fact that the former has a higher PageRank ranking than the latter even though their NRCH values are almost the same. Likewise, the PageRank ranking of *Geography* is higher than *Information* and *Albert Einstein*, although the NRCH value for *Geography* is smaller. The fact that *Geography* has a more concentrated and steeper distribution than *Information* and *Albert Einstein* would explain this.

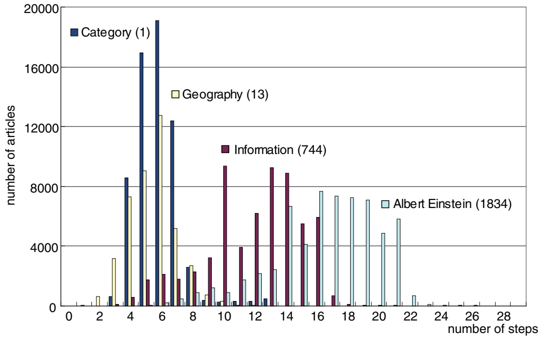


Figure 5: Distributions of the number of steps to reach other articles

The last five rows correspond to the top five articles ranked by HITS authority rankings. The set of articles with a high HITS ranking is quite different from that of a high PageRank ranking, and the fundamental concepts do not appear at all in the former. We know that HITS should be applied to a small network retrieved by a query and not to the entire network, but it is still worth investigating. In short, HITS rankings are affected strongly by in-link degrees: the top article *Albums by artist* has the highest in-link degree. In fact, the top fifty articles are all album-related articles that have strong connections with the top article. The top fifty hub articles are also all album-related ones.

5 Conclusions

We have investigated the structural properties of Wikipedia. The network grows by following the densification power law. The k-dense method gives us the most informative view of the community structure. In fact, we can observe that the high-degree communities exist across the base network as clusters and form a characteristic structure. PageRank most effectively reveals the basic category structure.

References

S. Brin and L. Page. The anatomy of a large-scale hypertextual Web search engine. In *7th int. conf. on WWW*, pages 107–117. Elsevier, 1998.

T. Kamada and S. Kawai. An algorithm for drawing general undirected graph. *Information Processing Letters*, 32:7–15, 1989.

J. Kleinberg. Authoritative sources in a hyperlinked environment. *J. ACM*, 46(5):604–632, 1999.

A. Langville and C. Meyer. A survey of eigenvector methods for Web information retrieval. *SIAM Review*, 47(1):135–161, 2005.

J. Leskovec, J. Kleinberg, and C. Faloutsos. Graphs over time: Densification laws, shrinking diameters and possible explanations. In *KDD '05*, pages 177–187. ACM Press, 2005.

G. Palla, I. Derenyi, I. Farkas, and T. Vicsek. Uncovering the overlapping community structure of complex networks in nature and society. *Nature*, 435(9):814–818, 2005.

K. Saito, T. Yamada, and K. Kazama. Extracting network communities based on the k-dense method. Technical report, NTT Communication Science Laboratories, 2006.

S.B. Seidman. Network structure and minimum degree. *Social Networks*, 5:269–287, 1983.

Social Insect Behaviour: Theory and Applications

5th April 2006

Organisers

Robert Planqué, University of Bristol
James Marshall, University of Bristol
Tim Kovacs, University of Bristol
Nigel Franks, University of Bristol
François-Xavier Dechaume-Moncharmont, University of Bristol
Ana Sendova-Franks, University of the West of England

Programme Committee

Nick Britton, University of Bath
François-Xavier Dechaume-Moncharmont, University of Bristol
Jean-Louis Deneubourg, Université Libre de Bruxelles
Nigel Franks, University of Bristol
Tim Kovacs, University of Bristol
James Marshall, University of Bristol
Kevin Passino, Ohio State University
Robert Planqué, University of Bristol
Francis Ratnieks, University of Sheffield
Ana Sendova-Franks, University of the West of England
David Sumpter, University of Oxford

Contents

Natural social concepts in simulated artificial societies

Pablo Lucas dos Anjos, Ruth Aylett
Heriot-Watt University, School of Mathematical and Computer Sciences
Edinburgh, EH14 4AS, United Kingdom
{anjos, ruth}@macs.hw.ac.uk

Abstract

The present workshop is mainly entitled to address decentralized decision making mechanisms by using local information and distributed behavioural rules; however this work does not seek an optimization-like discourse but rather an alternative perspective on the interplay between social adaptivity, and internal architecture for agents as a propelling social change engine.

Computational application developments, inspired by eusocial heuristic properties, is the most employed analogy of such characteristic mostly found in insects like termites, wasps and bees – but also observed on some mammal and shrimp species (Lacey and Sherman, 2005). Nevertheless there are other social animals closer to humans, such as primates, which present interesting organizational characteristics that might also benefit from simulated artificial eusocial concepts. For example, groups sharing fission-fusion organizational distributions, where each member can identify another, with the community size and its composition dependent on contextual and individual circumstances (Dunbar, 1993). Thus, groups are expected to change formation over time due to necessary arrangements for adaptation to social and environmental constraints. The fission is done by the highest ranked individual, splitting the composition into smaller groups for task specialization and compliance to circumstantial imposed restrictions. Conversely, a fusion is expected later on, for ensuring group consistence and benefits share. It could be argued that such observation is analogous to eusocial adaptivity, but differentiated by removal of anonymity and global social unawareness.

This approach suggests a focus shift to the identification of members and its interactivity within the society. Initial results on simulated tasks are then kept in a short memory system of each agent till the next fission event, when eusociality among those with close kinship, acts then like a social convention. In this kind of organization, all agents store a minimal history about its recent relationships. So its computational implementation must, then, be restricted to a much smaller amount of executing agents than eusocial ones – aiming to provide better responses to tasks and resource management (Castelfranchi, 2000). Improvements are then introduced on the fission process: the simulated leading agent may separate groups reactively, or pro-actively, using an internal emotional architecture biased by the eusocial kin selection process and the subject behavioural attributes.

As a consequence, a homologous analogy with simplified social norms (Takashi and Egashira, 2001) could be drawn upon this agency cognitive ability to manage social conventions, driven by contextual affective and utilitarian functions. Such influences depend on the fission and fusion causal relations, for example while establishing social relations for foraging and grooming behaviours. This proposal integration is therefore contextually grounded on socio-aware applications exploring behavioural-based modelling, while addressing agent action-selection architectures for coping with global and individual situated limitations.

References

C. Castelfranchi. Engineering Social Order. In Proceedings of the First international Workshop on Engineering Societies in the Agent World, *Lecture Notes In Computer Science*, Springer-Verlag, London, 1-18, 2000.

R. I. M Dunbar. Coevolution of neocortical size, group size and language in humans. *Behavioral and Brain Sciences,* volume 16 (4): 681-735, 1993.

Eeileen A. Lacey and Paul W. Sherman. Redefining eusociality: concepts, goals, and levels of analysis. *Annales Zoologici Fennici*, 42:573-577, 2005.

Takashi Hashimoto and Susumu Egashira. Formation of social norms in communicating agents with cognitive frameworks. *Journal of Systems Science and Complexity*, 14 (1), 2001, 54—74, 2001

The selection of building material in the ant *Temnothorax albipennis*

Antony Aleksiev
University of Bristol
antony.aleksiev@bris.ac.uk

Ana Sendova-Franks
University of the West of England
ana.sendova-franks@uwe.ac.uk

Nigel R Franks
University of Bristol
nigel.franks@bris.ac.uk

Abstract

Animals have to make choices. Unlike solitary species, the decisions made by social insects influence the survival and reproductive success of the whole colony. One example of such collective decision-making is nest building. Nest building in social insects has been studied in detail with the focus on its behavioural mechanisms and ecological significance. The selection of building materials has received little attention.

Social insects are central-place foragers. Foraging for building material bares parallels with the foraging for food. A theoretical prediction applicable to both is the exclusive choice for the most profitable option. In contrast, the currency of foraging decisions will differ between these two contexts. Building materials bring no energy gain like food does.

With a set of binary-choice experiments, we tested the preference of colonies of *Temnothorax albipennis* for two sizes of sand grains in nest building. *T. albipennis* colonies build some of the simplest nests in nature by merely encircling themselves with a dry-stone wall. When sand grains were foraged from different distances to the nest, the colonies exhibited partial preferences. The more profitable big grains dominated the choice except close to the nest where they were foraged equally with small grains. The profitability of the two building materials decreased with the increase of time costs, with big grains remaining the "optimal" choice at each distance. However, the wall always contained both grain sizes.

We tested the properties of sand piles such as angles of stability and packing density in considering the benefits from building mixture walls. Walls built with both grain sizes instead of either only big or small grains have greater stability and compactness.

We suggest a behavioural mechanism for optimal construction where individual foragers make "mistakes" by transporting small grains. The frequency of these "mistakes" relates inversely to the increasing time costs of grain transport. Hence, the foraging from different distances involves different likelihoods of acquiring the less profitable grain size. In addition, at longer distances, only a small number of foragers remain in action thereby having a greater opportunity for learning through repetition the more profitable grain size.

Partial preferences are optimal if not only the costs and benefits of the foraging but also the utilization of the retrieved materials is considered. In the case of the wall building in *T. albipennis*, foraging "mistakes" are beneficial for the resilience of the construction.

References:

S. Camazine, J-L. Deneubourg, N.R. Franks, J. Sneyd, G. Theraulaz, E. Bonabeau, *Self-organization in biological systems*. Princeton: Princeton University Press, 2003.

M. Hansell, *Animal architecture*. Oxford: Oxford University Press, 2005.

J.M. McNamara, A.I. Houston, Partial preferences and foraging, *Animal Behaviour,* 35, 1084-1099, 1987.

G.H. Orians, N.E. Pearson**,** On the Theory of Central Place Foraging, In: *Analysis of Ecological Systems* (Ed. by D.J. Horn, G.R. Stairs, R.D. Mitchell), pp. 155-177. Columbus: Ohio State University Press, 1979.

T.W. Schoener, Theory of feeding strategies, *Annual Review of Ecology and Systematics*, 2, 369-404, 1971.

T.W. Schoener, Generality of the size-distance relation in models of optimal feeding.
American Naturalist, 114, 902-914, 1979.

D.W. Stephens, J.R. Krebs, *Foraging Theory*. Princeton: Princeton University Press, 1986.

When slow is quick and quick is slow: ants optimize sequential recruitment methods

FX Dechaume-Moncharmont[⋆†]
[⋆]School of Biological Sciences,
University of Bristol,
Woodland Road, Bristol BS8 1UG, UK
`fx.dechaume@bristol.ac.uk`

Anna Dornhaus[†]
[†]Department of Ecology and Evolutionary Biology,
University of Arizona,
P.O. Box 210088 Tucson, AZ 85721 USA
`dornhaus@email.arizona.edu`

John M. McNamara[‡]
[‡]School of Mathematics,
University of Bristol,
University Walk, Bristol BS8 1TW, UK
`john.mcnamara@bristol.ac.uk`

Alasdair I. Houston[⋆]
[§]School of Biological Sciences,
University of Bristol,
Woodland Road, Bristol BS8 1UG, UK
`a.i.houston@bristol.ac.uk`

Edmund J. Collins[‡]
[¶]School of Mathematics,
University of Bristol,
University Walk, Bristol BS8 1TW, UK
`e.j.collins@bristol.ac.uk`

Nigel R. Franks[†]
[‖]School of Biological Sciences,
University of Bristol,
Woodland Road, Bristol BS8 1UG, UK
`nigel.franks@bristol.ac.uk`

Abstract

For a social insect colony, emigration from one nest to another is a risky venture. When its nest is damaged, a colony of *Temnothorax albipennis* has to emigrate as quickly as possible to the best available nest site. Two recruitment behaviours are involved in this emigration. A recruiter can either lead with a tandem run a single nestmate from the old nest to the new nest, or it can simply pick up a nestmate and carry it to the new nest. In both cases they can summon only one nestmate at a time. The tandem process is three times slower than carrying, but the ants recruited by tandem runs can learn the location of the new nest and are able to recruit other nestmates. However, carried ants may not be able to recruit others. Thus there must be a trade-off between these two tasks. What is the optimal task allocation that would minimize the emigration time? In this theoretical study, we compare two strategies. The first one assumes constant proportions of tandem runners and carriers. The second one assumes that each ant is able to perform both tasks. They begin by leading tandem runs until the number of ants recruited reaches a quorum threshold. Then every tandem run leader switches to social carrying. We show that a task allocation mechanism based on non-specialized workers sequentially involved in the two tasks performs significantly better than a strategy based on a strict division of labour with specialized recruiters. This theoretical prediction about task allocation mechanisms is consistent with experimental observations.

An Information Theoretic Approach to Stigmergy: The Case of Foraging Ants

László Gulyás[*][†]
[*]AITIA International Inc.
H-1039 Budapest
Czetz János utca 48-50, Hungary
`lgulyas@aitia.ai`

László Laufer[♣]
[♣]Budapest University of Technology
and Economics
H-1111 Budapest,
Műegyetem rkp.3-9, Hungary
`laufer@erg.bme.hu`

Richárd Szabó[†]
[†]Loránd Eötvös University
H-1117 Budapest
Pázmány Péter sétány 1/c, Hungary
`richard_szabo@invitel.hu`

Abstract

Algorithms inspired by the behavior of social insects became popular over the last decade. This popularity is due to the growing need to 'engineer emergent phenomena' in order to cope with today's and tomorrow's software engineering problems. A growing number of applications have been put forward based on the 'ant colony' approach, as well as a generalized and well-formalized metaheuristic under the name of Ant Colony Optimization (ACO). However, most of these algorithms and applications follow closely one of the well-known and widely discussed insect models, like ant foraging, ant sorting, or task differentiation. ACO generalizes the foraging model, making it applicable to a truly wider set of problems, but provides no deeper understanding about why exactly the heuristic works and under what specific circumstances. A limited number of papers addressed this issue, but the question still remains essentially open.

In this paper we propose an approach motivated by information theory to analyze this question. As a first step, we investigate how the foraging ant colony's performance depends on the 'information content' of the task at hand (approximated by the (dis)order in the initial configuration of food in the environment). We find that execution time depends about linearly on the initial disorder in case of a single food source. We also study the time-trajectory of food disorder ('entropy') together with the level of coordination in the ant colony. Our main finding is that the ant colony reaches its minimum entropy about the same time when the food disorder peaks. This is when the ants established the 'optimized' trail between the source and the nest. We also find that, in case of a single food source, increasing the disorder of the initial food configuration makes the colony's performance more sensitive to the stochastic elements governing its behavior.

These results show that this type of analysis has a potential for success in understanding the 'driving force' of stigmergetic algorithms. Given this understanding one would be in the position to design completely novel 'insect-like' distributed algorithms for complex problems.

Acknowledgements

The partial support of the GVOP-3.2.2-2004.07-005/3.0 (ELTE Informatics Cooperative Research and Education Center) grant of the Hungarian Government is gratefully acknowledged.

Performance of insect-inspired self-organized task allocation mechanisms

Franziska Kluegl[*]
[*]Artificial Intelligence, University of Würzburg
Am Hubland, D 97074 Wuerzburg
`kluegl@informatik.uni-wuerzburg.de`

Anna Dornhaus[†]
[†] Ecology and Evolutionary Biology, University of Arizona
P.O. Box 210088, Tucson, AZ 85721s
`dornhaus@email.arizona.edu`

Abstract

Who conducts which task in a group or team? This is one of the central questions for any form of division of labour, in social insects as well as in multi-agent systems. Real social insect division of labour achieves the efficiency, solution quality, robustness or flexibility that artificial systems aim for. The "superorganism" of ants or bees still is able to maintain its function even if complete task groups are taken away. Through evolution, the decision rules of every single individual ant are optimised in its environmental context and in coordination with the others. However, it is not trivial to simply transfer task allocation mechanisms found in social insects to engineering multi-agent systems like swarm robots. One obstacle is that there are several models of insect task allocation, yet from biological point of view it is not clear which social insect species applies what mechanism and why. This was the motivation behind our study, where we compare the performance of different task allocation mechanisms under different conditions and constraints.

Performance is tested in an abstract model where a number of A agents have to perform T tasks. Every task t is characterised by one of Y types and a dynamic value r_t that increases in every timestep and denotes the urgency of the task t. Execution of task t by agent a means that a reduces r_t by an amount dependent on ability, f_a. In the model, the basic decision mechanism is threshold-based, meaning that tasks are executed if their urgency exceeds a specific threshold. Each agent has distinct thresholds for different task types. Based on different configurations of this threshold vector we could model caste-like systems (each agent only executes one task type), purely stimulus-based systems (no threshold is active) or different randomly distributed thresholds. We varied number of agents, their abilities, number of tasks and number of task types. We compared the performance of the different threshold configurations with completely random and optimal centralised task allocation. Also, additional aspects of decision making have been tested that show how constraints on the individuals reasoning capabilities affect the performance of these task allocation mechanisms. For example, we have varied the task selection mechanism such that agents can either only assess urgency of one task per evaluation cycle or assess all tasks that they perceive; we have also varied whether they can take the current urgency or its derivates into account when selecting the next task. Analogous tests were made for the perception abilities concerning when to terminate task execution.

Overall, we could show that of the distributed task selection algorithms, random choice of the next task works surprisingly well in many situations. A fixed allocation (a 'caste' system) performs worse than the alternatives in most cases. If agents are able to compare several task objects at no cost, variable task thresholds in different agents improve collective performance as they improve selectivity and concentration on the most urgent tasks.

Similar Models of Optimal Decision Making in Social Insect Colonies and in the Primate Brain

James A. R. Marshall[*1], Rafal Bogacz[*], Anna Dornhaus[‡], Nigel R. Franks[†], Tim Kovacs[*], Robert Planqué[*†]
[*] *Department of Computer Science, University of Bristol, Bristol BS8 1UB, UK.*
[†] *School of Biological Sciences, University of Bristol, Bristol BS8 1UG, UK.*
[‡] *Department of Ecology and Evolutionary Biology, University of Arizona, Tucson AZ 85721, USA*
[1] marshall@cs.bris.ac.uk

Abstract

Striking parallels can be seen between current models of decision-making mechanisms in the primate brain, and of collective decision-making in social insect colonies. In a particular region of the primate brain, integration of noisy sensory information during a binary decision task has been modelled as populations of mutually-inhibiting neurons (Usher & McLelland, 2001). Recent work has examined how such a model can be parameterised to realise a statistically optimal decision-making strategy, the sequential-probability ratio test (SPRT), that minimises error rate for a given desired reaction time (Brown *et al*., 2004; Bogacz *et al*., 2006). Current models of decision-making during house-hunting in different species of social insects (*Temnothorax albipennis* and *Apis mellifera*) (Britton *et al*., 2002; Pratt *et al*., 2002) bear a striking resemblance to the neuron model proposed to describe decision-making in certain regions of the primate brain. We shall explore these similarities, and examine the extent to which the social insect models can be parameterised to realise the optimal SPRT strategy. We shall show that different models can either realise or approximate the optimal strategy for a binary choice problem. The crucial differences in the models arise from our current understanding of different recruitment mechanisms in the different species modelled. Comparisons between the models will enable us to make predictions about how recruitment should be carried out in each species if colonies are to be able to make optimal collective decisions, and evaluate these predictions against our current understanding of their behaviour.

References

Bogacz, Brown, Moehlis, Hu, Holmes & Cohen (2006) *The physics of optimal decision making: a formal analysis of models of performance in two-alternative forced choice tasks*. Under review.

Britton, Franks, Pratt & Seeley (2002) *Deciding on a new home: how do honeybees agree?* Proc. R. Soc. Lond. B 269, 1383-1388.

Brown, Gao, Holmes, Bogacz, Gilzenrat & Cohen (2005) *Simple neural networks that optimize decisions*. Int. J. Bifurc. Chaos 15, 803-826.

Pratt, Mallon, Sumpter & Franks (2002) *Quorum sensing, recruitment, and collective decision-making during colony emigration by the ant Leptothorax albipennis*. Behav. Ecol. Sociobiol. 52, 117-127.

Usher & McLelland (2001) *On the time course of perceptual choice: the leaky competing accumulator model*. Psych. Rev. 108, 550-592.

Honey Bee Swarm Cognition

Kevin M. Passino[⋆]
[⋆]Dept. Electrical and Computer Eng.
Ohio State University
passino@ece.osu.edu

Thomas D. Seeley[†]
[†]Dept. Neurobiology and Behavior
Cornell University
tds5@cornell.edu

Abstract

During nest-site selection, parts of a honey bee swarm can be viewed as a physical basis for cognition mechanisms. Individual bees are cognition units interconnected via dances and cues, and sensory units that are allocated to search or nest-site assessment tasks. Nest-site quality evidence accumulates in parallel for each candidate site, is built on "early" sensory processing and bee-to-bee communications, and is held in a spatially distributed group-level memory. The group memory is physically represented via multiple sets of bees dancing for different sites on the cluster and aggregates of bees at candidate sites. Group memory at the cluster is a filtered swarm-level estimate of the relative site qualities, while at the nest sites it represents the swarm's current preferences. A second layer of "late" processing is based on multiple uses of the group memory. At the cluster when recruits follow randomly encountered dances, the number of recruits directed to each site is proportional to the total number of dancers for each site, which is a group memory. Bees are allocated to search for potential nests, rather than nest-site assessment, when it is difficult for a bee to locate a dancing bee, and this is also a clear use of group memory. Simultaneously, layered processing occurs at the nest sites during quorum sensing which is a parallel self-referential check of swarm preference exceeding a threshold. Here, we conduct a series of simulated behavioral tests to evaluate the swarm's ability to (i) discriminate between site qualities even in the presence of significant individual bee nest-site assessment noise, (ii) avoid being misled by multiple inferior distractor nest sites and simultaneously focus on the best sites, and (iii) order the percentage of choices for each site according to relative nest-site qualities and thereby avoid negative context-dependent effects on choice performance. Next, we show that the parameters of the swarm cognition mechanisms and the characteristics of group memory have been tuned by natural selection to provide a balance between speed and accuracy of choice. Our analysis at multiple levels, from mechanisms and behavioral to the adaptation level, is a generic approach to simulation-based analysis of group cognition systems. For all our simulation studies we use the model in K.M. Passino and T.D. Seeley, "Modeling and Analysis of Nest-Site Selection by Honey Bee Swarms: The Speed and Accuracy Trade-off," Behavioral Ecology and Sociobiology, Vol. 59, No. 3, pp. 427-442, Jan. 2006.

It takes two to tango: the role of late recruitment in collective decision making

Robert Planqué⋆, James A. R. Marshall⋆, Tim Kovacs⋆,
François-Xavier Dechaume-Moncharmont†,
Nigel R. Franks†, Ana Sendova-Franks‡

⋆Department of Computer Science, University of Bristol, Bristol BS8 1UB, UK

†School of Biological Sciences, University of Bristol, Bristol BS8 1UG, UK

‡ Faculty of Computing, Engineering and Mathematical Sciences,
University of the West of England, Bristol BS16 1QY, UK

Abstract

The study of decentralized decision making in social insects and other social groups has revealed a number of prominent mechanisms such as positive feedback, inhibition of behaviours and response thresholds (Sumpter (2006)). House hunting in colonies of the ant *Temnothorax albipennis* is a prime example of collective decision making in which many of these mechanisms may be found (Mallon et al. (2001); Pratt et al. (2002)). When the old nest is destroyed, scouts go looking for potential new nests, recruit other ants to these nests through tandem-running (Franks and Richardson (2006)), and switch from recruitment to carrying by monitoring if a quorum of ants has been reached inside a new nest (Pratt (2005)).

One of the behaviours commonly observed during colony emigrations has sparked much speculation, and does not fit the above emigration paradigm. After the quorum has been met and recruiters have started carrying ants and brood, some of these recruiters perform so-called reverse tandem runs (RTRs) from the new to the old nest. Common sense dictates that if the ants are to invest a certain amount of time and effort into recruiting others, they should do this as early as possible, in order to optimize emigration speed. So why are RTRs necessary at all, and why do they appear so late?

Using a number of mathematical models, we explore different hypotheses that might explain the role of RTRs. These theoretical considerations lead us to suggest that RTRs are used when certain bottleneck situations are encountered. In particular, we argue that late recruitment happens because early tandem runs were impossible due to a lack of ants at the old nest to follow them. Apart from regular tandem runs, graded commitment to nests and quorum mechanisms, these ants thus seem to flexibly employ security mechanisms to control the decision making process.

References

N.R. Franks and T. Richardson. Teaching in tandem-running ants. *Nature*, 439:153, 2006.

E.B. Mallon, S.C. Pratt, and N.R. Franks. Individual and collective decision-making during the nest site selection by the ant *Leptothorax albipennis*. *Behav. Ecol. Sociobiol.*, 50:352–359, 2001.

S.C. Pratt. Quorum sensing by encounter rates in the ant *Temnothorax albipennis*. *Behav. Ecol.*, 16(2):488–496, 2005.

S.C. Pratt, E.B. Mallon, D.J.T. Sumpter, and N.R. Franks. Quorum-dependent recruitment and collective decision-making during colony emigration by the ant *Leptothorax albipennis*. *Behav. Ecol. Sociobiol.*, 52:117–127, 2002.

D.J.T. Sumpter. The principles of collective animal behaviour. *Phil. Trans. Roy. Soc. B*, 361:5–22, 2006.

Decision-making during Brood Sorting in Ants

Ana B. Sendova-Franks
School of Mathematical Sciences, University of the West of England, Bristol, Frenchay Campus, Coldharbour Lane, Bristol, BS16 1QY, UK

`Ana.Sendova-Franks@uwe.ac.uk`

Abstract

Brood sorting in ants is a prime example of the robustness and adaptability of decision-making in social insects. The adult ant workers in *Temnothorax* colonies organise the eggs, larvae, prepupae and pupae in concentric annuli so that the smallest items are in the middle and the largest on the periphery. When a colony emigrates to a new nest site the brood are sorted anew within 48 h.

Earlier work implied that the underlying mechanism for such sorting is a combination of the individual and mutually dependent decisions of workers about the time and place for picking up or putting down a brood item. Recent studies, however, suggest that brood sorting may also require consensus decisions. Experimental quantification of the dynamics of the process when associated with colony emigration, revealed two phases. During the first phase, all brood items are gathered together in a single cluster. During the second phase, items are moved in random directions so that the cluster spreads out. Larger items diffuse outwards more quickly than smaller items and this is likely to lead to the sorted structure.

New experiments, where the new nest was seeded with different brood patterns, dissociate the phenomenon of brood sorting from that of colony emigration to a new nest site. The results from these experiments demonstrate that the clustering phase is not a by-product of colony emigration and suggest that it is a necessary condition for sorting to occur.

These new results require a reconsideration of brood sorting as a decision-making process. First, they suggest that the clustering phase could be regarded as the building of a consensus decision about the location of the brood pile and hence the location of the colony in the new nest site. This is an important decision because potentially it determines the location of wall building as well as the locations of all other tasks in the division of labour. Second, these new results suggest that ants make a synchronised switch from clustering to spreading, which in turn requires a consensus decision.

A comparison of current understanding of the underlying mechanisms of brood sorting with current algorithms for object sorting inspired by ants in the domains of computer science and collective robotics pose the following question. Why do ants not appear to be using any of the proposed algorithms for the sorting of objects into structures that are compatible with the brood pattern in ants? Decision-making in ant colonies has evolved over millions of years of natural selection to be robust and adaptable to a plethora of complicated and often unpredictable concurrent situations. Computer algorithms tend to be perfectly fitted to a small number of conditions.

Decision-making by animal groups

David Sumpter
Department of Zoology
South Parks Road
Oxford OX1 3PS, UK
david.sumpter@zoology.oxford.ac.uk

Abstract

Joining behaviour, aggregation in response to the presence of a group of others, is a trait found in many diverse animal species. Chickens pecking at the ground near other birds, cockroaches sheltering together and humans following the gaze of a crowd of onlookers all involve an increased probability of joining a group dependent on its size. In these examples and many more the probability of joining increases sharply at a threshold number of group members. I argue that such threshold-based joining behaviour is central to apparent collective decision-making. I illustrate this argument with experimental data from ant foraging, pigeon homing and locust marching. Through simple mechanisms of positive feedback and response thresholds, these animal groups are able to choose between food sources and migration routes. I discuss how these groups can balance exploitation and exploration and how we might use their behaviour to design decentralised algorithms to solve optimisation problems.

Author Index