AISB'06

Adaptation in Artificial and Biological Systems

Volume 2

Editors: Tim Kovacs and
James A. R. Marshall

University of
BRISTOL

ML
BC

hp
invent

EPSRC
Engineering and Physical Sciences
Research Council

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

GC5: Architecture of Brain and Mind Integrating High Level Cognitive Processes with Brain Mechanisms and Functions in a Working Robot

3rd - 4th April 2006

Organisers

Aaron Sloman, University of Birmingham

Invited Speakers

Jackie Chappell, Univ. of Birmingham
Mike Denham, Plymouth University
Steve Furber, Manchester University
Jeffrey Krichmar, The Neurosciences Institute
Mark Lee, Univ. of Wales Aberystwyth
Peter Redgrave, Sheffield University
Murray Shanahan, Imperial College London
Aaron Sloman, University of Birmingham
Mark Steedman, University of Edinburgh
Tom Ziemke, University of Skövde

Contents

1 Invited speakers

Jackie Chappell

The University of Birmingham, UK

http://users.ox.ac.uk/.kgroup/jackie.html

How do animals gather useful information about their environment and act on it?

Slides online (PDF)

Abstract

Animals are much more successful than current robots in their ability to gather information from the environment, detect affordances, attribute causes to affects, and sometimes generate individually novel behaviour. What kinds of mechanisms might make this possible? I will discuss different mechanisms for acquiring information in animals, and their strengths and weaknesses given different life histories and niches. I will discuss experiments which have attempted to uncover the extent of animals' abilities to use information from their environment, and the mechanisms that might be used to accomplish this. The development of these kinds of competences (in evolutionary time and over the course of an individual's lifetime) is another interesting problem. Exploration and play seem to be very important for some kinds of behaviour, particularly flexible responses to novel problems, but there is also the possibility that animals come equipped with certain kinds of 'core knowledge', which might help to direct and structure the acquisition of more complex competences.

[1] R. C. Barnet, R. P. Cole, and R. R. Miller. Temporal integration in second-order conditioning and sensory pre- conditioning. Animal Learning and Behavior, 25:221–233, 1997.

[2] J. Chappell and A. Kacelnik. New Caledonian crows manufacture tools with a suitable diameter for a novel task. Animal Cognition, 7:121–127, 2004.

[3] S. E. Cummins-Sebree and D. M. Fragaszy. Choosing and using tools: Capuchins (*Cebus apella*) use a different metric than tamarins (*Sanguinus oedipus*). Journal of Comparative Psychology, 119(2):210–219, 2005.

[4] M. Domjan and N. E. Wilson. Specificity of cue to consequence in aversion learning in the rat. Psychonomic Science, 26:143–145, 1972.

[5] M. Hayashi and T. Matsuzawa. Cognitive development in object manipulation in infant chimpanzees. Animal Cognition, 6:225–233, 2003.

[6] A. A. S. Weir, J. Chappell, and A. Kacelnik. Shaping of hooks in New Caledonian crows. Science, 297(9 August 2002):981, 2002.

Mike Denham

Plymouth University

http://www.plymouth.ac.uk/pages/dynamic.asp
?page=staffdetails.id=mdenham.size=l

The role of the neocortical laminar microcircuitry in perception, cognition, and consciousness

Paper online (PDF)

Abstract

The talk will focus on the objectives of the EPSRC- and EU-funded projects I am involved in, including the role of the neocortical laminar microcircuitry in perception, cognition, and (dare I say it) consciousness. Of particular interest is the question of how the brain uses context to modify perceptual awareness, as illustrated for example by visual illusions.

Steve Furber

Manchester University

http://www.cs.manchester.ac.uk/apt/people/sfurber/

High-Performance Computing for Systems of Spiking Neurons

Paper online (PDF)

Abstract

We propose a bottom-up computer engineering approach to the Grand Challenge of understanding the Architecture of Brain and Mind as a viable complement to top-down modelling and alternative approaches informed by the skills and philosophies of other disciplines. Our approach starts from the observation that brains are built from spiking neurons and then progresses by looking for a systematic way to deploy spiking neurons as components from which useful information processing functions can be constructed, at all stages being informed (but not constrained) by the neural structures and microarchitectures observed by neuroscientists as playing a role in biological systems. In order to explore the behaviours of large-scale complex systems of spiking neuron components we require high-performance computing equipment, and we propose the construction of a machine specifically for this task - a massively parallel computer designed to be a universal spiking neural network simulation engine.

Jeffrey L. Krichmar

The Neurosciences Institute, San Diego, USA
http://www.nsi.edu/users/krichmar/

Brain-based devices for the study of nervous systems and the development of intelligent machines.

Paper online (PDF)

Abstract

Without a doubt the most sophisticated behavior seen in either biological or artificial agents is demonstrated by organisms whose behavior is guided by a nervous system. Thus, the construction of behaving devices based on principles of nervous systems may have much to offer. Our group has built series of brain-based devices (BBDs) over the last 14 years to provide a heuristic for studying brain function by embedding neurobiological principles on a physical platform capable of interacting with the real world. These BBDs have been used to study perception, operant conditioning, episodic and spatial memory, and motor control through the simulation of brain regions such as the visual cortex, the dopaminergic reward system, the hippocampus, and the cerebellum. Following the brain-based model, we argue that an intelligent machine should be constrained by the following design principles[1, 2]: (i) it should incorporate a simulated brain with detailed neuroanatomy and neural dynamics that controls behavior and shapes memory, (ii) it should organize the unlabeled signals it receives from the environment into categories without a priori knowledge or instruction, (iii) it should have a means to adapt the device's behavior, called value systems, when an important environmental event occurs, (iv) it should have a physical instantiation, which allows for active sensing and autonomous movement in the environment, (v) it needs to engage in a task that is initially constrained by minimal set of innate behaviors or reflexes, and (vi) it should allow comparisons with experimental data acquired from animal nervous systems. Like the brain, these devices operate according to selectional principles through which they form categorical memory, associate categories with innate value, and adapt to the environment. Moreover, this approach may provide the groundwork for the development of intelligent machines that follow neurobiological rather than computational principles in their construction.

1. Krichmar, J.L. and G.M. Edelman, Brain-based devices for the study of nervous systems and the development of intelligent machines. *Artif Life*, 2005. 11(1-2): p. 63-77.

2. Krichmar, J.L. and G.N. Reeke, The Darwin Brain-Based Automata: Synthetic Neural Models and Real-World Devices, in *Modeling in the Neurosciences: From Biological Systems to Neuromimetic Robotics*, G.N. Reeke, et al., Editors. 2005, Taylor & Francis: Boca Raton. p. 613-638.

Mark Lee

University of Wales, Aberystwyth
http://users.aber.ac.uk/mhl/

Developmental Robotics: an emerging paradigm for intelligent agents.

Abstract

Much research has explored the issues involved in creating truly autonomous embodied learning agents but only recently has the idea of a developmental approach been investigated as a serious strategy for robot learning. This is now emerging as a vibrant new research area. We examine the goals and methods of Developmental Robotics, and assess the current state of play. We give some requirements for a developmental system and relate these to the UK Grand Challenge 5 (The architecture of mind and brain) in terms of design issues for future robotic systems.

Peter Redgrave

Dept of Psychology, Sheffield University
http://www.shef.ac.uk/psychology/staff/academic/
peter-redgrave.html

Is it just a question of priority? Inspiration from the vertebrate basal ganglia.

Abstract

As soon as an agent, biological or physical, is provided with two or more parallel processing sensory or motivational systems that can guide movement there is a problem. Indeed, the same problem arises when a single system has the capacity to represent two or more features that can guide movement. If competing systems/features seek to guide incompatible movements (e.g. approach/avoidance) which one should be given priority ? Our supposition is that one of the vertebrate brain's fundamental processing units, the basal ganglia, has evolved to deal with such issues. Throughout the brains of vertebrates, parallel processing sensory, motivational and cognitive systems that can direct movement all provide phasic excitatory inputs to the basal ganglia. In turn, the basal ganglia output nuclei provide returning tonically active inhibitory connections to all input structures. Thus, the architectural principle describing basal ganglia connections with both cortical and sub-cortical systems is one of largely segregated parallel projecting loops. Winner-take-all selection is achieved by selective disinhibition of behavioural

systems targeted by basal ganglia output. Within this general framework, the implications of having of subcortical motivational systems (basic urges?) competing directly for behavioural expression with cortical (intellectual models of the world) will be considered. Additionally, an important quality of adaptive action selection systems is the capacity to adjust response probabilities (selections) based on reinforcement outcome. What is being reinforced by phasic dopaminergic neurotransmission within the basal ganglia is currently a topic of some dispute. Evidence will be considered suggesting dopaminergic teaching signals play a central role in identifying components of context and behaviour that are critical for causing unexpected biologically significant outcomes; in other words, learning those events for which the agent is responsible.

Murray Shanahan

Imperial College, London

http://www.doc.ic.ac.uk/˜mpsha/

Cognition and Consciousness: Is There a Fundamental Link?

Abstract

Some contemporary theories posit an intimate link between cognition and consciousness. For example, according to Baars's global workspace theory, the hallmark of consciously processed information is that it involves competition between and broadcast to widespread, multiple brain regions, while nonconscious information processing is localised. On this account, consciously processed information - because it integrates the activity of massively parallel processing resources, sifting out the relevant contributions given the ongoing situation for the organism - is cognitively efficacious in a way that non-conscious information processing is not. From the perspective of understanding the “architecture of mind and brain”, this suggests that the issue of consciousness cannot be ignored, but should be a central element of the research programme of our Grand Challenge.

Aaron Sloman

University of Birmingham

http://www.cs.bham.ac.uk/˜axs/

Requirements for a robot with human child-like or crow-like visual and learning capabilities.

Abstract

I regard explaining vision as the hardest unsolved problem in AI and psychology. In part that's because identifying the functions of vision is so difficult. What are the functions of vision? There are many AI and robot systems that include a small set of visual abilities, e.g. the ability to analyse static or changing video images, usually in a very limited way, e.g. identifying instances of a few types of objects (e.g. vehicles), or tracking moving objects treated merely as blobs, or locating a robot relative to a previously stored map. A vast amount of research is driven by benchmark-based competitions which use arbitrarily selected collections of tests, based on human performances that are not understood at all, and which do not relate vision to its animal functions in enabling and controlling actions in a 3-D world.

In contrast, in humans and many animals, vision involves a rich and deep variety of functions, including perceiving static and changing 3-D structures, perceiving many kinds of positive and negative affordances, controlling actions both ballistically and online, and (at least in humans) interpreting the intentions of others, reading text and music, interpreting gestures, understanding how some mechanism works, solving mathematical problems with the aid of diagrams, and many more.

I previously[1] thought (like many others) that most of these functions could be explained in terms of perception of *structure* at different levels of abstraction processed concurrently, from which perception of affordances (information about what is and is not possible) could arise. Recently, while working on 3-D manipulation tasks for the CoSy robot project[2], I realised that most normal perception is not of structures but of complex *processes* (represented at different levels of abstraction concurrently). For instance, as two objects move in relation to each other each typically has parts that move in relation to other parts and to parts of the other object. Thus we are surrounded by “multi-strand” processes.

This simple obsrvation has profound implications regarding requirements for explanatory models, which I shall attempt to explain. This is closely related to the Emulation Theory of Representation pre-

---

[1] http://www.cs.bham.ac.uk/research/cogaff/crp/chap9.html

[2] http://www.cs.bham.ac.uk/research/projects/cosy/PlayMate-start.html

sented by Rick Grush in BBS 2004.[3] In particular, detailed analysis of requirements for such capabilities sheds light on the variety of types of learning that need to occur, e.g. as a result of active and playful exploration of the environment, and also points to some deep requirements for cognition that are ignored by many researchers who emphasise the importance embodiment, for example the requirement to perceive "vicarious affordances" (affordances for others, or for oneself in the past or future). The mechanisms that provide this have been misleadingly labelled "mirror neurons" rather than "abstraction neurons". Moreover analysis of the implications reveals that as far as humans and some other altricial species are concerned, the role of embodiment is reduced compared with, for instance insects, and sensorimotor contingencies are replaced by more "objective" condition-consequence contingencies.

An incomplete overview of background ideas for my talk can be found in a PDF presentation on vision as perception of processes[4], a PDF presentation on Two views of child as scientist: Humean and Kantian[5] and this web page on Orthogonal recombinable competences acquired by altricial species.[6]

Mark Steedman

University of Edinburgh

http://www.cogsci.ed.ac.uk/˜steedman/

Plans and the Structure of Mind and Language

Abstract

For both neuro-anatomical and theoretical reasons, it has been argued for many years that language and planned action are related. I will discuss this relation using a formalization related to those used in AI planning, drawing on linear and combinatory logic. This formalism gives a direct logical representation for the Gibsonian notion of "affordance" in its relation to action representation. Its relation to universal syntactic combinatory primitives implicated in language is so direct that it raises an obvious question: since higher animals make certain kinds of plans, and planning seems to require a symbolic representation closely akin to language, why don't those animals possess language in the human sense of the term? I will argue that the lexicalization of recursive propositional attitude concepts concerning the mental state of others provides almost all that is needed to generalize planning to fully lexicalized natural language grammar. The conclusion will be that the evolutionary development of language from planning may have been a relatively simple and inevitable process. A much harder question is how the capacity for symbolic planning evolved from neurally embedded sensory-motor systems in the first place.

Tom Ziemke

University of Skövde, Sweden

http://www.ida.his.se/˜tom/

Integrating Cognition, Emotion and Autonomy: Embodied Cognition in Organisms and Robots

Abstract

Much research in embodied AI and cognitive science emphasizes the fact that robots, supposedly unlike purely computional models of cognition, are "embodied". However, in this talk it is argued that the physical embodiment that robots share with animals provides only one aspect of the "organismic embodiment" that is underlying natural cognition, emotion and consciousness. The talk discusses the living body's relevance to embodied cognition and agency, and outlines a European research project that aims to model the integration of cognition, emotion and bioregulation (self-maintenance) in robots.

---

[3] http://journals.cambridge.org/action/displayIssue?jid=BBS.volumeId=27.issueId=03

[4] http://www.cs.bham.ac.uk/research/projects/cosy/papers/#pr0505

[5] http://www.cs.bham.ac.uk/research/projects/cosy/papers/#pr0506

[6] http://www.cs.bham.ac.uk/research/cogaff/misc/orthogonal-competences.html

2 Poster summaries

Ruth Aylett

MACS, Heriot-Watt University, Riccarton, Edinburgh EH10 4ET

Emotion as an integrative process between non-symbolic and symbolic systems in intelligent agents

Abstract

This paper briefly considers the story so far in AI on agent control architectures and the later equivalent debate between symbolic and situated cognition in cognitive science. It argues against the adoption of a reductionist position on symbolically-represented cognition but in favour of an ac- count consistent with embodiment. Emotion is put forward as a possible integrative mechanism via its role in the management of interaction between processes and a number of views of emotion are considered. A sketch of how this interaction might be modelled is discussed.

Joanna Bryson

Artificial models of natural Intelligence, University of Bath, United Kingdom

Embodiment vs. Memetics: Is Building a Human getting Easier?

Abstract

This heretical article suggests that while embodiment was key to evolving human culture, and clearly affects our thinking and word choice now (as do many things in our environment), our culture may have evolved to such a point that a purely memetic AI beast could pass the Turing test. Though making something just like a human would clearly require both embodiment and memetics, if we were forced to choose one or the other, memetics might actually be easier. This short paper argues this point, and discusses what it would take to move beyond current semantic priming results to a human-like agent.

Nick Hawes, Aaron Sloman and Jeremy Wyatt

School of Computer Science, University of Birmingham

Requirements & Designs: Asking Scientific Questions About Architectures

Abstract

This paper discusses our views on the future of the field of cognitive architectures, and how the scientific questions that define it should be addressed. We also report on a set of requirements, and a related architecture design, that we are currently investigating as part of the CoSy project.

John Knapman

School of Computer Science
The University of Birmingham (Formerly IBM)

Integration and Decomposition in Cognitive Architecture

Abstract

Given the limitations of human researchers' minds, it is necessary to decompose systems and then address the problem of how to integrate at some level of abstraction. Connectionism and numerical methods need to be combined with symbolic processing, with the emphasis on scaling to large numbers of competencies and knowledge sources and to large state spaces. A proposal is briefly outlined that uses overlapping oscillations in a 3-D grid to address disparate problems. Two selected problems are the use of analogy in commercial software evolution and the analysis of medical im- ages.

Hagen Lehmann

University of Bath

How to build a brain - An Evolutionary Approach

(Poster only, no paper)

Abstract

To simulate processes in the human brain it will be very useful to create a model of how individuals plan and how the decision making in the process of action selection works. It would be important to understand what consciousness is, how it evolved and why humans have the ability to become aware of them selves. Since these phenomena are the results of evolutionary processes, the key to create something like a human brain is to replicate and model these processes. The question is not what material we would need to build a brain, the question is how and can we describe and model the complex operations the human brain does. In order to answer this question we have to gain consolidated knowledge about the natural processes, which made our brain evolve to the state it is in now. How are humans able to learn in a way, that they can adapt their behaviour very quickly in dynamic environments in order to respond to new situations. The concept of intentionality and the concept of consciousness seem to play an important role in this action selection process.To be able to make complex plans about what to do in the future and then overthrow them quickly if the situation makes it necessary an individual has to understand itself and other individuals in its group as intentional agents. The evolution of language is another key problem to be addressed thinking about how human intelligence works. In order to communicate ideas and solutions for certain problems it was necessary to develop a system of abstract vocal symbols.

Maria Petrou and Roberta Piroddi

Department of Electrical and Electronic Engineering,
Imperial College London

On the structure of the mind

Abstract

The focus of any attempt to create an artificial brain and mind should reside in the dynamic model of the network of information. The study of biological networks has progressed enormously in recent years. It is an intriguing possibility that the architecture of representation and exchange of information at high level closely resembles that of neurons. Taking this hypothesis into account, we designed an experiment, concerning the way ideas are organised according to human perception. The experiment is divided into two parts: a visual task and a verbal task. A network of ideas was constructed using the results of the experiment. Statistical analysis showed that the verbally invoked network has the same topological structure as the visually invoked one, but the two networks are distinct.

Mark A. Wood

University of Bath, Department of Computer Science

Artificial models of natural Intelligence

Abstract

Social learning is an important source of human knowledge, and the degree to which we do it sets us apart from other animals. In this short paper, I examine the role of social learning as part of a complete agent, identify what makes it possible and what additional functionality is needed. I do this with reference to COIL, a working model of imitation learning.

Acknowledgements

Thanks to the euCognition network for funding this symposium, and for members of the EU-funded CoSy project for help with ideas and with arrangements for the symposium.

Introduction to Symposium
GC5: Architecture of Brain and Mind
Integrating high level cognitive processes with brain mechanisms and functions in a working robot

Aaron Sloman*†
*Computer Science, University of Birmingham UK
http://www.cs.bham.ac.uk/~axs/
`A.Sloman@cs.bham.ac.uk`

Abstract

This symposium is inspired by UKCRC Research Grand Challenge 5: Architecture of Brain and Mind. The aim is to provoke unified discussion of long term research goals in AI, Cognitive Science, and related disciplines, especially goals concerned with giving computers a useful and general subset of human capabilities, implemented in a biologically inspired fashion. The symposium can also be seen as part of a series of related events attempting to promote a high-level long-term vision of achievable scientific goals of AI/Cognitive Science, including The DAM (Designing an Mind) Symposium at AISB'00 (Davis, 2005), the Tutorial on Philosophical Foundations of AI at IJCAI'01 (Sloman and Scheutz, 2001), the St. Thomas symposium in 2002 (Minsky et al., 2004), and the IJCAI'05 Tutorial on Learning and Representation in Animals and Robots (Sloman and Schiele, 2005). It presents themes central to the EC-funded Cognitive Systems initiative,[*] including the CoSy project[†] which is part of that initiative, whose members have helped to organise this symposium, and the euCognition project[‡] which is funding this meeting. A common feature is the focus on *scientific* goals rather than useful *applications* though implementation of working systems is central to the proposed methodology. This introduction to the symposium provides some background and highlights some of the major problems to be overcome.

1 Introduction

In October 2002, under the auspices of The UK Computing Research Committee (UKCRC)[1], Tony Hoare and Robin Milner initiated discussions of "grand challenge" research projects in computing[2]. Seven grand challenge proposals emerged, listed in the booklet available on the UKCRC web site. One of them was "GC-5: Architecture of Brain and Mind — Integrating high level cognitive processes with brain mechanisms and functions in a working robot."[3]

It is concerned with the attempt to understand and model natural intelligence at various levels of abstraction, demonstrating results of our improved understanding in a succession of working robots, along with a succession of increasingly realistic implementations of models of brain mechanisms capable of implementing the competences to be explained.

Robots produced within this grand challenge project should have an interesting and challenging subset of the capabilities of a child aged somewhere between 2 and 5, including the ability to go on learning, and the ability (some of the time) to understand what they are doing and why. One way for such a robot to demonstrate all of that functionality would be being capable of helping and conversing with a disabled person who wishes to avoid being dependent on other humans, at least around the house, without the robot first having to be programmed explicitly with knowledge about that house and its contents, and that person's needs and preferences.

However, it is not enough to produce something that works: we can already do that, thanks to biological evolution. The deep problem that makes this a scientific challenge is explaining how this is possible.

---

[*] http://www.cordis.lu/ist/cognition/projects.htm
[†] http://www.cognitivesystems.org
[‡] http://www.eucognition.org/
[1] http://www.ukcrc.org.uk/
[2] http://www.ukcrc.org.uk/grand_challenges
[3] http://www.cs.bham.ac.uk/research/cogaff/gc

2 The need to re-integrate AI

Achieving this scientific understanding requires us to bring together work in neuroscience, cognitive science, various areas of AI, linguistics, and other relevant disciplines, to produce an integrated theory of how a functioning system can combine many human capabilities, including various kinds and levels of perception, different kinds of reasoning, planning, problem solving, wondering about, varieties of learning (including grasping new abstract concepts and developing new fluent skills), many kinds of actions of varying complexity, different uses of language, varieties of affect including motivation and emotions, social interaction, and various forms of creativity.

Current robots perform many tasks (some practical, some merely for entertainment) but usually they do not combine their perceptual and manipulative skills with the ability to communicate and cooperate, and they do not know what they are doing, why they are doing it, what difference it would make if they did things in a different way, or how they would have had to change their actions if circumstances had changed, etc., and they cannot give help or advice to another robot or a person performing such tasks.

The most advanced chess playing programs could be installed in a robot, but that would not enable the robot to detect the need to adjust the level of its play to help a beginner, as even a not very advanced human chess player might.

Most current robots need to be given goals because they have no internally generated motives or concerns of their own (although if they include AI planning mechanisms they can generate subgoals of externally provided goals). In particular, they lack the playful, exploratory, curiosity-driven activities that seem to enable human children and some other animals to learn much about their environment in a relatively short time.

However impressively current robots may perform specific tasks on the factory floor or in some demonstration, they do not have the variety of competences, the integration, or the self-understanding of a 3 or 4 year old child and they cannot learn most of the things a child can learn.

3 Why not? What is missing?

There are many reasons for these limitations in the current state of the art, but some of the main ones are:

- We lack deep, comprehensive characterisations of the competences of typical children, at any age, including their visual and other perceptual competences, their manipulative, deliberative, problem-solving, communicative competences, what sort of ontology they use and how it changes, what their motivations and affective states are, what kinds of development and learning go on.
- We do not know what sorts of architectures, forms of representation, virtual machines, brain mechanisms, are capable of producing those competences, including the division between innate, genetically determined, mechanisms and information structures, and what is, grown, developed or learnt as a result of interacting with the physical and cultural environment (Sloman and Chappell, 2005).

It is also fair to say that, although we can specify in very general terms what is required of a useful domestic robot, e.g. the ability to learn to find its way around the building, using vision and other sensors, the ability to manipulate domestic objects of various sorts, the ability to communicate in natural language well enough to obey instructions, answer questions, accept advice, and offer help, we don't really have any clear and detailed set of requirements that are both worth aiming for and eventually achievable.

For instance, although there is often reference to the need for AI systems to be able to 'scale up' to 'human-level' competence, it is generally forgotten that humans do not scale up: as tasks become more complex we can degrade quite drastically, and in many narrowly defined tasks machines are already better than most humans, for instance doing arithmetic, playing chess and controlling sophisticated modern aeroplanes. Although machines often scale up better in specific tasks they do not do what I've called 'scaling out', namely combining old competences in novel ways as circumstances demand.

Another thing that is often said, following (Gibson, 1986), is that robots need to be able to perceive and understand *affordances*, but people have different views of what affordances are, and as far as I know there is no representative list of affordances a good domestic robot will need to perceive and use.

4 Developing a 'roadmap'

So among the main tasks involved in the Grand Challenge are specifying long term requirements and explaining what sorts of mechanisms are capable of satisfying those requirements. Both tasks are so difficult and will last so long into the future that we cannot hope to get them right soon. AI 'prophets' of all fashions and factions have been notoriously over-optimistic in the past, bringing the whole field into disrepute as a result. Can we avoid this mistake?

In the booklet for the IJCAI'05 Tutorial on Learning and Representation in Animals and Robots[4], a scenario-based strategy for achieving greater realism in goals and timescales was proposed. This required collaboratively producing a partially ordered network of scenarios of various kinds and degrees of complexity, ordered according to dependency relationships and difficulty. If such a graph ended with a wide range of scenarios representing, for example, performances of a child-like but very useful and congenial autonomous domestic robot able to help a partially disabled (e.g. blind) person in a typical house, and included very many intermediate scenarios, with relatively small steps between the different stages, all shown to be ultimately dependent on scenarios that might be achieved in the next few years, then that could define a long term 'roadmap' or collection of roadmaps that could be used both to guide research plans and to measure progress.

We would know at any time which bits of the graph of scenarios had been achieved and what remained to be done. People could agree on that without necessarily agreeing on which methods, architectures, forms of representations, algorithms, mechanisms, etc. should be used. But the agreed roadmap could provide a common way evaluating progress which is now lacking except for very narrow, specialised, and often arbitrary benchmarks (e.g. sets of images to be classified – a task that may not have much to do with the use of vision in action).

Of course our understanding of what should and should not go into the graph would continue to develop, so the graph would not be fixed permanently from the start: one effect of research inspired by it would be to rebuild the roadmap as we learn more both about requirements and about usable mechanisms and designs.

5 Methods and tools to help build roadmaps

A problem with this proposal is that many people find it very difficult to think up a systematic and comprehensive collection of future scenarios of the kind required. (That was the experience of members of the CoSy project team at Birmingham, for instance.) So we have been working on a methodology and some (initially simple) supporting tools to help with development of this network of roadmaps.

The idea is to think in terms of a three-dimensional grid of competences. One dimension represented by columns in Figure 1,[5] is concerned with types of *entity* to which competences can be applied (e.g. 2-D and 3-D spatial locations, regions, routes, inert objects, mobile objects, objects with goals, perception, and action, and various kinds of more abstract objects such as beliefs, proofs, numbers, plans, concepts).

Entity-types **Competences**	**E1**	**E2**	**E3**	**E4**	**E5**	**E6**	**E7**	**E8**
C1								
C2								
C3								
C4								
C5								
C6								
C7								

Figure 1

Another dimension illustrated in the rows in Figure 1, is concerned with types of *competence* that can be applied to instances of some or all of the types of entities; for instance competences like perceiving, manipulating, referring to in thought, referring to in language, constructing, destroying, explaining, wondering about, and many more. These two dimensions determine a grid of possible sets of requirements that could form targets for a robot project. Some of the boxes in the grid would, of course, be empty. For instance some of the abstract entities, such as numbers, beliefs, and plans cannot acted on physically. So there would be no scenario examples of such actions.

The third dimension of the grid could be thought of as the depth of the boxes. Within a category some scenarios would be very complex, very difficult and a long way into the future, whereas others might be much more easily and quickly attainable. An example of the former might involve a robot reliably able to clear away things from a dinner table and put dirty objects into a dishwasher, including glasses, plates, cups sauces, knives, forks, etc., while putting other things into drawers and cupboards, or the refrigerator. A much simpler task could involve the robot using a special-purpose device to carry a book from a shelf to its owner.

When large numbers of examples of compe-

---

[4] http://www.cs.bham.ac.uk/research/projects/cosy/conferences

[5] And at http://www.cs.bham.ac.uk/research/projects/cosy/matrix, and described in http://www.cs.bham.ac.uk/research/projects/cosy/papers/#dp0602 'Towards a Requirements Grid: A Conceptual Framework and Draft Tool for Generating Requirements and Scenarios'

tence/object pairs have been described, and their dependencies analysed, it would be possible to define scenarios involving different subsets of entries in the grid.

A new kind of research project could be concerned almost entirely with producing entries for the grid and analysing their dependencies, with a view to identifying promising short term targets that are very likely to be intermediate steps towards very difficult long term goals. Both collaborating and competing groups of researchers could use sets of entries from the grid to specify agreed benchmarks and milestones.

The idea of a rectangular grid of rows and columns was introduced here merely for the sake of exposition. In fact, as already implied by the claim that some boxes will be empty, a more complex structure for the grid will be needed. For instance some of the boxes may have many more sub-divisions than others. There is also a need to be able to refer conveniently to combinations of competences in different parts of the grid as forming a new competence, for example using what is seen to disambiguate a spoken sentence while it is being uttered, or using what someone is saying to facilitate seeing a complex structure. These refinements and elaborations of the grid are topics for future research.

6 Implementation in biological mechanisms

One strand of the research programme as described in the previous sections involves the top-down specification of requirements and exploration of possible ways of meeting those requirements using whatever tools and mechanisms seem to be up to the task. However it is possible that there are deep, not yet understood, difficulties in doing this that mean that we shall eventually learn that only mechanisms with many of the important properties of biological brains are capable of supporting such working human-like performances. However, it is totally implausible that existing computational models of brain mechanisms are anywhere near being adequate to the task.

So a major feature of this grand challenge is that in parallel with the top down research into requirements and design possibilities there should be bottom up (and middle out) research both investigating what the biological mechanisms are and how they work and also attempting to produce artificial systems based on the same principles. It is not likely that merely applying empirical techniques of psychology and neuroscience will suffice to unravel these mysteries, without thinking about designs for virtual machines able to support the many competences that can be observed both in natural settings and in laboratory experiments. If researchers do not know which high level capabilities the brain mechanisms need to support they may not notice or investigate subtle features of the mechanisms that are required for such support, just as a physicist or electronic engineer studying computers without knowing anything about operating systems, compilers, virtual memory, privileges, security, recursive languages, and so on, will probably not come up with a good description of what a computer, considered as an electronic machine, is and does.

Even if everything can be done using conventional computers, the aim of this grand challenge, is not *merely* to understand how such diverse functions can be integrated in single system at a high level of abstraction which might be modelled on computers, or future artificial information-processing machines, but also to explain how they can be implemented in actual biological mechanisms. So an aim of the project is to continue developing our understanding of brain mechanisms (e.g. chemical, neural, etc. mechanisms) including showing how those mechanisms are able to support the high level functionality required by a child or human-like robot robot.

For this purpose, natural minds can be viewed as virtual machines implemented in brains. Since human minds surpass artificial minds in many ways at present, we may discover that this is partly due to using a different kind of physical implementation from current computers. There could be other reasons: it may be that our current designs for AI systems are simply far too simple because we have not yet understood what kinds of functionality they need nor what kinds of architectures, forms of representation and algorithms can provide those kinds of functionality in an integrated system.

Another Grand Challenge being pursued in parallel with this one is GC7: Journeys in Non-Classical Computation, investigation forms of computation that are different from those expressed in conventional programs using conventional computers.[6]. It may be that as both GC5 and GC7 make progress they will have to be more closely intertwined.

7 GC5 and the EC Cognitive Systems initiative

By coincidence, at the same time as this Grand Challenge project was being discussed in the UK in late 2002 and 2003, the European Commission was for-

---

[6] See http://www.cs.york.ac.uk/nature/gc7/

mulating a very closely related initiative, the *Cognitive Systems* initiative of Framework Programme 6. Since then that initiative has begun to fund a variety of projects, as described on its website[7], including the recently initiated euCognition project[8] which is funding this symposium – for which we are very grateful. Several of the projects already being funded include people who were involved in discussions of the Grand Challenge.

8 Themes for GC5 and the symposium

This grand challenge is far too complex and ambitious to be covered exhaustively in a two-day symposium. The following collection of themes was set out as possible topics for discussion when the symposium was proposed. However it is not possible to cover more than a small subset in the time available, so the list can serve as background to the presentations and help to determine what sorts of comments and questions are relevant. Additional context for the symposium is provided by the enduring symposium website, along with other events, past[9] and future, will help to stimulate discussion long after the conference is over.

A guiding principle in formulating the themes is to replace destructive factional debates about which are the right goals, methods and theories with constructive collaborative analysis of the alternatives and their tradeoffs; and to provide agreed ways of determining whether progress has been made towards the long term goals, avoiding the criticism made by H.L. Dreyfus that results achieved in AI are no more progress towards its goals than climbing trees is progress towards travel to the moon.

Theme 1. Requirements
What needs to be explained/modelled and how can we check that we have good requirements specifications?

This is the core question that drives everything else. There are many things humans can do and selecting a set of competences to be explained and modelled requires great care. The focus on abilities, including learning abilities of young children, arose from the observation that adult competences are typically based on a vast amount of individual learning and idiosyncratic history, whereas the common competences of young children are the basis for many kinds of future development in different cultures and different physical environments, from cave-dwellings to homes in skyscrapers.

But that still leaves open what those competences are and that requires extremely careful observation. (Notice that finding what sorts of competences children are capable of developing is different from doing research on the precise age at which they occur or whether particular environments can accelerate or retard their development.)

Often it is not clear what some human competence is until many examples have been analysed. E.g. some people think that what needs to be explained about vision is simply how a depth-map is computed – distance to contact in all directions. Others believe that the function of a visual system is to segment retinal images and recognise objects. Three more very different alternative requirements specifications are found in the work of D. Marr, in J.J. Gibson and people who emphasise dynamical systems. It is arguable they have all done only a *partial* requirements analysis for vision (or more generally perception).

Similar things can be said about what needs to be explained/modelled regarding: learning, motivation, emotions, affect in general, linguistic ability, reasoning, action control, mathematical abilities, creativity, exploration-based learning, aesthetic capabilities, humour, consciousness, etc.

Work on the requirements grid may help to clarify many of these issues though there is still a vast amount to be done.

Theme 2. Empirical evidence and theories
Research on the grand challenge needs to be informed by research in other empirical disciplines, including biology (e.g. animal behaviour, evolutionary theories), neuroscience, linguistics, cognitive/clinical/developmental psychology, social sciences, that helps either to refine the requirements or support/contradict proposed models and explanations (theme 3).

In particular work on empirical evidence from very young children, unusual humans (e.g. people with brain damage) or other animals can be useful in helping to avoid narrow thinking based on what normal adult humans (in our culture) can do.

What other animals do may provide evidence about evolutionary precursors and about unnoticed subsystems in human capabilities.

Theme 3. Designs for Models/Explanations
This includes:

---

[7] http://www.cordis.lu/ist/cognition/projects.htm

[8] http://www.eucognition.org/

[9] E.g. the UKCRC grand challenge conferences in 2004 and 2006, the EC Cognitive Systems ‘Kickoff’ Conference http://www.cognitivesystems.org and also the 2005 IJCAI tutorial on learning and representation in animals and robots http://www.cs.bham.ac.uk/research/projects/cosy/conferences

3.1. High-level (virtual machine) designs
proposed as meeting some subset of requirements (whether for explanation or for applications). [The 'top-down' approach.]
3.2. Implementation designs
3.2.a. Natural/biological mechanisms
3.2.b. Artificial mechanisms for neural/ chemical/developmental mechanisms proposed as capable of supporting the high level designs. [The 'bottom-up' approach.]
Suitable topics for discussion would include reports on current work in progress, and reports on current systems with analyses of how they are inadequate (they are ALL inadequate, in relation to the long term goals of GC5), and how those inadequacies may be overcome or reduced.

I.e. *merely* reporting on what some system can do and how it does it is not appropriate for a GC5 symposium.

Theme 4. Philosophical/conceptual issues
Discussion of what is meant by ascribing various kinds of capabilities and processes to animals or machines, and whether conceptual categories currently in use (e.g. cognition, learning, intentionality, sub-symbolic, emotion, information-processing, consciousness) are confused and in need of refinement.

Eg. do our current concepts allow us to discuss adequately which organisms do or do not process information, or use representations, or have motives or beliefs. Instead of focusing so much on particular capabilities and how to implement them we also need research into a good taxonomy (or other framework) of varieties of types of information-processing capability.

Theme 5. Recommendations for managing GC5
There is an important need for a project like this to have some sort of roadmap, even if it is regularly revised, and criteria for assessing progress.

Developing the three-dimensional requirements grid described above (and improving its structure) could be part of the process of producing such a roadmap. It is also important to resist fragmentation of effort and destructive rivalries.[10]

Theme 6. Social/ethical implications
Many people think social and ethical implications should always be discussed as part of such a project. Often such discussions tend to be either very shallow or wildly speculative or based on some strong ethical bias against advances in AI, or some combination of all of those. Care will be needed to ensure that we do not fall into such traps.

There are potentially profound applications that could follow from significant progress with GC5, not just in the obvious areas of building new smart robots and other machines, but in connection with applications arising from a deeper understanding of how humans work, including new ways of doing education, counselling, therapy, diagnosis of brain disorders and other kinds of mental disorders, and perhaps new forms of treatment.

I am especially interested in ways in which understanding better what enables a toddler to grow up to be mathematician. Such understanding could revolutionise mathematics education at all levels, including primary schools. But discussing all this may still be premature.

Acknowledgements

Thanks to the euCognition network for funding this symposium, and for members of the EU-funded CoSy project for help with ideas in this introduction and with arrangements for the symposium.

References

D. N. Davis, editor. *Visions of Mind: Architectures for Cognition and Affect*. Idea Group Inc., Hershey, PA, USA, 2005.

J.J. Gibson. *The Ecological Approach to Visual Perception*. Lawrence Erlbaum Associates, Hillsdale, NJ, 1986. (originally published in 1979).

M. Minsky, P. Singh, and A. Sloman. The St. Thomas common sense symposium: designing architectures for human-level intelligence. *AI Magazine*, 25(2):113–124, 2004. http://web.media.mit.edu/˜push/StThomas-AIMag.pdf.

A. Sloman and J. Chappell. The Altricial-Precocial Spectrum for Robots. In *Proceedings IJCAI'05*, pages 1187–1192, Edinburgh, 2005. Available at www.cs.bham.ac.uk/research/cogaff/05.html#200502.

A. Sloman and M. Scheutz. Tutorial on philosophical foundations: Some key questions. In *Proceedings IJCAI-01*, pages 1–133, Menlo Park, California, 2001. AAAI. http://www.cs.bham.ac.uk/˜axs/ijcai01.

A. Sloman and B. Schiele, editors. *Tutorial on Learning and Representation in Animals and Robots*. IJCAI'05, Edinburgh, 2005. Available at www.cs.bham.ac.uk/research/projects/cosy/conferences.

---

[10] For more on this see
http://www.cs.bham.ac.uk/research/cogaff/gc/targets.html (discussion of targets, milestones, in the context of a scenario-based research methodology), and
http://www.cs.bham.ac.uk/research/projects/cosy/papers/#tr0503 (extract from the IJCAI'05 tutorial booklet.)

How do animals gather useful information about their environment and act on it?

Jackie Chappell*
*Centre for Ornithology, School of Biosciences
University of Birmingham, Edgbaston, Birmingham B15 2TT, UK
`j.m.chappell@bham.ac.uk`

Abstract

Animals are much more successful than current robots in their ability to gather information from the environment, detect affordances, attribute causes to affects, and sometimes generate individually novel behaviour. What kinds of mechanisms might make this possible? I will discuss different mechanisms for acquiring information in animals, and their strengths and weaknesses given different life histories and niches. I will discuss experiments which have attempted to uncover the extent of animals' abilities to use information from their environment, and the mechanisms that might be used to accomplish this. The development of these kinds of competences (in evolutionary time and over the course of an individual's lifetime) is another interesting problem. Exploration and play seem to be very important for some kinds of behaviour, particularly flexible responses to novel problems, but there is also the possibility that animals come equipped with certain kinds of 'core knowledge', which might help to direct and structure the acquisition of more complex competences.

How do animals gather useful information about their environment and act on it?

Jackie Chappell

Center for Ornithology
School of Biosciences
University of Birmingham
`j.m.chappell@bham.ac.uk`

What is involved in gathering information and acting on it?

- How do you perceive objects in ways that allow manipulation?
- What do you pay attention to (filtering and selective attention)?
- How do you detect affordances?
- How do you assign causality to actions, events or agents?
- How can competences be re-combined flexibly to generate appropriate behaviour in novel contexts, or creativity?
- How does this all develop?

If you were trying to build a robot to behave spontaneously like the chimp in the following clip, how would you do it?

Pal, 2.5 years old

video taken by Misato Hayashi, Primate Research Institute, Kyoto University, used with permission

Hayashi & Matsuzawa (2003) Animal Cognition

Questions raised

- Why did she specifically pay attention to the blocks (attention)?
- What mechanism could have allowed Pal to learn that she could stack the blocks (detect the affordances of blocks)?
- Did she understand causal relationships (e.g. that hitting the blocks would make them fall)?
- Would she be able to stack other shapes or different objects (re-combinable competences)?
- How did this behaviour develop?

What kinds of mechanisms make it possible for animals to find out about affordances, attribute causes to effects and generate appropriate (sometimes novel) behaviour?

What mechanisms do we know of?

- Developmentally-fixed behaviour - usually genetically determined
  - Fast and reliable, but inflexible
- Associative learning
  - Gradual process, but fairly flexible and surprisingly subtle
- Social learning
  - Can provide a short-cut to learning a novel behaviour
- Some extended learning mechanism—some ‘core knowledge’, new competences acquired, extended and re-combined through exploration and play?

Developmentally-fixed behaviour


© USGS.gov


© Wildlife Film & Foto

- Complex behaviour triggered by simple cues
- Useful when:
  - Limited opportunity for learning
  - Behaviour needs to be perfect on the first attempt (e.g. flight in cliff or tree-nesting birds)
  - There are time constraints (e.g. short life span)
- Common in precocial species where young are relatively independent from birth

Associative learning

- Classical conditioning and operant conditioning
- Can lead to a complex chain of behaviour → novel responses to the environment
- Relatively slow and gradual process (though one-trial learning is possible)

Social learning

- Learn from the behaviour of others:
  - Directly, by observation
  - Or via products of another’s behaviour
- Can spread novel behaviour rapidly through a population → cultural transmission → cultural evolution

Extended learning mechanism and exploration

- Animals can learn about the space of possible actions with an object, unusual properties etc.
- Time consuming, but possible for altricial species during development, when parent(s) care for offspring
- May also enable very rapid learning if ‘chunks’ of knowledge about the environment can be reused
- Exploration (not directly reinforced) may be very important

What do you pay attention to?

- Some genetically-determined biases which limit the stimuli that form associations (e.g. taste conditioning in rats)
- Exploration → classification of some things as ‘interesting’?

“Appropriateness” of the stimulus or response matters (Domjan & Wilson, 1972)

	Group taste	Group noise
Train	Sweet water → illness	Noisy water → illness
Test	Sweet water vs. Plain water	Noisy water vs. Silent water
RESULT	LEARNING	NO LEARNING
Train	Sweet water → shock	Noisy water → shock
Test	Sweet water vs. Plain water	Noisy water vs. Silent water
RESULT	NO LEARNING	LEARNING

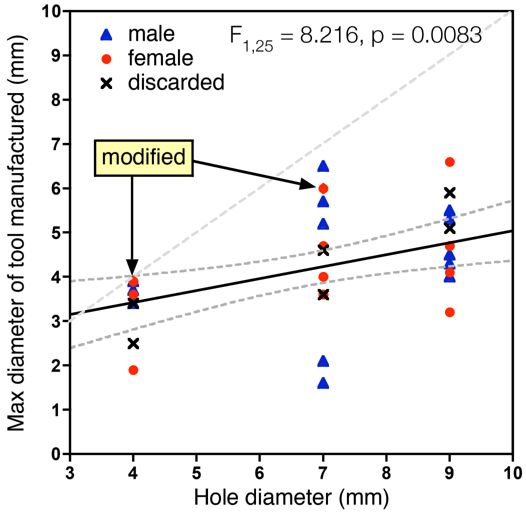
So, natural selection constrains associations to those likely to be causally linked

---

How to detect affordances?

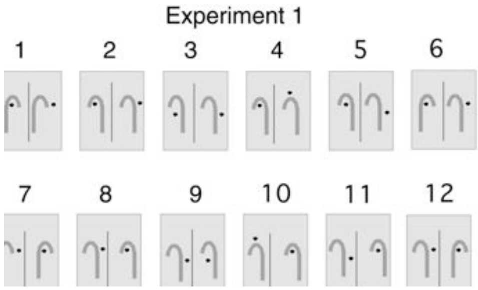
- Are affordances tied to specific stimuli, or can animals abstract more general properties?
- What is the role of experience?
- Is this an adaptation specific to the tool-using domain?

Making an appropriate tool for a novel task (New Caledonian crows)



(Chappell & Kacelnik 2004)

What do non-tool users understand about the function of tools?



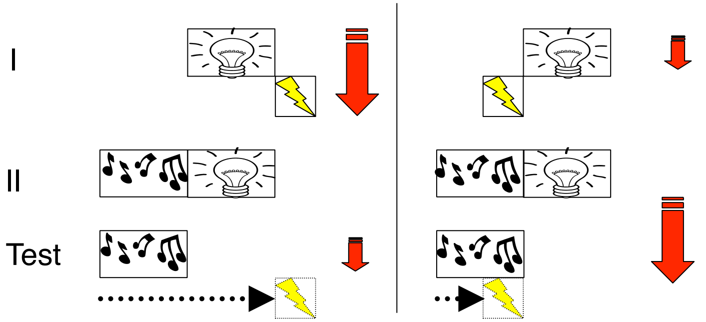
(Santos et al. 2005)

How to assign causality?

- Probabilistically, through contingency and contiguity (Rescorla & Wagner 1972)
- Test hypotheses by performing interventions (Gopnik & Schultz 2004)
- Core knowledge about the structure of the world (acquired or developmentally fixed) → expectations about causal structure (not all causes are equally possible) (Carey & Spelke 1996)

Animals can learn about the temporal relationship between events → causal attribution

(Barnet, Cole & Miller, 1997)



What causes objects to fall?

Possibly gaining dynamic feedback from environment, and adjusting behaviour appropriately

Re-combinable competences

- To what degree can animals re-combine existing competences to generate novel behaviour?
- How does this depend on previous experience?

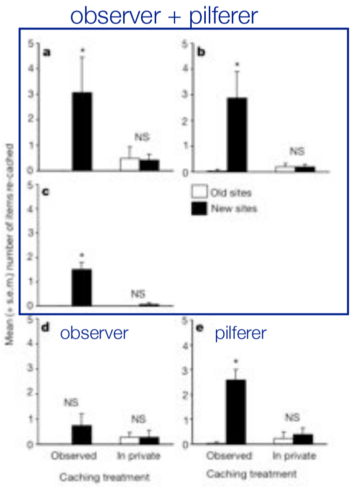
Pilfering in scrub jays: it helps to have been a thief to catch a thief

- Three groups:
  - Observer + Pilferer—had experience of both observing conspecifics caching, and of pilfering others caches
  - Observer—only experience with observing caching
  - Pilferer—listened to others caching, then allowed to pilfer caches

(Emery and Clayton 2001)

Experimental protocol

- Birds allowed to cache food in a tray:
  - With an observer bird watching from an adjoining cage (‘observed’ trial)
  - With no bird watching them (‘in private’ trial)
- Then allowed to retrieve cache and also given opportunity to re-cache in old tray or a new one



- Pilferers re-cached food when observed caching (in new sites)
- Specific to the tray which was observed, not a general increase in re-caching
- Observation of caching not sufficient to prompt re-caching

(Emery and Clayton 2001)

Novel manufacturing behaviour with a new material

- In an experiment on choice between a hooked wire and a straight one, Betty bent the hook spontaneously on the 5th trial
- In a subsequent experiment, she bent the hook and used it to remove the bucket on 9/10 trials

(Weir, Chappell & Kacelnik 2002)

[Weir, Chappell & Kacelnik 2002]

What might the mechanism allowing re-combination of competences be?

- Built-in drive to explore (with no immediate reinforcement consequences)
- Cognitive structures (genetically determined) which might guide or constrain exploration (‘bootstrapping’ of behaviour)
- Construction of reusable ‘chunks’ which can themselves be recombined into more complex structures (e.g. language learning)

How do these abilities develop?

- Exploration and play
  - Lack of neophobia—you can’t discover properties of objects you never go near
- Altricial species often have a large amount brain development going on after birth/hatching
  - Is it important that the developing brain is exposed to the environment?
- To what degree are animals limited by their exploratory tendencies?

Are animals limited by species-specific representational capacities, or by the their exploratory tendencies?

- Representational view vs. Ecological view (Cummins-Sebree and Fragaszy, 2005)
- Capuchin monkeys spontaneously re-positioned canes to pull a food reward towards them, unlike tamarins
- Is this difference because of species differences in exploratory/manipulatory behaviour?

Summary

- We need to combine the richness of animals' behaviour with the depth of knowledge of the mechanisms involved in artificial systems to explore this
- There is almost certainly more than one solution to the problem (*in vivo* and *in silico*)—the optimal solution depends on the 'habitat' of the agent
- Animals (and robots) need to be tested in ethologically valid ways to reveal their competences fully
- It's a very difficult (but interesting) problem!

A Novel Computing Architecture for Cognitive Systems based on the Laminar Microcircuitry of the Neocortex – the COLAMN project

Michael Denham
Centre for Theoretical and Computational Neuroscience
University of Plymouth, UK
mdenham@plym.ac.uk

Abstract

Understanding the neocortical neural architecture and circuitry in the brain that subserves our perceptual and cognitive abilities will be an important component of a "Grand Challenge" which aims at an understanding of the architecture of mind and brain. We have recently embarked on a new five-year collaborative research programme, the primary aim of which is to build a computational model of minimal complexity that captures the fundamental information processing properties of the laminar microcircuitry of the primary visual area of neocortex. Specifically the properties we aim to capture are those of self-organisation, adaptation, and plasticity, which would enable the model to: (i). develop feature selective neuronal properties and cortical preference maps in response to a combination of intrinsic, spontaneously-generated activity and complex naturalistic external stimuli; and (ii) display experience-dependent and adaptation-induced plasticity, which optimally modifies the feature selectivity properties and preference maps in response to naturalistic stimuli. The second aim of the research programme is to investigate the feasibility of designing VLSI circuitry which would be capable of realising the computational model, and thus demonstrate that the model can form the basis for a novel computational architecture with the same properties of self-organisation, adaptation, and plasticity as those displayed by the biological system. A basic premise of the research programme is that the neocortex is organised in a fairly stereotyped and modular form, and that in this form it subserves a wide range of perceptual and cognitive tasks. In principle, this will allow the novel computational architecture also to have wide application in the area of cognitive systems.

.

1 Introduction

The neocortex of the brain subserves sensory perception, attention, memory and a spectrum of other perceptual and cognitive functions, which combine to provide the biological system with its outstanding powers. It is clear that the brain carries out information processing in a fundamentally different way to today's conventional computers. The computational architecture of the brain involves the use of highly parallel, asynchronous, nonlinear and adaptive dynamical systems, namely the laminar microcircuits of the neocortex. The neurons which make up a neocortical microcircuit (Silberberg et al, 2002; Mountcastle, 1997) are precisely connected to each other and to their afferent inputs through synapses in specific layers of the laminar cortical architecture, and on specific locations on their dendritic trees Thomson and Bannister 2003; Callaway, 1998). Each synapse acts as a unique adaptive filter for the transmission of data into the circuit and between pairs of cells. Thus whilst a single neuron may connect to many hundreds of other neurons, a signal sent by one neuron will be interpreted by each target neuron in a unique way. Furthermore, these connections are not static but change their transmission characteristics dynamically and asynchronously, on a millisecond timescale, partly determined by their highly precise spatial location in the dendritic tree (Häusser et al, 2003) but also in relation to the function of the different neuronal types that they connect. In addition, both the synaptic connections and the transmission properties of the dendritic tree have the remarkable ability to continuously adapt and optimise themselves to meet the requirements of novel tasks and environments. This takes place both through unsupervised, self-organising modification of their dynamic parameters, and through optimisation of the synaptic and dendritic dynamics by specific adaptation-induced and experience-dependent plasticity mechanisms.

Capturing the fundamental information processing properties of the laminar microcircuitry of the neocortex in the form of a computer model could provide the foundation for a radical new generation of machines that have human-like performance in perceptual and cognitive tasks. Such machines would be capable of using self-organisation, adaptation, and plasticity mechanisms which are inherent in the neocortex, in order to deal with complex, uncertain and dynamically changing information. They would potentially be much more powerful, require minimal programming intervention, and be resilient to failures and errors. Creating the necessary understanding of these properties of the neocortex, expressing them as a computational model of minimal complexity, and translating this model into the design of a computer architecture capable of realisation in VLSI, will require the collaborative efforts of neuroscientists, computer scientists, mathematicians, and engineers.

2 The aims of the research programme

The aim of this research programme is to create a new "brain-inspired" computational architecture which possesses the basic properties of self-organisation, adaptation and plasticity manifest in the laminar neural microcircuitry of the neocortex. The principal objective is a functional model of a "stereotypical" cortical microcircuit which captures these basic properties of the neocortex, and provides the basis for the design of a novel, modular computational architecture capable of realisation in a combination of analogue and digital VLSI circuits. The ultimate goal of this avenue of research is a "brain-inspired" architecture which will deliver human-like levels of performance for a wide range of perceptual and cognitive tasks, and deal with all sensory modalities.

This goal is well beyond the scope of the currently envisaged research programme; however, as a first step towards this goal, the programme will aim at capturing the fundamental properties of self-organisation, adaptation and plasticity of the neuronal circuitry in the primary visual area of the mammalian neocortex. This will allow us to build on the wealth of current neurobiological knowledge concerning the properties and interconnectivity of neurons and the behaviour of local and long-range neuronal circuitry in this area of neocortex in response to visual stimuli. It must be stressed that our aim is not to build a detailed, biologically-precise model of neocortex, but rather it is to identify and capture in a minimally complex model these key fundamental properties that underlie its remarkable information processing capabilities.

The specific aims of the proposed research programme can therefore be summarised as follows:

1. To build a computational model of minimal complexity that captures the fundamental information processing properties of the laminar microcircuitry of the primary visual area of neocortex. Specifically the properties we aim to capture are those of self-organisation, adaptation, and plasticity, which would enable the model to:
    i. develop feature selective neuronal properties and cortical preference maps in response to a combination of intrinsic, spontaneously-generated activity and complex naturalistic external stimuli, and
    ii. display experience-dependent and adaptation-induced plasticity, which optimally modifies the feature selectivity properties and preference maps in response to naturalistic stimuli.
2. To investigate the feasibility of designing VLSI circuitry which would be capable of realising the computational model, and thus demonstrate that the model can form the basis for a novel computational architecture with the same properties of self-organisation, adaptation, and plasticity.

3 The research programme

The research programme involves a high level of integration of activities in neurobiological modelling, experimental neurobiology and the VLSI circuit design. It is organised into a set of such activities, each of which addresses a well-defined aim of the research programme, as described below.

3.1 Novel neocortical neuron and circuit connectivity models

A basic premise of the research programme is that the neocortex is organised in a fairly stereotyped and highly modular fashion. Although much is already known about the structure and functional connectivity of microcircuits in the neocortex, the current state of knowledge is only sufficient to inform the initial design and construction of the proposed computational model. Recent work eg Thomson and Bannister (2003), has contributed important and detailed insights into the synaptic connectivity and the dynamic and plastic aspects of information transmission along these synaptic connections within a cortical column. The research will draw on this and on further, on-going work in order to more fully elucidate the neuronal and synaptic connectivity which it is necessary to capture within the computational model in order to endow it with the self-organisation, adaptation and plasticity properties of cortical microcircuits.

In particular, the behaviour of circuit models of spiking neurons strongly depends on the properties of their constituents, the individual neurons, as well as on the synaptic connectivity between them. For example, the phase diagrams describing the dynamics of sparsely connected networks of excitatory and inhibitory neurons, which can exhibit different synchronous and asynchronous states (Brunel, 2000) change fundamentally when current-based integrate-and-fire neurons are replaced by conductance-based integrate-and-fire neurons as the constituents of the network. In order to provide the cortical modelling and the VLSI designs with the best possible description of single neurons (in terms of both accuracy and computational efficiency), the plan is to construct new types of integrate-and-fire neuron models that represent the biophysical mechanisms operating in biological neurons in a more realistic way, including the role of neuronal dendrites in the transformation of synaptic input into spike output.

Models will be validated by direct comparison with experimental data from experiments in brain slices and in the intact animal in vivo which describe the input-output relation of different types of neurons both at a functional, eg Chadderton et al., 2004, and a biophysical level, eg Häusser et al., 2001. We will focus on those characteristics of real neurons that are currently not, or only with insufficient accuracy, captured by the integrate-and-fire or spike response models currently available. We expect that models including the subthreshold dynamics of voltage-dependent conductances, including oscillatory behaviour, as well as the shunt conductances associated with action potential firing, which provide only a partial reset of the membrane potential in the neuron, will lead to more realistic yet compact descriptions of the input-output relations of different types of cortical neurons.

Single-neuron models will be complemented by three-dimensional geometric models of synaptic connectivity based on anatomical and physiological data from a large dataset of anatomically and physiologically identified, synaptically connected neurons which are being generated in a number of laboratories. Together these will provide an intra- and interlaminar wiring diagram of the cortical microcircuit. The functional properties of the synaptic connections between different types of neurons will be described by statistical distributions of the amplitudes and time courses of the synaptic conductances, including a representation of short- and long-term synaptic plasticity.

3.2 Functional analysis and modelling of the neocortical microcircuit

It will be essential to provide constraints for the proposed computational model. This is a non-trivial but essential task if we are to ensure that the modelling work does not result in “parameter explosion”. In particular, it will be necessary to constrain the model on the basis of the functional properties of the neurons and their interconnectivity in the cortical microcircuit. *In vivo*, cortical cells receive input from several thousand synaptic connections simultaneously, and only a proportion of these are connections from other cells within the cortical microcircuit. Some aspects of the intra-columnar connectivity revealed by intracellular recordings will form an essential part of the function of the cortical column, while other aspects are unimportant details that are best ignored in the proposed computational model. Constraining the model therefore means deciding which aspects are important, and estimating the strength of their contribution relative to other, external inputs and influences. This will require combining new extracellular recording techniques and novel statistical analysis and modelling approaches. Silicon array electrode techniques make it possible to record spiking activity simultaneously from dozens of neurons throughout a cortical microcircuit, in the living brain while it is carrying out its natural information processing tasks. In the past, simple filter models have been used to predict responses of individual neurons in sensory cortex (Schnupp et al, 2001). The research programme will aim at a dramatic improvement in these simplistic models through the use of novel statistical modelling techniques.

3.3 Learning rules for the development of stable self-organised feature selectivity

The role of self-organisation in the stimulus-dependent development of orientation selectivity was first suggested by von der Malsburg and recently reviewed by Miller et al (1999) and Sur and Leamey, 2001. The latter suggest that spontaneous patterns of neural activity in the absence of visual stimuli may be sufficient in the early periods of development, after the initial cortical circuitry has been established, for the early development of orientation selectivity, but that the formation of orientation selectivity is strongly influenced by input activity to the developing cortex (Sur and Leamey, 2001; Sur et al, 1988). Experiments show that input activity has an influence on synaptic connections in the cortical circuitry which gives rise to orientation map development and long-range horizontal intracortical connections in layers 2/3. A cortical microcircuit model would thus need to embody development of the dynamical interactions provided by intracortical connections in an activity-instructed self-organising process of map development. As yet, it would appear that no biological models exist which implement this activity-dependent self-organising process of development.

It has been demonstrated experimentally that the self-organised modification of synapses depends on the precise timing of spikes, causing the neuron to evolve in such a way as to be driven by its fastest and most reliable inputs. Therefore it seems reasonable to hypothesise that the learning rules which govern the self-organised emergence of cortical orientation selectivity should yield populations of selective cells, large enough to perform fast and reliable computation, yet small enough to be efficient. The investigation of these issues will lead to an understanding of how learning rules can self-organise the synaptic interconnectivity in the cortical microcircuit to produce a stable, sparse coded orientation selective network. An important component of this work will be to investigate how the stability of feature selectivity might be helped by recurrent interactions between neurons. These connections could break the symmetry and stabilise the synaptic weights, improving the stability of feature selectivity. The precise details of the learning rules are expected to be of crucial importance for the final selectivity patterns learned. This holds for both rate based as for spike timing dependent learning rules (van Rossum et al, 2000).

3.4 Neural coding of feature selectivity properties of cortical circuits

Intimately related to the investigation of developmental self-organisation learning rules is the question of neural coding, i.e. of how neuronal populations represent sensory information. This is even more evident in the case of spike timing dependent learning rules. For instance, if stimulus features are coded across the cortical microcircuit by either precise spike times of individual neurons or by synchronous neuronal activity across neurons, it is of importance to know how such coding affects learning. Likewise, the developmental learning rules which result in specific patterns of synaptic connectivity have to support the selective neural coding of the stimulus feature set. A major objective of this part of the research programme will be to understand what advantages the laminar architecture of the neocortex offers in terms of efficiency of information representation.

By using mathematical analysis techniques based on the principles of information theory, the role of columnar organization in cortical information representation has recently been investigated (Panzeri et al, 2003), but it is clear that the laminar organisation can provide both advantages and constraints that are as important. It has been shown that real cortical neurons encode information by timing of individual spikes with millisecond precision (Panzeri et al, 2001a) and investigated what mechanisms are need to read out this information, eg dendritic processing must be important to decode information if most information is encoded by the “label” of which neuron fired each spike, and not very important if instead neurons can sum up all spikes at the soma and still conserve all information (Panzeri et al, 2003).

Research in this part of the programme will extend these ideas by investigating in detail the information processing capabilities of laminar cortical circuits. In particular, we will determine (i) the “neuronal code” used in different laminae, i.e. which of the features (e.g. spike count, precise spike times, synchronization) characterizing the responses of neuronal populations in different laminae convey the most sensory information (ii) whether the precise synaptic connectivity within the laminar neocortical architecture is to some extent “optimal” for fast information transmission from one neuron/layer to another neuron/layer. By “optimal” we mean that the observed wiring comes close enough in terms of transmitted information to the best possible one. By fast we mean that all of this information must be transmitted by the model synaptic system in time scales as fast as the cortical ones (Panzeri et al, 2001b).

3.5 Learning rules for experience-dependent and adaptation-induced plasticity in the developed cortical microcircuit

It is well known that the ability to detect small orientational differences can be significantly improved through training on a visual discrimination task over an extended period of time. This perceptual learning process is also seen to have a long-lasting effect, indicating that it must be the result of some form of long-term synaptic plasticity in the brain. Other characteristics of the learning, which can be psychophysically observed, such as the lack of transfer of the learning from one orientation to the orthogonal orientation or from one learned retinal location to a nearby nonoverlapping location, indicate that the plasticity must involve the primary visual cortex, where the neurons have localised orientation selectivity and small receptive fields. Orientation plasticity has also been demonstrated in response to continuous visual stimulation for a period of seconds to minutes, a process known as adaptation. The results suggest that adaptation-induced orientation plasticity involves changes in circuit connectivity which then define a new preferred orientation. As proposed in the review by Dragoi and Sur (2003), the changes in orientation selectivity following adaptation imply a circuit mechanism that reorganizes responses across a broad range of orientations, and suggest that adaptation-induced orientation plasticity in primary visual cortex is a self-organised emergent property of a local cortical circuitry acting within a non-uniform orientation map. Research in this part of the programme will investigate the learning rules necessary to support the proposed emergence of adaptation-induced modification of orientation selectivity, and whether such learning rules can also support long-term experience-dependent plasticity of orientation selectivity.

3.6 Novel neocortical neuron and circuit connectivity models

Spatiotemporal response properties of neurons in fully developed primary sensory areas are not static but can change on various timescales. Dynamic changes of response properties on long timescales have been assigned to adaptation and plasticity mechanisms. But responses also change on fast time-scales of a few to a few hundred milliseconds revealing rich dynamic features that result from the neural and synaptic activation dynamics and ongoing interactions between neurons within and across cortical microcircuits eg Bringuir et al (1999). Recent experiments eg Ringach et al (2002) indicate even more complex responses of cortical neurons and circuits to naturalistic stimuli. Spatiotemporal responses look similar to those for simple bar or grating stimuli, but there are also significant differences (Ringach et al, 2002). In part these differences seem to be related to influences from outside the classical receptive field: These experiments provide evidence that spatiotemporal response properties of cortical neurons are dynamically shaped in quite intricate ways by intrinsic neuronal and synaptic activation dynamics, interactions between neurons within the microcircuit, and longer ranging synaptic recurrent, feedforward and feedback circuits. These dynamical properties may underlie the surprisingly fast and adaptable information processing within the cortical microcircuit.

A general analytical approach has recently been described (Wennekers, 2002) that relates differently tuned enhanced and suppressed phases in a spatiotemporal response function to feedforward or recurrent pathways between participating cell classes. Although useful for some spatiotemporal phenomena, much of the complexity in real neural responses remains unexplained by such models. Models of complex spatiotemporal phenomena which incorporate the influence of ongoing and spreading activity, or responses to real-world stimuli, are still scarce.

3.7 Feasibility analysis for VLSI circuit design

A major aim of the research programme is to use the computational model of the neocortical laminar microcircuit to define an efficient and implementable VLSI “building block” for a novel computational architecture. The mapping of the model of the cortical microcircuit model into the VLSI circuit design for a novel computational architecture will require the investigation of detailed issues with respect to the numerical accuracy, performance, power consumption and area cost of novel analogue and digital circuit alternatives. These investigations will form the activity of this workpackage. We envisage a structure for the VLSI design based upon an analogue VLSI spiking neural substrate, interconnected via a digital VLSI address-event communication network, all controlled by a software configuration and control system. However, the integration of low-level neural models implemented by analogue VLSI circuits, with digital VLSI for signal routing and communication will need to go far beyond the simple protocols currently used by the neuromorphic engineers, and presents a major challenge. The research will also aim at understanding the implications, in both directions, for including or omitting certain components in the computational model, and assessing the relationship between the levels of description chosen for the computational model and the constraints of VLSI circuit design. In addition, the issues of optimisation (area, power) are present both in the neocortical microcircuit, and in silicon, so some direct analogies on a physical level will be investigated, eg the possible arrangement of the physical layout of devices in a way which is inspired by the 3-dimensional laminar architecture of the cortical microcircuit, in which connections between the cortical microcircuit “building blocks” are predominantly within or between certain layers.

4 Summary

The neocortex of the brain subserves sensory perception, attention, memory and a spectrum of other perceptual and cognitive functions, which combine to provide the biological system with its outstanding powers. It is clear that the brain carries out information processing in a fundamentally different way to today's conventional computers. The computational architecture of the brain clearly involves the use of highly parallel, asynchronous, nonlinear and adaptive dynamical systems, namely the laminar neural microcircuits of the neocortex. The fundamental aim of this research programme is to create a new brain-inspired" computing architecture which possesses the basic properties of self-organisation, adaptation and plasticity manifest in the neural circuitry of the neocortex. The objective is a modular architecture based on a representation of a "stereotypical" cortical microcircuit. The research will focus on the laminar microcircuits of the primary visual cortex in order to build on the wealth of neurobiological knowledge concerning the behaviour and interconnectivity of neurons in this area of neocortex. However the wider objective would be to use the laminar microcircuitry of primary visual cortex as an exemplar for a stereotypical neocortically-inspired architecture. This will allow the architecture to be deployed in a wide range of perceptual tasks, and potentially also in cognitive tasks such as decision making,, with minimal changes to the basic circuitry. The aim is not simply to build a detailed, biologically-precise model of primary visual cortex, but rather the challenge is to identify and capture the key fundamental principles and mechanisms that underlie the remarkable and ubiquitous information processing power of the neocortex.

Acknowledgements

The collaborators in this research programme are: Alex Thomson (School of Pharmacy, London); Michael Häusser and Arnd Roth (UCL); Jan Schnupp (Oxford); Mark van Rossum and David Willshaw (Edinburgh); Stefano Panzeri, Piotr Dudek and Steve Furber (Manchester); Thomas Wennekers and Susan Denham (Plymouth). I would like to fully and gratefully acknowledge their individual contributions to the description of the aims, objectives and research activities which I have set out on their behalf in this paper. Any mistakes herein are however my own. The COLAMN research programme is supported by a grant from the UK Engineering and Physical Research Council under the Novel Computation Initiative.

References

Silberberg G, Gupta A, Markram H (2002) *Trends Neurosci* 25, 227-230

Mountcastle V (1997) *Brain* 120:701-722

Thomson AM, Bannister AP (2003) *Cereb Cortex* 13, 5-14

Callaway EM.(1998) *Annu Rev Neurosci* 21:47-74

Häusser M, Spruston N, Stuart GJ (2000) *Science*, 290:739-744;

Brunel N (2000) *Network* 11:261-280

Chadderton P, Margrie TW, Häusser M (2004) *Nature* 428:856-860

Hausser M, Major G, Stuart GJ (2001) S*cience.* 291:138-41.

Schnupp JW, Mrsic-Flogel TD, King AJ (2001) *Nature* 414:200-204

Miller KD, Erwin E, Kayser A. (1999) *J Neurobiol.* 41:44-57

Sur M, Leamey CA (2001) *Nature Reviews Neurosci.* 2:251-262

Sur M, Garraghty PE, Roe AW (1988) *Science* 242:1437-41

van Rossum M, Bi GQ, Turrigiano GG (2000), *J. Neurosci.* 20, 8812-8821

Panzeri S, Petroni F, Petersen RS, Diamond ME (2003) *Cerebral Cortex* 13: 45-52

Panzeri S, Petersen R, Schultz SR, Lebedev M, Diamond ME (2001a) *Neuron* 29: 769-777

Panzeri S, Rolls ET, Battaglia F, Lavis R (2001b) *Network* 12: 423- 440

Dragoi V, Sur M (2003) In Eds. Chalupa LM and Werner JS, *The Visual Neurosciences,* MIT Press.

Bringuir V, Chavane F, Glaeser L, Fregnac Y (1999) *Science* 283, 695-699

Ringach DL, Hawken M, Shapley R (2002) *J Vision* 2, 12-14

Wennekers T. (2002) *Neural Computation* 14: 1801-25

High-Performance Computing for Systems of Spiking Neurons

Steve Furber and Steve Temple[*]
[*]The University of Manchester
School of Computer Science
Oxford Road, Manchester M13 9PL, UK
steve.furber,steven.temple@manchester.ac.uk

Andrew Brown[†]
[†]The University of Southampton
Department of Electronics and Computer Science
Southampton, Hampshire, SO17 1BJ, UK
adb@ecs.soton.ac.uk

Abstract

We propose a bottom-up computer engineering approach to the Grand Challenge of understanding the Architecture of Brain and Mind as a viable complement to top-down modelling and alternative approaches informed by the skills and philosophies of other disciplines. Our approach starts from the observation that brains are built from spiking neurons and then progresses by looking for a systematic way to deploy spiking neurons as components from which useful information processing functions can be constructed, at all stages being informed (but not constrained) by the neural structures and microarchitectures observed by neuroscientists as playing a role in biological systems. In order to explore the behaviours of large-scale complex systems of spiking neuron components we require high-performance computing equipment, and we propose the construction of a machine specifically for this task – a massively parallel computer designed to be a universal spiking neural network simulation engine.

1 Introduction

1.1 Neurons

The basic biological control component is the neuron. A full understanding of the 'Architecture of Brain and Mind' (Sloman, 2004) must, ultimately, involve finding an explanation of the phenomenological observations that can be expressed in terms of the interactions between the neurons that comprise the brain (together with their sensory inputs, actuator outputs, and related biological processes).

Neurons appear to be very flexible components whose utility scales over systems covering a vast range of complexities. Very simple creatures find a small number of neurons useful, honey bees find it economic to support brains comprising around 850,000 neurons, and humans have evolved to carry brains comprising 10^{11} neurons or so. The component neuron used this range of complexities is basically the same in its principles of operation, so in some sense it has a universality similar to that enjoyed by the basic logic gate in digital engineering.

There is a further similarity between neurons and logic gates: both are multiple-input single-output components. However, while the typical fan-in (the number of inputs to a component) and fan-out (the number of other components the output of a particular component connects to) of a logic gate is in the range 2 to 4, neurons typically have a fan-in and fan-out in the range 1,000 to 10,000. (It is easy to show that that mean fan-in and fan-out in a system are the same – they are just different ways of counting the number of connections between components.)

A more subtle difference between a logic gate and a neuron is in the dynamics of their internal processes. Whereas a logic gate implements a process that is essentially static and defined by Boolean logic, so that at any time from a short time after the last input change the output is a well-defined stable function of the inputs, a neuron has complex dynamics that includes several time constants, and its output is a time series of action potentials or 'spikes'. The information conveyed by the neuron's output is encoded in the timing of the spikes in a way that is not yet fully understood, although rate codes, population codes and firing-order codes all seem offer valid interpretations of certain observations of spiking acitivity.

Accurate computer models of biological neurons exist, but these are very complex. Various simpler models have been proposed that capture some of the features of the biology but omit others. The difficulty lies in determining which of the features are essential to the information processing functions of the neuron and which are artefacts resulting from the way the cell developed, its need to sustain itself, and the complex evolutionary processes that led to its current form.

1.2 Neural microarchitecture

The universality of the neuron as a component is also reflected in certain higher-level structures of the brain. For example, the cortex displays a 6-layer structure and a regularity of interconnect between the neurons in the various layers that can reasonably deserve the application of the term ‘microarchitecture’. The same regular laminar cortical microarchitecture is in evidence across the cortex in regions implementing low-level vision processes such as edge-detection and in regions involved in high-level functions such as speech and language processing. This apparent ‘universality’ (used here to describe one structure that can perform any function) of the cortical microarchitecture suggests there are principles being applied here the understanding of which could offer a breakthrough in our understanding of brain function.

In contrast to the regularity and uniformity of the microarchitecture, the particular connectivity patterns that underpin these structures appear to be random, guided by statistical principles rather than specific connectivity plans. The connectivity is also locally adaptive, so the system can be refined through tuning to improve its performance.

1.3 Engineering with neurons

As computer engineers we find the neuron’s universality across wide ranges of biological complexity to be intriguing, and there is a real challenge in understanding how this component can be used to build useful information processing systems. There is an existence proof that this is indeed possible, but few pointers to how the resulting systems might work.

There are other ‘engineering’ aspects of biological neurons that are interesting, too. We have already mentioned the regularity of neural microarchitecture. The power-efficiency of neurons (measured as the energy required to perform a given computation) exceeds that of computer technology, possibly because the neuron itself is a very low performance component. While computer engineers measure gate speeds in picoseconds, neurons have time constants of a millisecond or longer. While computer engineers worry about speed-of-light limitations and the number of clock cycles it takes to get a signal across a chip, neurons communicate at a few metres per second. This very relaxed performance at the technology level is, of course, compensated by the very high levels of parallelism and connectivity of the biological system. Finally, neural systems display levels of fault-tolerance and adaptive learning that artificial systems have yet to approach.

We have therefore decided to take up the challenge to find ways to build useful systems based upon spiking neuron components (for example, Furber, Bainbridge, Cumpstey and Temple, 2004), and we hope that this will lead to mutually-stimulating interactions with people from many other disciplines whose approach to the same Grand Challenge, of understanding the Architecture of Brain and Mind, will be quite different from our own.

2 Relevance to GC5

What has any of this engineering really got to do with the Grand Challenge of understanding the Architecture of Brain and Mind?

As this is aimed at a broad audience, not many of whom are computer engineers, we will digress briefly to consider what computer engineers may bring to this Grand Challenge. To begin with, it is useful to appreciate the skills and mindset that a computer engineer, for better or for worse, possesses. What can a person whose stock-in-trade consists of logic gates, microchips and printed circuit boards contribute to the bio-psycho-philosophical quest to understand the workings of the mind?

2.1 A Computer Engineer’s manifesto

To a computer engineer ‘understand’ has a specific meaning that is different from what a scientist means by the same word, which is in turn probably different from the meanings used by other disciplines. To the scientist, understanding is to have a repeatably-verifiable explanation of a phenomenon. To the engineer, understanding means to be able to go away and build another artefact that works in the same way. The scientist’s analysis reduces a complex phenomenon into its basic components; this is complemented by the engineer’s ability to take those components, or components that encapsulate the same essential behaviour, and synthesize them back into a functioning system.

Thus, when a computer engineer claims to ‘understands’ how a mobile phone works, the statement can be interpreted as meaning that they can (at least in principle) explain when every one of the 100 million or so transistors switches, why it switches, what will happen if it fails to switch, and so on. OK, we might get on less secure ground when describing the chemistry of the lithium-ion battery and the details of the radio and antenna design or the higher levels of the software. And when it comes to explaining why the plastic case is pink and the buttons are arranged in swirling patterns with no obvious ergonomic objective we are completely lost! But back in the familiar territory of the digital transistor circuits we have a vocabulary comprising baseband processors, DSPs, maximum likelihood error correctors, RAMs, buses, interrupts, and so on, that together provide a language of description at multiple levels of abstraction from an individual transistor to the lower levels of the system software. This enables us to describe in very fine detail how the phone works and, more particularly, how you might make another working phone at lower cost and with better battery life.

This is the approach we bring to understanding the Architecture of Brain and Mind. In neuroscience we see that there are pretty accurate models of the basic component from which brains are built – the neuron. There are some rather sketchy and limited descriptions of how these components are interconnected and how they behave in natural networks, and there is rather better information about their macro-level modular organisation and gross activity. The weakest part of the neuroscientists' analysis (for very good reason – it is hard to apply reductionist principles to systems whose interesting characteristics depend on their organizational complexity) is at the intermediate levels between the component neurons (where analysis is applicable) and the macro-organisation (where mean field statistics work).

This intermediate level is precisely the level at which the computer engineer may have something to offer. Assembling basic components into functional units, implementing useful computational processes based on networks of dynamical systems, these are all second nature to the computer engineer once we have come to grips with the spiking neuron as a component. As we observed earlier, it even looks a bit like a logic gate – several inputs but only one output.

The intrinsic dynamics of a neuron may confound the computer engineer who is used to working only with digital circuits that are controlled by the extrinsic straitjacket of a clock signal, but a small minority of us are proficient in building circuits whose sequential behaviour is intrinsic – members of the class of digital circuit generally described as asynchronous or self-timed. The knowledge we hold on how to build reliable, highly complex asynchronous digital systems *may just* provide us with new insights into the highly complex asynchronous neural systems that provide the hardware platform upon which the brain and mind are built.

2.2 GC5 methodology

Our approach to this Grand Challenge is essentially bottom-up, which will complement the top-down and middle-out approaches that are better-suited to those who bring different skills and mindsets from other disciplines.

The bottom-up approach starts from the concept of a neuron as a basic component, and then seeks useful compositions of neurons to create (and implement) increasingly higher levels of functional abstraction. These compositions may be inspired by neuroscience; for example, we have an involvement in the EPSRC-funded COLAMN project which has as its goal the creation of novel computational architectures based on the laminar microarchitecture of the neocortex, with considerable input from the 'wet' neuroscientists in the project. Or they may be designed in the abstract; for example our earlier work on *N*-of-*M* coded sparse distributed memories (Furber, Bainbridge, Cumpstey and Temple, 2004) – with at best tenuous relevance to biology.

A feature of this research is that it can yield a positive outcome in two distinct ways. It may contribute to the scientific objective of understanding the architecture of brain and mind, and/or it may contribute to the engineering objective of delivering better/different/novel models of computation. Either of these outcomes would justify our engagement, and with a following wind we might just achieve both...

In order to pursue this research agenda we need a sandpit in which we can experiment with neuron components on a large scale, hence the massively parallel high-performance computer theme that we will turn to shortly. This large-scale engineering project brings with it additional research aspects relating to fault-tolerance, autonomic computing, self-healing, networks-on-chip, and so forth, all of which add to the engineering challenge but probably contribute little to the GC5 agenda.

3 Objectives

We have set ourselves the objective of simulating a billion spiking neurons in real time while making as few assumptions as possible about what a neuron is and how the neurons are connected. We approach this by viewing a neural system as an event-driven dynamical system – a hybrid system where a (large) set of components, each of which operates in continuous time (and is characteristically described by a set of differential equations), interact through discrete events.

In order to retain complete flexibility in the internal neural dynamics we implement the real-time differential equation solvers (which will typically use discrete-time fixed-point approximations) in software, and then exploit the high speeds of electronic signalling to communicate the discrete inter-neuron communication events around the system in a time which is close to instantaneous on the time-scales of the neuron dynamics. This allows us to use a virtual mapping from the physical structure of the biological system we are modelling to the physical structure of the electronic system we are running the model on.

4 Neural computation

Any computation system must achieve a balance between its processing, storage and communication functions. It is useful to consider how these three functions are achieved in neural systems.

4.1 Processing

The neuron itself performs the processing function. It produces output events in response to input events through a non-linear transfer function, which we will model using suitable differential equations whose complexity is limited only by the available computing power.

The simplest neuron models process inputs by taking a linear sum of the inputs, each weighted by the strength of its respective synapse. When the inputs are spike events the multiplication implied by the weighting process reduces to repeated addition. Multiplication by repeated addition is usually inefficient, but here many inputs are likely to be inactive at any time and multiplication by zero by repeated addition is supremely efficient!

The weighted input sum is then used to drive the neural dynamics. A leaky-integrate-and-fire (LIF) model applies an exponential decay to the effect of each input, but if enough inputs fire close together in time to push the total activation past a threshold, the neuron fires its output and resets. More sophisticated models have more complex dynamics. For example, the models by Izhikevich (2004) are based on mathematical bifurcation and display a more diverse range of biologically-relevant behaviours that the LIF model.

4.2 Communication

Communication in neural systems is predominantly through the propagation of spike 'events' from one neuron to the next. The output from the neuron's body – its soma – passes along its axon which conveys the spike to its many target synapses. Each synapse use chemical processes to couple the spike to the input network – the dendritic tree – of another neuron.

Since the spike carries no information in its shape or size, the only information is which neuron fired and when it fired. In a real-time simulation the timing is implicit (and the communication, being effectively instantaneous, preserves the timing), so all we need to communicate is the identity of the neuron that fired, and we must send that to every neuron to which the firing neuron connects.

In the biological system the identity of a firing neuron is spatially encoded – each neuron has its own physical axon. In our system we cannot implement an equivalent level of physical connectivity so instead we use logical encoding by sending a packet identifying the firing neuron around a network that connects all of the components together.

4.3 Storage

It is in the storage of information that the neuron's story becomes most complex. There are many processes that can be seen as storing information, some operating over short time scales and some very long-term. For example:

- the neural dynamics include multiple time constants, each of which serves to preserve input information for some period of time;
- the dynamical state of the network may preserve information for some time;
- the axons carry spikes at low speeds and therefore act as delay lines, storing information as it propagates for up to 20ms;
- the coupling strength of a synapse is, in many cases, adaptive, with different time constants applying to different synapses.

The primary long-term storage mechanism is synaptic modification (within which we include the growth of new synapses).

In a real-time modelling system we expect the modelling to capture the neural and networks dynamics, and hence the contributions these mechnisms make to information storage. The axon delay-line storage does not come so easily as we have deliberately exploited the high speeds of electronic signalling to make spike communication effectively instantaneous in order to support a virtual mapping of the physical structures. It is likely that the axon delay is functionally important, so we must put these delays back in, either by delaying the issue of the spike or by delaying its effect at the destination. Either solution can be achieved in software, but both have drawbacks, and this remains one of the trickier aspects of the design to resolve to our complete satisfaction.

The final storage process is the most fundamental: synaptic weight adaptivity. Here we require long-term stability and support for a range of learning algorithms. We will exploit the fact that digital semiconductor memory is a mass-produced low-cost commodity, and the proposed machine is built around the use of commodity memory for storing synaptic connectivity information.

Indeed, as we shall see in the next section, the major resources in a neural computation system revolve around the synapses, not around the neural dynamics.

5 Computing requirements

Various estimates have been offered for the computational power required to run a real-time simulation of the human brain based on reasonably realistic neuron models. The answer generally comes out in the region of 10^{16} instructions per second, which is some way beyond the performance of a desktop PC or workstation, but not far beyond the performance of the petaFLOP supercomputers currently in design.

The route to this performance estimate can be summarized as follows: the brain comprises around 10^{11} neurons, each with of the order of 1,000 inputs. Each input fires at an average rate of 10 Hz, giving 10^{15} connections per second, and each connection requires perhaps 10 instructions.

Note that this estimate is based on the computing power required to handle the synaptic connections. Modelling the neuron dynamics is a smaller part of the problem: 10^{11} neurons each requiring a few 10s of instructions to update their dynamics perhaps 10^3 times a second, requiring in total an order of magnitude less computing power than the connections.

A similar calculation yields the memory requirements of such a simulation: 10^{14} synapses each require of the order of a few bytes, so around 10^{14} bytes of synaptic connection data are required.

At present the only way a machine of such capacity can be conceived is to employ a massively parallel architecture. This is likely to remain true even with future developments in CMOS technology as further increases in clock speed and individual processor throughput are unlikely to be great, as evidenced by the recent trend towards multi-core processors from all of the leading microprocessor vendors. The future of the microprocessor is in chip multiprocessors, and the future of high-performance computing is in massively parallel systems.

Fortunately, the problem of simulating very large numbers of neurons in real time falls into the class of 'embarrassingly' parallel applications, where the available concurrency allows the trade-off of processor performance against the number of processors to be totally flexible. The issue, then, is to determine how such a system might be optimised. What are the relevant metrics against which to make decisions on the systems architecture?

We propose that the primary metrics should be *performance density* (measured in MIPS/mm^2 of silicon) and *power-efficiency* (measured in MIPS/watt). The former is the primary determinant of the capital cost of the machine, while the latter influences both the capital cost – in terms of the cooling plant – and the running cost – a machine such as this demands a significant electrical power budget.

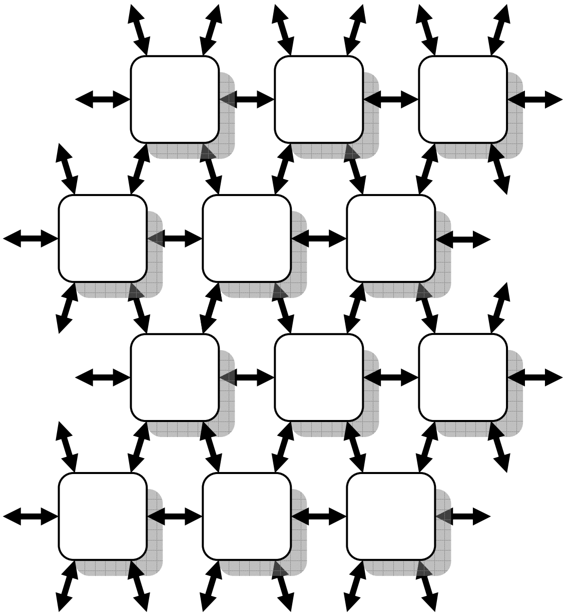


Figure 1: The system architecture.

A choice then has to be made between using a large number of high-performance processors or an even larger number of more power-efficient embedded processors. Here the metrics can be our guide – embedded processors win handsomely on power-efficiency, and to a lesser extent on performance density, over their much more complex high-end counterparts.

That, then sets the course for this work. The objective is to build a machine, based on large numbers of small processors, that has the potential to scale up to the levels of parallelism and performance necessary to model a brain in real time. Admittedly, modelling a complete human brain is some way beyond our current goals, but we should be able to model substantial parts of the human brain and complete brains of less complex species with what we propose here, which is a machine capable of modelling a billion spiking neurons in real time.

6 SpiNNaker

A spinnaker is a large foresail that enables a yacht to make rapid progress in a following wind (see reference to 'following wind' in Section 2.2 above!). We have adopted SpiNNaker as a name for our project because it comes close to a contraction of 'a (uni-

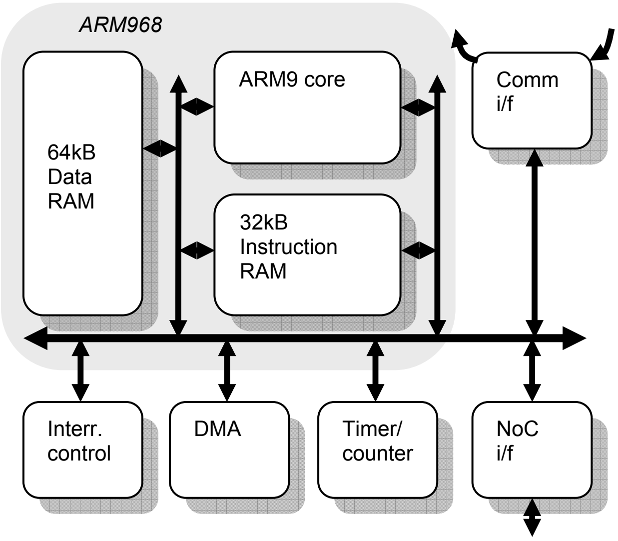


Figure 2: Processor subsystem organization.

versal) Spiking Neural Network architecture', provided you say it quickly enough. Again, this is our goal: to build a computer system that is as universal as we can make it in its ability to simulate large systems of spiking neurons, preferably in real time.

The following description of the system is largely extracted from Furber, Temple and Brown (2006).

6.1 System architecture

The system is implemented as a regular 2D array of nodes interconnected through bi-directional links in a triangular formation as illustrated in Fig. 1. The 2D mesh is very straightforward to implement on a circuit board and also provides many alternative routes between any pair of nodes which is useful for reconfiguration to isolate faults. (Nothing in the communications architecture precludes the use of a more complex topology if this proves advantageous.)

Each node in the network comprises two chips: a chip multiprocessor (CMP) and an SDRAM, with the integer processing power of a typical PC but at much lower power and in a compact physical form. The six bidirectional links support a total of 6 Gbit/s of bandwidth into and out of the node. A system of 100 x 100 nodes will deliver a total of 40 teraIPS, sufficient to simulate perhaps 200 million spiking neurons in real time, and will have a bisection bandwidth of 200 Gbit/s.

6.2 ARM968 processor subsystem

For the reasons already outlined, we choose to base the system around a massively-parallel array of

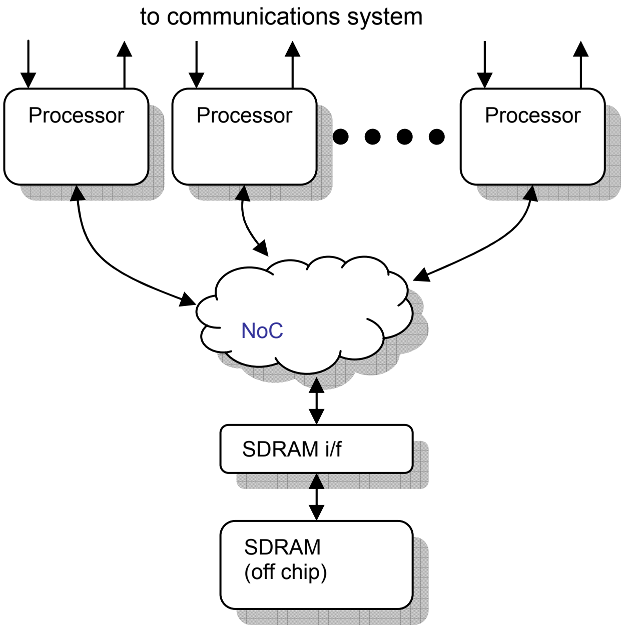


Figure 3: The CMP system NoC.

power-efficient embedded processors, and have chosen the ARM968 as offering the best balance of performance, area, power-efficiency and ease of use for our purposes. The ARM968 is a synthesizable ARM9 processor core with tightly-coupled instruction and data memories, and an integral on-chip bus (ARM Ltd, 2004). Each processor subsystem comprises a processor, instruction and data memory, timers, interrupt and DMA controllers and a communications NoC interface (Fig. 2).

We estimate that a 200 MIPS integer embedded ARM9 processor should be able to model 1,000 leaky-integrate-and-fire (or Izhikevich) neurons, each with 1,000 inputs firing on average at 10 Hz, in real time. The synaptic connectivity information for these neurons requires around 4 Mbytes of memory and the neuron state requires around 50 Kbytes of memory. These estimates have led us to adopt a hybrid architecture where the synaptic data is held in an off-chip SDRAM while the neural state data is held in on-chip memory local to each embedded processor. A processing node in our system therefore comprises two ICs: a chip multiprocessor (CMP) with about twenty 200 MIPS embedded ARM9 processors, and an SDRAM chip. The synaptic data is accessed in large blocks and this enables an SDRAM bandwidth of around 1 GByte/s to provide this data at the required rate.

The processors on a CMP share access to the SDRAM using a self-timed packet-switched Network-on-Chip (NoC). This fabric will use the CHAIN technology (Bainbridge and Furber, 2002), developed at the University of Manchester and commercialized by Silistix Ltd, which gives a through-

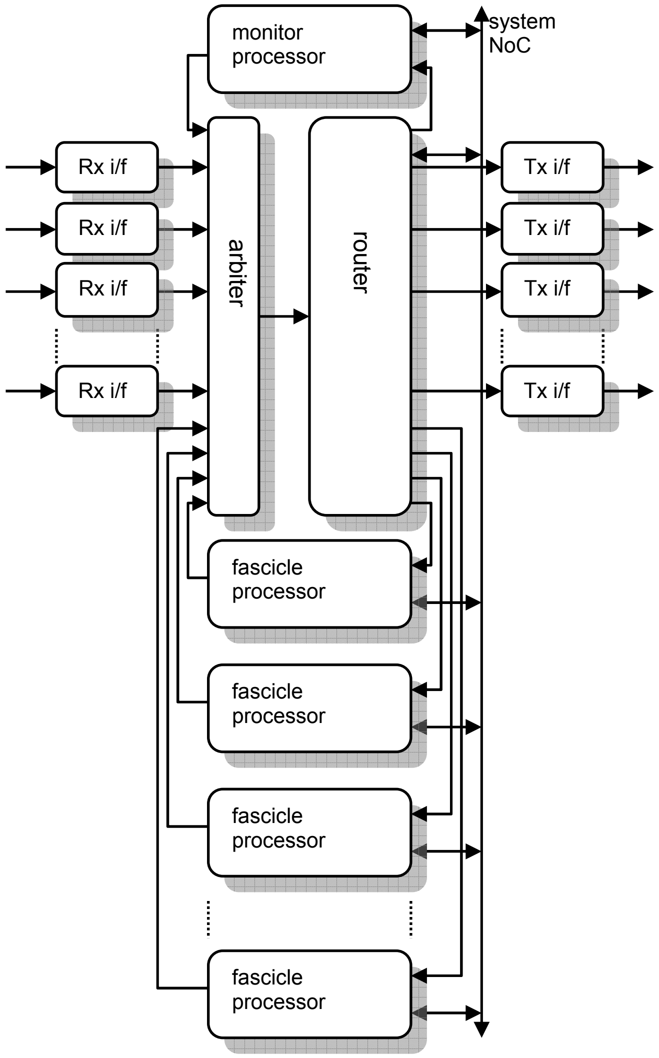


Figure 4: The communications NoC.

put of around 1 Gbit/s per 6-wire link (Bainbridge, Plana and Furber, 2004). The organization of the system NoC that connects the processor subsystems to the SDRAM is shown in Fig. 3.

6.3 The communications system

The major challenge in designing a scalable multi-chip neural modeling system is to emulate the very high connectivity of the biological system. The high fan-in and fan-out of neurons suggests that an efficient multicast communication system is required. We propose a communication NoC fabric based upon address-event signaling, but carried over a second self-timed packet-switched fabric rather than the usual bus-based fabric. The self-timed fabric decouples the many different clock domains within and across the CMPs.

The inter-chip communication uses a self-timed signalling system on an 8-wire inter-chip link that employs a self-timed 2-of-7 non-return-to-zero (NRZ) code (Bainbridge, Toms, Edwards and Furber, 2003) with an NRZ acknowledge. 16 of the 21 possible 2-of-7 codes are used to carry four bits of data, and a 17th code carries end-of-packet (EOP). Each 8-wire link has a capacity of around 1 Gbit/s when connecting two CMPs on the same circuit board, matching the on-chip bandwidth of a CHAIN link, and the self-timed protocol guarantees correct operation (albeit at a lower data rate) when the CMPs are on different circuit boards, automatically adapting to the addition delays incurred by any signal buffering that may be required.

The complete communications subsystem on a CMP is illustrated in Fig. 4. The inter-chip links are accessed via input protocol converters ('Rx i/f' in Fig. 4) that translate the off-chip 2-of-7 NRZ codes to the on-chip CHAIN codes, and output protocol converters ('Tx i/f') that perform the inverse translation. Each of the on-chip processing subsystems ('fascicle processor') is also a source of network traffic and a potential destination. All of the on- and off-chip sources are merged through an asynchronous arbiter into a single stream of packets that passes through the multicast router which will, in turn, propagate the packet to a subset of its on- and off-chip outputs. The monitor processor is identical to a fascicle processor but is dedicated to system management functions rather than neural modeling. It is chosen from among the fascicle processors at boot time; the flexibility in its selection removes another possible single point of failure on the CMP, improving fault tolerance.

The heart of the communication subsystem is the associative multicast router which directs every incoming packet to one or more of the local processors and output links using a routing key based on the source ID and a route look-up table.

6.4 Fault-tolerance

The scale of the proposed machine demands that it be designed with a high degree of fault-tolerance. Since the neural system we are modelling has intrinsic fault-tolerant properties (healthy humans lose about one neuron a second throughout their adult life; neurodegenerative diseases incur much higher loss rates) this capacity will be transferred to the simulated system to some degree. However, many of the techniques we employ to map the natural system onto the electronic model concentrate distributed biological processes into single points of failure in the model: a single microprocessor models a thousand neurons; a single inter-chip link carries the spikes on perhaps a million axons. Thus we must engineer some additional resilience into the electronic system.

The highly regular structure of the machine comes to our aid here. If a processor fails we can migrate its workload to another, on the same or on a different chip. This will almost certainly lead to a glitch in the system's real-time performance, but our goal is to minimise the size of this glitch and to build a system that is continuously monitoring its own performance and migrating its workload to minimise congestion, so a major failure just puts a higher transient demand on the workload management processes.

An inter-chip link failure (whether permanent or transient, perhaps due to local congestion) will be handled in the first instance at the hardware level, redirecting traffic automatically via an adjacent link, before invoking the performance management software to carry out a more permanent solution.

At all stages in the design we are exploring opportunities to identify mechanisms that support real-time fault-tolerance, some of which exploit the intrinsic fault-tolerance of neural systems but many of which will contribute to a separate research agenda in the area of autonomic, self-healing systems.

7 Conclusions

The Grand Challenge of understanding the Architecture of Brain and Mind is a multidisciplinary quest that will require many complementary approaches to run concurrently, each feeding off the others as sources of inspiration, ideas and sanity checks. The system synthesis approach of computer engineers such as ourselves may have something to contribute as a component of the overall process. An understanding of complex asynchronous interactions within digital systems seems highly relevant to the task of understanding the complex asynchronous interactions between neurons.

In our quest to understand the dynamics of systems of asynchronous spiking neurons we hope to contribute both to providing tools that help understand biological brains and also to the creation of novel computational systems that are inspired by biology, but whose link to biology may ultimately become tenuous.

To this end we propose to construct a massively-parallel computer that implements a universal spiking neural network architecture, SpiNNaker. Based on a chip multiprocessor incorporating around twenty 200 MIPS embedded ARM968 processors, and employing a communications infrastructure specifically designed to support the multicast routing required for neural simulation, this system will scale to hundreds of thousands of processors modelling up to a billion neurons in real time. It will form a 'sandpit' in which we, and others with similar interests, can experiment with large-scale systems of spiking neurons to test our network topologies and neural models in order to validate (or disprove) our theories of how neurons interact to generate the hardware platform that underpins the Architecture of Brain and Mind.

Acknowledgements

This work is supported by the EPSRC Advanced Processor Technologies Portfolio Partnership Award. Steve Furber holds a Royal Society-Wolfson Research Merit Award. The support of ARM Ltd and Silistix Ltd for the proposed work is gratefully acknowledged.

References

ARM Ltd. ARM968E-S Technical Reference Manual. DDI 0311C, 2004. http://www.arm.com/products/CPUs/ARM968E-S.html

W. J. Bainbridge and S. B. Furber. CHAIN: A Delay-Insensitive Chip Area Interconnect. *IEEE Micro*, 22(5):16-23, 2002.

W. J. Bainbridge, L. A. Plana and S. B. Furber. The Design and Test of a Smartcard Chip Using a CHAIN Self-timed Network-on-Chip. *Proc. DATE'04*, 3:274, Paris, Feb 2004.

W. J. Bainbridge, W. B. Toms, D. A. Edwards and S. B. Furber. Delay-Insensitive, Point-to-Point Interconnect using m-of-n codes. *Proc. Async* :132-140, Vancouver, May 2003.

S. B. Furber, W. J. Bainbridge, J. M. Cumpstey and S. Temple. A Sparse Distributed Memory based upon N-of-M Codes. *Neural Networks* 17(10):1437-1451, 2004.

S. B. Furber, S. Temple and A. D. Brown. On-chip and Inter-Chip Networks for Modelling Large-Scale Neural Systems. *Proc. ISCAS'06*, Kos, May 2006 (to appear).

E. M. Izhikevich. Which Model to Use for Cortical Spiking Neurons? *IEEE Trans. Neural Networks,* 15:1063-1070, 2004.

A. Sloman (ed.). The Architecture of Brain and Mind. UKCRC Grand Challenge 5 report, 2004.

Principles Underlying the Construction of Brain-Based Devices

Jeffrey L. Krichmar
The Neurosciences Institute
10640 John J. Hopkins Drive
San Diego, CA 92121 USA
krichmar@nsi.edu

Gerald M. Edelman
The Neurosciences Institute
10640 John J. Hopkins Drive
San Diego, CA 92121 USA
edelman@nsi.edu

Abstract

Without a doubt the most sophisticated behaviour seen in biological agents is demonstrated by organisms whose behaviour is guided by a nervous system. Thus, the construction of behaving devices based on principles of nervous systems may have much to offer. Our group has built series of brain-based devices (BBDs) over the last 14 years to provide a heuristic for studying brain function by embedding neurobiological principles on a physical platform capable of interacting with the real world. These BBDs have been used to study perception, operant conditioning, episodic and spatial memory, and motor control through the simulation of brain regions such as the visual cortex, the dopaminergic reward system, the hippocampus, and the cerebellum. Following the brain-based model, we argue that an intelligent machine should be constrained by the following design principles: (i) it should incorporate a simulated brain with detailed neuroanatomy and neural dynamics that controls behaviour and shapes memory, (ii) it should organize the unlabeled signals it receives from the environment into categories without a priori knowledge or instruction, (iii) it should have a physical instantiation, which allows for active sensing and autonomous movement in the environment, (iv) it should engage in a task that is initially constrained by minimal set of innate behaviours or reflexes, (v) it should have a means to adapt the device's behaviour, called value systems, when an important environmental event occurs, and (vi) it should allow comparisons with experimental data acquired from animal nervous systems. Like the brain, these devices operate according to selectional principles through which they form categorical memory, associate categories with innate value, and adapt to the environment. Moreover, this approach may provide the groundwork for the development of intelligent machines that follow neurobiological rather than computational principles in their construction.

1 Introduction

Although much progress has been made in the neurosciences over the last several decades, the study of the nervous system is still a wide open area of research. This is not due to a lack of first-rate research by the neuroscience community, but instead it reflects the complexity of the problem. Therefore, novel approaches to the problem, such as computational modelling and robotics, may be necessary to come to a better understanding of brain function. Moreover, as our models and devices become more sophisticated and more biologically realistic, the devices themselves may approach the complexity and adaptive behaviour that we associate with biological organisms and may find their way in practical applications. In this review, we will outline what we believe are the design principles necessary to achieve these goals (Krichmar and Edelman, 2005; Krichmar and Reeke, 2005). We will illustrate how these principles have been put into practice by describing two recent brain-based devices (BBDs) from our group.

2 Brain-Based Modelling Design Principles

2.1 Incorporate A Simulated Brain With Detailed Neuroanatomy And Neural Dynamics

Models of brain function should take into consideration the dynamics of the neuronal elements that make up different brain regions, the structure of these different brain regions, and the connectivity within and between these brain regions. The dynamics of the elements of the nervous system (e.g. neuronal activity and synaptic transmission) are important to brain function and have been modelled at the single neuron level (Borg-Graham, 1987; Bower and Beeman, 1994; Hines and Carnevale, 1997), network level (Izhikevich et al., 2004; Pinsky and Rinzel, 1994), and synapse level in models of plasticity (Bienenstock et al., 1982; Song et al., 2000; Worgotter and Porr, 2005). However, structure at the gross anatomical level is critical for function, and it has often been ignored in models of the nervous system. Brain function is more than the activity of disparate regions; it is the interaction between these areas that is crucial as we have shown in Darwins IV through X (Edelman et al., 1992; Krichmar and Edelman, 2005; Krichmar et al., 2005b; Seth et al., 2004). Brains are defined by a distinct neuroanatomy in which there are areas of special function, which are defined by their connectivity to sensory input, motor output, and to each other.

2.2 Organize the Signals from the Environment into Categories Without a priori Knowledge or Instruction

One essential property of BBDs, is that, like living organisms, they must organize the unlabeled signals they receive from the environment into categories. This organization of signals, which in general depends on a combination of sensory modalities (e.g. vision, sound, taste, or touch), is called *perceptual categorization*. Perceptual categorization in models (Edelman and Reeke, 1982) as well as living organisms makes object recognition possible based on experience, but without *a priori* knowledge or instruction. A BBD selects and generalizes the signals it receives with its sensors, puts these signals into categories without instruction, and learns the appropriate actions when confronted with objects under conditions that produce responses in value systems.

2.3 Active Sensing and Autonomous Movement in the Environment

Brains do not function in isolation; they are tightly coupled with the organism's morphology and environment. In order to function properly, an agent, artificial or biological, needs to be situated in the real world (Chiel and Beer, 1997; Clark, 1997). Therefore, models of brain function should be embodied in a physical device and explore a real as opposed to a simulated environment. For our purposes, the real environment is required for two reasons. First, simulating an environment can introduce unwanted and unintentional biases to the model. For example, a computer generated object presented to a vision model has its shape and segmentation defined by the modeller and directly presented to the model, whereas a device that views an object hanging on a wall has to discern the shape and figure from ground segmentation based on its on active vision. Second, real environments are rich, multimodal, and noisy; an artificial design of such an environment would be computationally intensive and difficult to simulate. However, all these interesting features of the environment come for "free" when we place the BBD in the real world. The modeller is freed from simulating a world and need only concentrate on the development of a device that can actively explore the real world.

2.4 Engage in a Behavioural Task

It follows from the above principle that a situated agent needs to engage in some behavioural task. Similar to a biological organism, an agent or BBD needs a minimal set of innate behaviours or reflexes in order to explore and initially survive in its environmental niche. From this minimal set, the BBD can learn and adapt such that it optimizes its behaviour. How these devices adapt is the subject of the next principle, which describes value systems (see section 2.5). This approach is very different from the classic artificial intelligence or robotic control algorithms, where either rules or feedback controllers with pre-defined error signals need to be specified *a priori*. In the BBD approach, the agent selects what it needs to optimize its behaviour and thus adapts to its environment.

A second and important point with regard to behavioural tasks is that it gives the researcher a metric by which to score the BBD's performance. Moreover, these tasks should be made similar to experimental biology paradigms so that the behavioural performance of the BBD can be compared with that of real organisms (see section 2.6 below).

2.5 Adapt Behaviour when an Important Environmental Event Occurs

Biological organisms adapt their behaviour through value systems, which provide non-specific, modulatory signals to the rest of the brain that bias the outcome of local changes in synaptic efficacy in the direction needed to satisfy global needs. Stated in the simplest possible terms, behaviour that evokes positive responses in value systems biases synaptic change to make production of the same behaviour more likely when the situation in the environment (and thus the local synaptic inputs) is similar; behaviour that evokes negative value biases synaptic change in the opposite direction. Examples of value systems in the brain include the dopaminergic, cholinergic, and noradrenergic systems (Aston-Jones and Bloom, 1981; Hasselmo et al., 2002; Schultz et al., 1997) which respond to environmental cues signalling reward prediction, uncertainty, and novelty. Theoretical models based of these systems and their effect on brain function have been developed (Doya, 2002; Friston et al., 1994; Montague et al., 1996; Yu and Dayan, 2005) and embedded in real world behaving devices (Arleo et al., 2004; Krichmar and Edelman, 2002; Sporns and Alexander, 2002).

2.6 Comparisons with Experimental Data Acquired from Animal Models

The behaviour of BBDs and the activity of their simulated nervous systems must be recorded to allow comparisons with experimental data acquired from animals. The comparison should be made at the behavioural level, the systems level, and the neuronal element level. These comparisons serve two purposes: First, BBDs are powerful tools to test theories of brain function. The construction of a complete behaving model forces the designer to specify theoretical and implementation details that are easy to overlook in a purely verbal description and it forces those details to be consistent among them. The level of analysis permitted by having a recording of the activity of every neuron and synapse in the simulated nervous system during its behaviour is just not possible with animal experiments. The results of such situated models have been compared with rodent hippocampal activity during navigation, basal ganglia activity during action selection, and attentional systems in primates (Burgess et al., 1997; Guazzelli et al., 2001; Itti, 2004; Prescott et al., 2006). Second, by using the animal nervous system as a metric, designers can continually make their simulated nervous systems closer to that of the model animal. This, in turn, allows the eventual creation of practical devices that may approach the sophistication of living organisms.

3 Illustrative Examples of Brain-Based Devices

In this section, we will use our group’s two most recent BBDs as illustrative examples of the above principles. The first example, Darwin X (Krichmar et al., 2005a; Krichmar et al., 2005b), is a BBD that develops spatial and episodic memory by incorporating a detailed model of the hippocampus and its surrounding regions. The second example is a BBD capable of predictive motor control based on a model of cerebellar learning (McKinstry et al., 2006).

3.1 An Embodied Model of Spatial and Episodic Memory

Darwin X was used to investigate the functional anatomy specific to the hippocampal region during a memory task. Darwin X incorporates aspects of the anatomy and physiology of the hippocampus and its surrounding regions, which are known to be necessary for the acquisition and recall of spatial and episodic memories. The simulated nervous system contained 50 neural areas, 90,000 neuronal units, and 1.4 million synaptic connections. It included a visual system, a head direction system, a hippocampal formation, a basal forebrain, a value or reward system, and an action selection system. Darwin X used camera input to recognize the category and position of distal visual objects and used odometry to develop head direction sensitivity.

Darwin X successfully demonstrated the acquisition and recall of spatial and episodic memories in a maze task similar to the Morris water maze (Morris, 1984) by associating places with actions. The association was facilitated by a dopaminergic value system based on the known connectivity between CA1 and nucleus accumbens and frontal areas (Thierry et al., 2000). The responses of simulated neuronal units in the hippocampal areas during its exploratory behaviour were comparable to neuronal responses in the rodent hippocampus; i.e., neuronal units responded to a particular location within Darwin X’s environment (O'Keefe and Dostrovsky, 1971).

Darwin X took into consideration the macro- and micro-anatomy between the hippocampus and cortex, as well as the within the hippocampus. In order to identify different functional hippocampal pathways and their influence on behaviour, we developed two novel methods for analyzing large scale neuronal networks: 1) Backtrace - tracing functional pathways by choosing a unit at a specific time and recursively examining all neuronal units that led to the observed activity in this reference unit (Krichmar et al., 2005a), and 2) Causality - a time series analysis that distinguishes causal interactions within and between neural regions (Seth, 2005). These analyses allowed us to examine the information flow through the network and highlighted the importance of the perforant pathway from the entorhinal cortex to the hippocampal subfields in producing associations between the position of the agent in space and the appropriate action it needs to reach a goal. This functional pathway has recently been identified in the rodent (Brun et al., 2002).

As with other BBDs in the Darwin series, Darwin X follows the brain-based modelling principles. It is a physical device in a real world that carries out a task similar to that conducted with animal models. It adapts its behaviour based on its value system, and the dynamics of its nervous system were analyzed during its behaviour and compared with the responses of real nervous systems.

3.2 A Model of Predictive Motor Control Based On Cerebellar Learning and Visual Motion

Recently, our group constructed a BBD which included a detailed model of the cerebellum and cortical areas that respond to visual motion (McKinstry et al., 2006). One theory of cerebellar function proposes that the cerebellum learns to replace reflexes with a predictive controller (Wolpert et al., 1998). Synaptic eligibility traces in the cerebellum have recently been proposed as a specific mechanism for such motor learning (Medina et al., 2005). We tested whether a learning mechanism, called the delayed eligibility trace learning rule, could account for the predictive nature of the cerebellum in a real-world, robotic visuomotor task.

The BBD's visuomotor task was to navigate a path designated by orange traffic cones. The platform for this task was a Segway Robotic Mobility Platform modified to have a camera, a laser range finder, and infrared proximity detectors as inputs. The BBD's nervous system contained components simulating the cerebellar cortex, the deep cerebellar nuclei, the inferior olive, and a cortical area MT. The simulated cortical area MT, which responds to visual motion, was constructed based on the suggestion that the visual system makes use of visual blur for determining motion direction (Geisler, 1999; Krekelberg et al., 2003). The simulated nervous system contained 28 neural areas, 27,688 neuronal units, and 1.6 million synaptic connections. Using an embedded Beowulf computer cluster of six compact personal computers, it took roughly 40 ms to update all the neuronal units and plastic connections in the model each simulation cycle. Initially, path traversal relied on a reflexive movement away from obstacles that was triggered by infrared proximity sensors when the BBD was within 12 inches of a cone. This resulted in clumsy, crooked movement down the path. The infrared sensor input was also the motor error signal to the cerebellum via simulated climbing fibre input. Over time, the cerebellar circuit predicted the correct motor response based on visual motion cues preventing the activation of the reflex and resulting in smooth movement down the centre of the path. The system learned to slow down prior to a curve and to turn in the correct direction based on the flow of visual information. The system adapted to and generalized over different courses with both gentle and sharp angle bends.

The experiments, which depend both on the dynamics of the delayed trace eligibility learning and on the architecture of the cerebellum, demonstrated how the cerebellum can predict impending errors and adapt its movements. Moreover, by analyzing the responses of the cerebellum and the inputs from the simulated area MT during its behaviour, we were able to predict the types of signals the nervous system might select to adapt to such a motor task. The BBD's nervous system categorized the motion cues that were predictive of different collisions and associated those categories with the appropriate movements. The neurobiologically inspired model described here prompts several hypotheses about the relationship between perception and motor control and may be useful in the development of general-purpose motor learning systems for machines.

4 Conclusions

Higher brain functions depend on the cooperative activity of an entire nervous system, reflecting its morphology, its dynamics, and its interaction with its phenotype and the environment. BBDs are designed to incorporate these attributes such that they can test theories of brain function. Like the brain, they operate according to selectional principles through which they form categorical memory, associate categories with innate value, and adapt to the environment. These BBDs also provide the groundwork for the development of intelligent machines that follow neurobiological rather than computational principles in their construction.

Acknowledgements

This work was supported by grants from the Office of Naval Research, DARPA, and the Neurosciences Research Foundation.

References

Arleo, A., Smeraldi, F., and Gerstner, W. (2004). Cognitive navigation based on nonuniform Gabor space sampling, unsupervised growing networks, and reinforcement learning. IEEE Trans Neural Netw *15*, 639-652.

Aston-Jones, G., and Bloom, F. E. (1981). Nonrepinephrine-containing locus coeruleus neurons in behaving rats exhibit pronounced responses to non-noxious environmental stimuli. J Neurosci *1*, 887-900.

Bienenstock, E. L., Cooper, L. N., and Munro, P. W. (1982). Theory for the development of neuron selectivity: orientation specificity and binocular interaction in visual cortex. J Neurosci *2*, 32-48.

Borg-Graham, L. (1987). Modelling the electrical behavior of cortical neurons - simulations of hippocampal pyramidal cells., In Computer Simulation in Brain Science, R. M. J. Cotterill, ed. (Cambridge: Cambridge University Press).

Bower, J. M., and Beeman, D. (1994). The Book of GENESIS: Exploring Realistic Neural Models with the GEneral NEural SImulation System.: TELOS/Springer-Verlag).

Brun, V. H., Otnass, M. K., Molden, S., Steffenach, H. A., Witter, M. P., Moser, M. B., and Moser, E. I. (2002). Place cells and place recognition maintained by direct entorhinal-hippocampal circuitry. Science *296*, 2243-2246.

Burgess, N., Donnett, J. G., Jeffery, K. J., and O'Keefe, J. (1997). Robotic and neuronal simulation of the hippocampus and rat navigation. Philos Trans R Soc Lond B Biol Sci *352*, 1535-1543.

Chiel, H. J., and Beer, R. D. (1997). The brain has a body: adaptive behavior emerges from interactions of nervous system, body and environment. Trends Neurosci *20*, 553-557.

Clark, A. (1997). Being there. Putting brain, body, and world together again. (Cambridge, MA: MIT Press).

Doya, K. (2002). Metalearning and neuromodulation. Neural Netw *15*, 495-506.

Edelman, G. M., Reeke, G. N., Gall, W. E., Tononi, G., Williams, D., and Sporns, O. (1992). Synthetic neural modeling applied to a real-world artifact. Proc Natl Acad Sci U S A *89*, 7267-7271.

Edelman, G. M., and Reeke, G. N., Jr. (1982). Selective networks capable of representative transformations, limited generalizations, and associative memory. Proc Natl Acad Sci U S A *79*, 2091-2095.

Friston, K. J., Tononi, G., Reeke, G. N., Sporns, O., and Edelman, G. M. (1994). Value-dependent selection in the brain: simulation in a synthetic neural model. Neuroscience *59*, 229-243.

Geisler, W. S. (1999). Motion streaks provide a spatial code for motion direction. Nature *400*, 65-69.

Guazzelli, A., Bota, M., and Arbib, M. A. (2001). Competitive Hebbian learning and the hippocampal place cell system: modeling the interaction of visual and path integration cues. Hippocampus *11*, 216-239.

Hasselmo, M. E., Hay, J., Ilyn, M., and Gorchetchnikov, A. (2002). Neuromodulation, theta rhythm and rat spatial navigation. Neural Netw *15*, 689-707.

Hines, M. L., and Carnevale, N. T. (1997). The NEURON simulation environment. Neural Comput *9*, 1179-1209.

Itti, L. (2004). Automatic foveation for video compression using a neurobiological model of visual attention. IEEE Trans Image Process *13*, 1304-1318.

Izhikevich, E. M., Gally, J. A., and Edelman, G. M. (2004). Spike-timing dynamics of neuronal groups. Cereb Cortex *14*, 933-944.

Krekelberg, B., Dannenberg, S., Hoffmann, K. P., Bremmer, F., and Ross, J. (2003). Neural correlates of implied motion. Nature *424*, 674-677.

Krichmar, J. L., and Edelman, G. M. (2002). Machine psychology: autonomous behavior, perceptual categorization and conditioning in a brain-based device. Cereb Cortex *12*, 818-830.

Krichmar, J. L., and Edelman, G. M. (2005). Brain-based devices for the study of nervous systems and the development of intelligent machines. Artif Life *11*, 63-77.

Krichmar, J. L., Nitz, D. A., Gally, J. A., and Edelman, G. M. (2005a). Characterizing functional hippocampal pathways in a brain-based device as it solves a spatial memory task. Proc Natl Acad Sci U S A *102*, 2111-2116.

Krichmar, J. L., and Reeke, G. N. (2005). The Darwin Brain-Based Automata: Synthetic Neural Models and Real-World Devices, In Modeling in the Neurosciences: From Biological Systems to Neuromimetic Robotics, G. N. Reeke, R. R. Poznanski, K. A. Lindsay, J. R. Rosenberg, and O. Sporns, eds. (Boca Raton: Taylor & Francis), pp. 613-638.

Krichmar, J. L., Seth, A. K., Nitz, D. A., Fleischer, J. G., and Edelman, G. M. (2005b). Spatial navigation and causal analysis in a brain-based device modeling cortical-hippocampal interactions. Neuroinformatics *3*, 197-221.

McKinstry, J. L., Edelman, G. M., and Krichmar, J. L. (2006). A cerebellar model for predictive motor control tested in a brain-based device. Proc Natl Acad Sci U S A.

Medina, J. F., Carey, M. R., and Lisberger, S. G. (2005). The representation of time for motor learning. Neuron *45*, 157-167.

Montague, P. R., Dayan, P., and Sejnowski, T. J. (1996). A framework for mesencephalic dopamine systems based on predictive Hebbian learning. J Neurosci *16*, 1936-1947.

Morris, R. (1984). Developments of a water-maze procedure for studying spatial learning in the rat. J Neurosci Methods *11*, 47-60.

O'Keefe, J., and Dostrovsky, J. (1971). The hippocampus as a spatial map. Preliminary evidence from unit activity in the freely-moving rat. Brain Res *34*, 171-175.

Pinsky, P. F., and Rinzel, J. (1994). Intrinsic and network rhythmogenesis in a reduced Traub model for CA3 neurons. J Comput Neurosci *1*, 39-60.

Prescott, T. J., Montes Gonzalez, F. M., Gurney, K., Humphries, M. D., and Redgrave, P. (2006). A robot model of the basal ganglia: Behavior and intrinsic processing. Neural Netw *19*, 31-61.

Schultz, W., Dayan, P., and Montague, P. R. (1997). A neural substrate of prediction and reward. Science *275*, 1593-1599.

Seth, A. K. (2005). Causal connectivity of evolved neural networks during behavior. Network *16*, 35-54.

Seth, A. K., McKinstry, J. L., Edelman, G. M., and Krichmar, J. L. (2004). Active sensing of visual and tactile stimuli by brain-based devices. International Journal of Robotics and Automation *19*, 222-238.

Song, S., Miller, K. D., and Abbott, L. F. (2000). Competitive Hebbian learning through spike-timing-dependent synaptic plasticity. Nat Neurosci *3*, 919-926.

Sporns, O., and Alexander, W. H. (2002). Neuromodulation and plasticity in an autonomous robot. Neural Netw *15*, 761-774.

Thierry, A. M., Gioanni, Y., Degenetais, E., and Glowinski, J. (2000). Hippocampo-prefrontal cortex pathway: anatomical and electrophysiological characteristics. Hippocampus *10*, 411-419.

Wolpert, D., Miall, R., and Kawato, M. (1998). Internal models in the cerebellum. Trends in Cognitive Sciences *2*, 338-347.

Worgotter, F., and Porr, B. (2005). Temporal sequence learning, prediction, and control: a review of different models and their relation to biological mechanisms. Neural Comput *17*, 245-319.

Yu, A. J., and Dayan, P. (2005). Uncertainty, neuromodulation, and attention. Neuron *46*, 681-692.

Emotion as an integrative process between non-symbolic and symbolic systems in intelligent agents

Ruth Aylett
MACS, Heriot-Watt University
Riccarton, Edinburgh EH10 4ET
ruth@macs.hw.ac.uk

Abstract

This paper briefly considers the story so far in AI on agent control architectures and the later equivalent debate between symbolic and situated cognition in cognitive science. It argues against the adoption of a reductionist position on symbolically-represented cognition but in favour of an account consistent with embodiment. Emotion is put forward as a possible integrative mechanism via its role in the management of interaction between processes and a number of views of emotion are considered. A sketch of how this interaction might be modelled is discussed.

1. Embodied cognition

Within AI, the problem of relating cognition and action has led to a well-known division of opinion since about the mid 1980s between the older symbolic computing position that classically saw action as a one-many decomposition of abstract planned actions from a symbolically-represented control level and the situated agent view, as in Brooks (1986), that saw action as a tight stimulus-response coupling that did not require any symbolic representation. This can be – and originally was – posed as an engineering question of how to produce systems that were able to act competently in the real world, hence the origin of the argument in robotics, where the real world cannot be wished away and where robot sensing systems do not deliver symbols.

At this engineering level, the 1990s saw a pragmatic reconciliation of these divergent positions in hybrid architectures, usually with three levels (Gat 97, Arkin and Balch 97, Barnes et al 97), in which the relationship between symbolic planning systems and non-symbolic reactive systems was resolved by giving the reactive systems ultimate control of execution but either allowing planning to be invoked as a resource when needed (for example as a conflict resolution mechanism, or to provide sequencing capabilities) or giving planning systems to ability to constrain and contextualise – but not determine - the reactive system in what is sometimes known as supervisory control.

However the argument that was carried on from an engineering perspective in robotics, and which to some extent is now a done deal, has subsequently continued at a more scientific level in cognitive science, a discipline within psychology that arguably owed its existence to ideas from classical AI and was heavily influenced by the symbolic world-view as seen in large-scale computational cognitive models such as SOAR (Rosenbloom et al 93) and ACT-R (Anderson et al 04). Just as in pre-Brooksian robotics, these models can be criticised for not providing any adequate account of the role of sensing or motor action, which are implicitly seen as peripheral to a cognitive model much as I-O capabilities are peripheral to a computer processor.

A deeper criticism arises from a view of agency which sees *embodiment* as a key starting point rather than an optional extra (Clark 98) and starts from a body in a specific environment that needs a mind to control it rather than a mind considering abstract problems. In the world-view of embodied cognition (Wilson 02), sensori-motor engagement is the ground from which cognitive abilities are constructed (as in Piaget's developmental psychology); thus neither cognitive abilities nor specific environments and interactions can be detached in the way that had been previously assumed, and a dynamic and process-based view supersedes a declarative and logical-inferencing based view. The Cartesian separation between mind and body which still seems to exist in multi-level architectures is abandoned in favour of brain-body integration, in which processes such as the endocrine system play a vital coordinating role.

This does not mean however that a symbolic account of cognition is a pointless activity either from a scientific or an engineering perspective (so we do not actually have to abandon the whole of cognitive science up to now as well as a large chunk of psychology). Just as computational neuro-physiology does not operate at the explanatory level of physics, there is no reason why cognition based on symbolic reference must be reduced to neuro-physiology, even though what is known about the way in which the brain works suggests that symbol manipulation is an emergent property of the dynamic system formed by interaction between neurons. Indeed, the very concept of emergence dictates that an emergent phenomenon is modelled at its own level of representation since it cannot be decomposed into any one of the components whose interaction produces it. Thus an account at the symbolic level may be a valid one as long as it does not incorporate incorrect assumptions about the relationship between this activity and sensori-motor engagement.

The power of symbolic reference to abstract out of the current sensory context, to project into imagined contexts, to discretise continuous experiences into conceptual aggregates, to communicate through language and to use the environment to scaffold engagement with the world and other humans (as through writing) has a substantial impact on both cognition and interaction for humans. Thus current work on social agents that aims to produce more human-like systems whether via robots or graphical characters must incorporate symbol-manipulation systems as well as the sensor-driven systems that allow them to act with some degree of competence in their physical or virtual world.

However an important characteristic of these human abilities that has not been replicated in computer-based systems is that unification of symbol-manipulation abilities with non-symbolic behaviours driven more directly by sensing that we observe in human activity. In the human case, the hypothesis that symbol manipulation is an emergent property of interaction between brain components suggests that the ability to move smoothly between cognition and reactive engagement with the world is probably a matter of adjusting the interaction between these components and does not therefore require explicit conversion between multiple representations in the way this is typically carried out in current hybrid agent architectures. Unfortunately the computational neuro-physiology account of specific brain subsystems is still fragmentary, and there is no short-term (or even medium-term) likelihood of producing the principled interactional account that this hypothesis requires. Arguably, solving the problem of emergent symbolic reference would not only deal with the origin of language but possibly also with consciousness. It is thus a non-trivial enterprise.

How then are we to proceed with an integrated account that is not merely a pragmatic engineering kludge needed to produce competent social agents? The argument of this paper is that one should see process regulation as the key to the enterprise since even if the detailed mechanisms adopted may be invalidated by more extensive models of the brain, the basic approach is likely to be compatible with such models. The specific hypothesis of this paper is that affective systems should be considered as a key component of such regulation because of their role in attentional focus, in relation both to perception and cognition, as well as the management of goals, the allocation of internal resources, and the balance between thinking and acting.

2. Accounts of emotion

Just as accounts of action split into two camps in AI from the mid 1980s, there are two corresponding accounts of emotion and its role with respect to agency, one more related to models of the brain and nervous system, and process-oriented, and one related to symbolic models of cognition dealing with goal management and inferencing.

The first of these views, which aims at neurophysiologically plausibility, models emotion as part of a homeostatic control mechanism. Often incorporating a model of the endocrine system (Canamero 98) it suggests that emotion should be viewed as the set of brain-body changes resulting from the movement of the current active point of a brain-body process outside of an organism-specific ‘comfort zone’. It does not therefore require a single meter-like component in an agent architecture to represent *an* emotion, but offers a distributed representation interpretable in terms of the internal process states and external expressive behaviour as an emotion.

As well as an independent system state, one can also regard emotion in this framework as modifying the impact of an incoming stimulus. Thus emotion can be incorporated into action selection both indirectly and directly in much the same way as perception, and indeed can be thought of as functioning rather like an internal sensing process concerned with all the other running processes.

The second view of emotion is usually known as *appraisal theory* since it assumes a cognitive appraisal associated with perception that assesses the relationship between symbolic categories established via a model hierarchy using perceptual input and the cognitive-level goals of the agent.

Specific appraisal-based theories that have proved highly influential in the construction of graphical characters are those of Ortony et al (1988), which was based on a taxonomy of 22 emotion types, each with an associated appraisal rule, and that of Lazarus (1984), which links appraisal to action via the concept of coping behaviour. Sherer

(01) decomposes appraisal into a sequence made up of Relevance Detection, Implication Assessment, Coping Potential Determination and Normative Significance Evaluation, but remains tied to a top-down view of emotion in which cognitive processing results in later physiological changes.

Interestingly however one can interpret appraisal as an abstraction on a process of the same type as the first view in which a goal takes the place of a comfort zone. This does not mean that goals could never be independently determined at the cognitive level, but it offers the possibility of propagating the state of the homeostatically-regulated non-symbolic systems into the more abstract representational space of symbolic cognition.

In contrast to both of the views discussed above, Izard (1993) takes a heterogeneous approach which does not rule out appraisal but argues that emotion can also be generated directly by the nervous system, as in the first account, by empathy and by highly intense states of physiological drives such as hunger or lust. Such an approach is consistent with the integration between both views of emotion as part of an integration between different types of process within an agent.

3. A sketch of interaction

It is very tempting to view the integrative functions we are seeking as a way of linking different *levels* within a multi-level hierarchy. Within robotics this is exactly how these issues are discussed: symbolic cognition is a high-level system while non-symbolic reactive systems are low-level. We have avoided this terminology so far because it is highly ambiguous in other fields. Within psychology and cognitive science, high-level could also mean evolutionarily more recent or conscious as distinct from sub-conscious.

However we have argued above that it does make sense to think of a representational level for symbolic processing even if the way in which we implement it in a computer is very different from the way it is implemented in the brain. Once time is taken into account it is also clear that the processes on which symbol manipulation depends run with fewer real time constraints in terms of delivering motor commands, are able conceptually to stay at what we would call a higher level of abstraction and as a result provide discrete categories covering what at a non-symbolic level would be seen as dynamic processes.

It is however misleading to think of reflection as an activity that runs wholly as symbol manipulation and reaction as an activity that does not use symbol manipulation at all. As Wilson (02) argues, reflection is grounded in the mechanisms of sensory processing and motor control that evolved for interaction with the environment even when it is being applied for purposes that do not require immediate activity in the specific environment of the current moment; what she describes as *offline cognition*. At the same time, appraisal is an example of online cognition which may be quite reactive.

In fact, emotion seems to be closely tied to the distribution of internal resource between the processes producing symbol manipulation and others with tighter connections to motor action, as witness emotional flooding, in which cognitive activity seems to substantially shut down. We can think of this as a type of internal attentional focus which may be more closely tied to the attentional focus proposed by perception for a tight loop with the current environment or less tightly coupled to it when more offline cognition is taking place.

As a sketch of interaction, we finally consider a possible relationship from symbolic to non-symbolic (what might be called top-down processing if we adopt the levels vocabulary) and then from non-symbolic to symbolic.

3.1 From symbolic to non-symbolic

As mentioned in section 1 above, this is an issue that has been relatively extensively discussed in robotics, though in practice, the difficulties of creating a robot that is able to carry out more than a very narrow repertoire of actions has made most implementations either highly reactive or partially scripted.

Here, the issue is how to avoid the one-many expansion of discrete categories – for example planned actions – from symbolic form into non-symbolic form in a rigid mapping independent of sensing capabilities as in the classic Shakey-like approach. A view of reflection as a set of constraints on reactive systems allows this deterministic mapping to be avoided and replaced by contextual activation of groups of reactive processes. Thus in Barnes et al (1997) a planner mapped planning operator pre-conditions into sensory pre-conditions that could be detected by reactive processes and named the set of processes to be activated or deactivated on such pre-conditions being perceived.

It has the advantage that it allows reflection to overrule specific actions by an agent as well as to enable them. This supports a model of 'double appraisal' situations where for example hitting someone who has been offensive is overruled because of the way it will make the agent look to other people in a social group.

This approach does not require the initiative for reflection to come from an external task – it is equally compatible with the invocation of planning when reactive systems need it, whether to take advantage of sequencing capabilities or to deal with a situation in which reactive systems are not succeeding. However in this case it depends on the integration in the opposite direction, which is much more problematic and much less discussed.

3.2 From non-symbolic to symbolic

If constraints allow symbolic systems to impact non-symbolic ones without wholly determining them, pattern recognition is an obvious mechanism through which symbolic systems can discretise the dynamic variation of non-symbolic systems. Interpreting sub-symbolic configurations as either drives or emotions allows them to act as motivations within the symbolic systems and thus to initiate symbolic system activities such as planning.

This can be invoked from the symbolic process ‘how do I feel?’ ‘what do I want to do?’, but clearly can also, as just mentioned allow the non-symbolic processes to do the invoking, largely by associating high-levels of emotion with specific motivations. Here we think of a motivation as distinguished from goals by temporal scope and generality – thus dealing with hunger by eating is a motivation, while buying sandwiches or looking in the fridge would be examples of goals arising from this motivation.

Within the symbolic systems then, emotion can be thought of as an integral part of goal management, and also as a heuristic weighting mechanism for large search spaces creating a search-oriented attentional focus. Internal attentional focus is a further example of the use of constraints as a modelling device: in this case non-symbolic systems may constrain what goals are considered by the symbolic systems, the extent or type of memory retrieval that can be carried out, or the extent or type of actions that can be considered in planning. One could also allow the non-symbolic systems to exercise control over the pattern-recognition mechanism required to deliver motivations to the symbolic systems so that a greater or lesser number of motivations are handled.

These are aspects of the regulation of resources between symbolic and non-symbolic processes, and given that emotion is also heavily involved in sensory-motor coupling in non-symbolic systems, a very high level of emotion may truncate the symbolic process search space to the point where cognition almost halts, allowing us to model emotional flooding.

4. What cognitive model?

This approach to integrating non-symbolic and symbolic systems requires a model of a different type from ACT-R or SOAR, though these both include mechanisms which can be applied within the symbolic systems. Meanwhile, neuro-physiological models of the brain are still too fragmentary and small-scale to be useful for this purpose.

One interesting and possibly more useful approach is offered by the PSI model of Dorner (Dorner and Hille 95). This focuses on emotional modulation of perception, action-selection, planning and memory access. Emotions are not defined as explicit states but rather emerge from modulation of information processing and action selection. These modulators include arousal level (speed of information processing), resolution level (carefulness and attentiveness of behavior) and selection threshold (how easy it is for another motive to take over), and thus provide the type of interface discussed in the last section to non-symbolic systems. The model also applies two built-in motivators - level of competence and level of uncertainty, which are thought of as the degree of capability of coping with differing perspectives and the degree of predictability of the environment.

It would be an overstatement to suggest that this model can be applied without alteration to the discussion of this paper – not least because aspects are specified too broadly for straightforward implementation (Bach 2003) – but the role of emotion it outs forward does correspond in part to the suggestions made here.

Acknowledgements

The EU Humaine Network of Excellence on affective processing (http://emotion-research.net) provided an environment within which extended discussion has taken place on the topic of this paper: however it remains the view of the author not of the network as a whole. The European Commission partly funds Humaine but has no responsibility for the views advanced.

References

Anderson, J.R; D. Bothell, M. D. Byrne, S. Douglas, C. Lebiere, and Y. Qin. (2004) An integrated theory of the mind. Psychological Review, 111(4):1036–1060, 2004.

Arkin, R.A. and Balch,T. (1997) AuRA: Principles and Practice in Review. Journal of Experimental and Theoretical AI, 9(2), 1997.

Bach, J (2003) The MicroPsi Agent Architecture. Proceedings of ICCM-5, International Conference on Cognitive Modeling Bamberg, Germany (pp. 15-20), 2003

Barnes, D.P; Ghanea-Hercock, R.A; Aylett, R.S. & Coddington, A.M. (1997) “Many hands make light work? An investigation into behaviourally controlled co-operant autonomous mobile robots”. Proceedings, 1st International Conference on Autonomous Agents, Marina del Rey, Feb 1997 pp413-20

Brooks, R.A.(1986) A robust layered control system for a mobile robot. IEEE Journal of Robotics and Automation, RA-2(1):14–23, March 1986

Canamero, D. (1998) Modeling Motivations and Emotions as a Basis for Intelligent Behaviour, Proceedings of the First International Conference on Autonomous Agents eds. Johnson, W. L. and Hayes-Roth, B. ACM Press pp148—155, 1998

Clark, A. (1998). Embodied, situated, and distributed cognition. In W. Bechtel & G. Graham (Eds.), *A companion to cognitive science* (pp. 506-517). Malden, MA: Blackwell.

Dorner, D., Hille, K.(1995): Artificial souls: Motivated emotional robots. In: Proceedings of the International Conference on Systems, Man and Cybernetics. (1995) 3828– 3832

Gat E. (1997) On Three-Layer Architectures, in Kortenkamp D., Bonasso R.P., Murphy R. (eds.): Artificial Intelligence and Mobile Robots, MIT/AAAI Press, 1997.

Izard, C. E. (1993). Four systems for emotion activation: Cognitive and noncognitive processes. Psychological Review, 100, 68-90

Lazarus, R.S& Folkman, S. (1984). Stress, appraisal and coping. New York: Springer

Ortony, A; Clore, G. L. and Collins, A. (1988) The Cognitive Structure of Emotions, Cambridge University Press 1988

Rosenbloom P., Laird J., and Newell A. eds. (1993) The Soar Papers: Research on Integrated Intelligence. MIT Press, Cambridge, Massachusetts, 1993.

Scherer, K. R. (2001). Appraisal considered as a process of multi-level sequential checking. In K. R. Scherer, A. Schorr, & T. Johnstone (Eds.), Appraisal processes in emotion. New York: Oxford University Press.

Velasquez, J.D. (1997) Modeling Emotions and Other Motivations in Synthetic Agents. Proceedings, 14th National Conference on AI, AAAI Press, pp10-16

Wilson, M. (2002) Six views of embodied cognition Psychonomic Bulletin & Review 2002, 9 (4), 625-636

Embodiment *vs.* Memetics: Is Building a Human getting Easier?

Joanna Bryson⋆
⋆Artificial models of natural Intelligence
University of Bath, United Kingdom
`jjb@cs.bath.ac.uk`

Abstract

This heretical article suggests that while embodiment was key to evolving human culture, and clearly affects our thinking and word choice now (as do many things in our environment), our culture may have evolved to such a point that a purely memetic AI beast could pass the Turing test. Though making something just like a human would clearly require both embodiment *and* memetics, if we were forced to choose one or the other, memetics might actually be easier. This short paper argues this point, and discusses what it would take to move beyond current semantic priming results to a human-like agent.

1 Embodiment

There is no doubt that embodiment is a key part of human and animal intelligence. Many of the behaviours attributed to intelligence are in fact a simple physical consequence of an animal's skeletal and muscular constraints (Port and van Gelder, 1995; Paul, 2004). Taking a learning or planning perspective, the body can be considered as bias, constraint or (in Bayesian terms) a prior for both perception and action which facilitates an animal's search for appropriate behaviour (Bryson, 2001).

This influence continues, arguably through all stages of reasoning (Chrisley and Ziemke, 2002; Lakoff and Johnson, 1999) but certainly at least sometimes to the level of semantics. For example, Glenberg and Kaschak (2002) have demonstrated the *action-sentence compatibility effect*. That is, subjects take longer to signal comprehension of a sentence with a gesture in the opposite direction as the motion indicated in the sentence than if the motion and sentence are compatible. For example, given a joystick to signal an understanding of 'open the drawer', it is easier to signal comprehension by pulling the joystick towards you than pushing it away. Boroditsky and Ramscar (2002) have shown that comprehension of ambiguous temporal events are strongly influenced by the hearer's physical situation with respect to current or imagined tasks and journeys.

These sorts of advances have lead some to suggest that the reason for the to-date rather unimpressive state of natural language comprehension and production in Artificially Intelligent (AI) systems is a consequence of their lack of embodiment (Harnad, 1990; Brooks and Stein, 2004; Roy and Reiter, 2005). The suggestion is that, in order to be meaningful, concepts must be grounded in the elements of intelligence that produce either action or perception salient to action.

The pursuit of embodied AI has lead us to understand resource-bounded reasoning which explains apparently suboptimal or inconsistent decision-making in humans (Chapman, 1987). It has also helped us to understand the extent to which agents can rely on the external world as a resource for cognition — that perception can replace or at least supplement long-term memory, reasoning and model building (Brooks, 1991; Clark, 1997; Ballard et al., 1997; Clark and Chalmers, 1998). However, despite impressive advances in the state of artificial embodiment (e.g. Chernova and Veloso, 2004; Schaal et al., 2003; Kortenkamp et al., 1998), there have been no clear examples of artificial natural language systems improved by embodiment.

I believe this is because embodiment, while necessary, is not a sufficient explanation of semantics. We *have* seen neat examples of the embodied acquisition of limited semantic systems (e.g Steels and Vogt, 1997; Steels and Kaplan, 1999; Roy, 1999; Billard and Dautenhahn, 2000; Sidnera et al., 2005). These systems show not only that semantics can be established between embodied agents, but also the relation between the developed lexicon and the agents' physical plants and perception. However, such examples give us little idea of how words like INFIN-

ITY, SOCIAL or REPRESENT might be represented. Further, they do not show the *necessity* of physical embodiment for a human-like level of comprehension of natural language semantics. On the other hand, it is possible that the semantic system underlying abstract words such as ‘justice’ may also be sufficient for terms originally referencing physical reality.

I do not contest the importance of understanding embodiment to understanding human intelligence as a whole. I *do* contest one of the prominent claims of the embodied intelligence movement — that embodiment is the only means of grounding semantics (Brooks and Stein, 2004). Roy and Reiter (2005) in fact *define* the term GROUNDED as ‘embodied’, which might be fine (compare with Harnad, 1990) if GROUNDED hadn’t also come to be synonymous with MEANINGFUL. The central claim of this paper is that while embodiment may have been the origin of most semantic meaning, it is no longer the only source for accessing a great deal of it. Further, some words (including their meanings) may have evolved more or less *independently* of grounded experience.

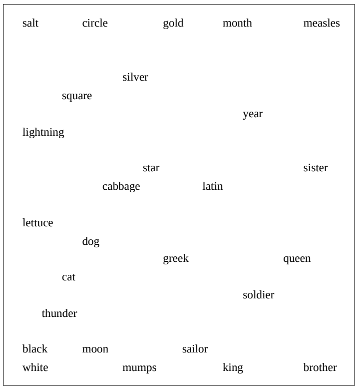
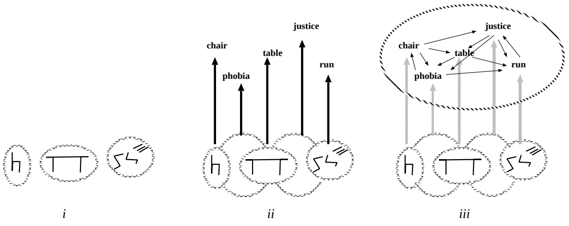


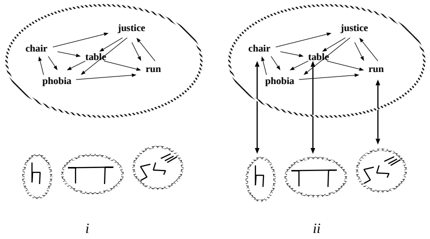
Figure 1: A two-dimensional projection of a semantic space, after Lowe (1997). The target words are taken from the experiments of Moss et al. (1995). Additional information on nearness is contained in the weights between locations in the 2-D space.

2 Memetics’ role in development

We now know that humans could very well develop an interconnected web of words *independently* of the process of developing grounded concepts (See Figure 1). Grounding then becomes a process of associating *some* of these statistically acquired terms with embodiment-based concepts. Thus children can learn and even use the word JUSTICE without a referent. Gradually as they gain experience of complexity of conflicting social goals and notions of fairness develop a richer notion of both what the word means and how and when both the word and the grounded concept can be used in furthering their goals. But even before that, a relatively naive reference to the term could well accomplish something.



(a) Deacon’s Theory



(b) Bryson’s Hypothesis

Figure 2: In Deacon’s theory, first concepts are learned *a(i)*, then labels for these concepts *a(ii)*, then a symbolic network somewhat like semantics *a(iii)*. I propose instead that grounded concepts and semantics are learned in parallel *b(i)*, then some semantic terms become understood *b(ii)*.

I want to be clear here: in my model, humans still acquire associations between these two representations, just as in the (Deacon, 1997) model that inspired it. What’s different is the ordering. In my model, lexical semantics is learned in parallel with embodied categories and expressed behaviour. Subsequently, *some* words become grounded as connections are formed between the two representations (see Figure 2). Nevertheless, this model also leaves the door open to true memetics — perhaps *justice* is an evolved concept that has fundamental impact in our culture and institutions without anyone truly ‘understanding’ it in any deeply grounded way.

3 Building someone cheaply

The previous sections have talked about what composes current human intelligence. But let’s change the topic now to trying to build someone capable of a decent conversation, even of coming up with the occasional good idea apparently on their own. Someone that could pass the Turing test if you chatted to them at the bus stop for 20 minutes, assuming you couldn't see what they looked like.

Figure 2(b) implies that memetics can only give us half the story, but this is wrong on two counts. First, I do not think embodiment is necessary for concept formation. We develop concept for justice to go along with the label, and I expect this same process could go on for quite a lot of other words.

It is possible that we'd need to provide some preformed seed concepts to get the system rolling. This may be necessary for two reasons:

- Purely for bootstrapping the learning system. It's possible that all concepts formed from memetic experience *are* formed partially in relation to or contrast with established concepts, so our poor disembodied mind might need some good, rich precocial concepts to get started (see further Sloman and Chappell, 2005).

- Because our memetic culture might not carry knowledge *everyone* gets for free. Given that a huge amount of what it means to be human is embedded in our semantic assumptions, it is possible that the brain can fill in the gaps. Stroke and lesion patients sometimes recover enormous functionality deficits if they still have enough of their brain intact that they can use the existing bits. If sufficiently stimulated (the main point of therapy), these surviving parts can provide enough information about what the *missing* information should look like that the individual may recover some lost skills. However, it is possible that some concepts are so incredibly universal to human experience that there just isn't enough information in the culture to reconstruct them.

But in general, I still think it might be easier to program some concepts (or proto-concepts) by hand than to build and maintain a robot that is sufficiently robust and long-lived, and has a sufficiently rich motor and sensor capacities, that it could do a better job of learning such concepts from its embodied experience.

But the other reason Figure 2(b) is not showing us that memetics is half the story is because a very important part of the story is left out. Even if we had an agent with all the knowledge of a human (or say we had a search engine with all the knowledge any human has ever put on the web), if all that agent ever does is *learns*, it isn't very human-like. To build someone, we need not only basic capacities for perception and action (which in the meme machine's case is just language in and out) but also motivation and action selection (Bryson, 2001). Even the cheapest human-like agent would need to have a set of prioritised goals, probably some sort of emotional / temporally dependent state to oscillate appropriately between priorities, and a set of plans (in this case, syntax and dialog patterns) to order its actions in such a way that it can achieve those goals.

Fortunately, nearly everyone in AI who builds agents (even roboticists) builds this part of the system in software, so again, there is no driving reason to bring in embodiment. Of course, without a body these goals would have to be purely intellectual or social (find out about you, talk about me, figure out how to use new words appropriately) — many but not all human goals would be inaccessible to a disembodied meme machine.

4 Conclusion

This short paper argues that although embodiment is clearly involved in human thought and language usage, we have consequently evolved and developed a culture permeated with the knowledge we derive in our embodied existence, and as such a cheap but reasonably entertaining agent might be built with no embodiment at all. Of course AI has tried to do this for several decades, but I think they have come at it the wrong way, focusing on logic-based reasoning too much and case- or template-based reasoning too little. Humans however are imitation and case-learning machines — to such an extent that some of our wisdom / common sense may well have evolved memetically rather than ever having been fully understood or reasoned about by anyone.

Acknowledgements

I'd like to thank Push Singh for getting me to think about common sense, even though we never agreed what it meant. I'd like to thank Will Lowe for teaching me about statistical semantics, even though we still don't agree what the implications are.

References

Dana H. Ballard, Mary M. Hayhoe, Polly K. Pook, and Rajesh P. N. Rao. Deictic codes for the embodiment of cognition. *Brain and Behavioral Sciences*, 20(4), December 1997.

A. Billard and K. Dautenhahn. Experiments in social robotics: grounding and use of communication in autonomous agents. *Adap. Behav.*, 7(3/4), 2000.

Lera Boroditsky and Michael Ramscar. The roles of body and mind in abstract thought. *Psychological Science*, 13(2):185–188, 2002.

Rodney A. Brooks. Intelligence without representation. *Artificial Intelligence*, 47:139–159, 1991.

Rodney A. Brooks and Lynn A. Stein. Building brains for bodies. *Aut. Robots*, 1(1):7–25, 2004.

Joanna J. Bryson. *Intelligence by Design: Principles of Modularity and Coordination for Engineering Complex Adaptive Agents*. PhD thesis, MIT, Department of EECS, Cambridge, MA, June 2001. AI Technical Report 2001-003.

David Chapman. Planning for conjunctive goals. *Artificial Intelligence*, 32:333–378, 1987.

S. Chernova and M. Veloso. Learning and using models of kicking motions for legged robots. In *The Proc. of ICRA-2004*, New Orleans, May 2004.

Ronald Chrisley and Tom Ziemke. Embodiment. In *Encyclopedia of Cognitive Science*, pages 1102–1108. Macmillan, 2002.

Andy Clark. *Being There: Putting Brain, Body and World Together Again*. MIT Press, Cambridge, MA, 1997.

Andy Clark and David Chalmers. The extended mind. *Analysis*, 58(1):7–19, 1998.

Terrence Deacon. *The Symbolic Species: The co-evolution of language and the human brain*. W. W. Norton & Company, New York, 1997.

Arthur M. Glenberg and Michael P. Kaschak. Grounding language in action. *Psychonomic Bulletin & Review*, 9(3):558–565, 2002.

Stevan Harnad. The symbol grounding problem. *Physica D*, 42(1–3):335–346, 1990.

David Kortenkamp, R. Peter Bonasso, and Robin Murphy, editors. *Artificial Intelligence and Mobile Robots: Case Studies of Successful Robot Systems*. MIT Press, Cambridge, MA, 1998.

George Lakoff and Mark Johnson. *Philosophy in the Flesh: The embodied mind and its challenge to Western thought*. Basic Books, New York, 1999.

Will Lowe. Semantic representation and priming in a self-organizing lexicon. In J. A. Bullinaria, D. W. Glasspool, and G. Houghton, editors, *Proc. of the Fourth Neural Computation and Psychology Workshop: Connectionist Representations (NCPW4)*, pages 227–239, London, 1997. Springer.

H. E. Moss, R. K. Ostrin, L. K. Tyler, and W. D. Marslen-Wilson. Accessing different types of lexical semantic information: Evidence from priming. *Journal of Experimental Psychology: Learning, Memory and Cognition*, 21:863–883, 1995.

Chandana Paul. *Investigation of Morphology and Control in Biped Locomotion*. PhD thesis, Department of Computer Science, University of Zurich, Switzerland, 2004.

Robert F. Port and Timothy van Gelder, editors. *Mind as Motion: Explorations in the Dynamics of Cognition*. MIT Press, Cambridge, MA, 1995.

D. Roy and E. Reiter. Connecting language to the world. *Art. Intel.*, 167(1–2):1–12, 2005.

Deb Kumar Roy. *Learning from Sights and Sounds: A Computational Model*. PhD thesis, MIT, Media Laboratory, sep 1999.

Stefan Schaal, Auke Ijspeert, and Aude Billard. Computational approaches to motor learning by imitation. *Philosophical Transactions of the Royal Society of London, B*, 358(1431):537–547, 2003.

Candace L. Sidnera, Christopher Leea, Cory D. Kiddb, Neal Lesha, and Charles Richa. Explorations in engagement for humans and robots. *Artificial Intelligence*, 166(1–2):140–164, 2005.

Aaron Sloman and Jackie Chappell. The altricial-precocial spectrum for robots. In L. Pack Kaelbling and A. Saffiotti, editor, *Proceedings of the Nineteenth International Conference on Artificial Intelligence (IJCAI-05)*, pages 1187–1194, Edinburgh, August 2005. Professional Book Center.

Luc Steels and Frederic Kaplan. Bootstrapping grounded word semantics. In T. Briscoe, editor, *Linguistic evolution through language acquisition: formal and computational models*. Cambridge University Press., 1999.

Luc Steels and Paul Vogt. Grounding adaptive language games in robotic agents. In C. Husbands and I. Harvey, editors, *Proceedings of the Fourth European Conference on Artificial Life (ECAL97)*, London, 1997. MIT Press.

Requirements & Designs: Asking Scientific Questions About Architectures

Nick Hawes, Aaron Sloman and Jeremy Wyatt
School of Computer Science
University of Birmingham
{n.a.hawes,a.sloman,j.l.wyatt}@cs.bham.ac.uk

Abstract

This paper discusses our views on the future of the field of cognitive architectures, and how the scientific questions that define it should be addressed. We also report on a set of requirements, and a related architecture design, that we are currently investigating as part of the CoSy project.

1 What Are Architectures?

The first problem we face as researchers in the field of cognitive architectures is defining exactly what we are studying. This is important because the term "architecture" is so widely used in modern technological fields. An agent's cognitive architecture defines the information-processing components within the "mind" of the agent, and how these components are structured in relation to each other. Also, there is a close link between architectures and the mechanisms and representations used within them (where representations can be of many kinds with many functions). Langley and Laird (2002) describe a cognitive architecture as including "those aspects of a cognitive agent that are constant over time and across different application domains". We extend this to explicitly allow architectures to change over time, either by changing connection patterns, or altering the components present. Excluding such changes from the study of architectures may prevent the discussion of the development of architectures for altricial information-processing systems (Sloman and Chappell, 2005).

2 Related Work

Historically, most research into cognitive architectures has been based around specific architectures such as ACT-R, SOAR, and ICARUS (for a summary see (Langley and Laird, 2002)). A lot of work has been devoted to developing iterations of, and extensions to, these architectures, but very little work has been done to compare them, either to each other, or to other possible design options for cognitive architectures. In other words, little work has been done on the general science of designing and building cognitive systems. Anderson and Lebiere (2003) have recently attempted to address this by comparing two different architectures for human cognition on a set of requirements.

3 Architectures & Science

To advance the science of cognitive systems we need two related things: clear, testable questions to ask, and a methodology for asking these questions. The methodology we support is one of studying the space of possible *niches* and *designs* for architectures, rather than single, isolated, designs (Sloman, 1998b). Within such a framework, scientific questions can be asked about how a range of architecture designs relate to sets of requirements, and the manner in which particular designs satisfy particular niches. Without reference to the requirements they were designed to satisfy, architectures can only be evaluated in a conceptual vacuum.

The scientific questions we choose to ask about the space of possible architecture designs should ideally provide information on general capabilities of architectures given a set of requirements. This information may not be particularly useful if it is just a laundry list of instructions for developing a particular architecture for a particular application domain. It will be more useful if we can characterise the space of design options related to a set of requirements, so that future designers can be aware of how the choices they make will affect the overall behaviour of an agent. The questions asked about architectures can be motivated by many sources of information, including competing architecture designs intended for similar niches.

In order for questions about architectures, and their answers, to be interpreted in the same way

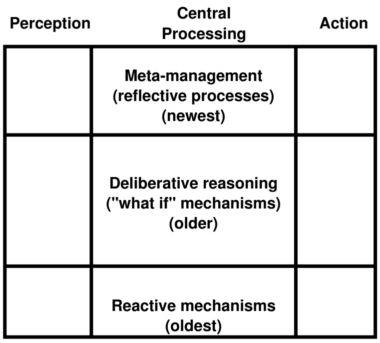


Figure 1: The CogAff Architecture Schema.

by researchers across many disciplines, we need to establish a common vocabulary for the design of information-processing architectures. As a step towards this, we use the CogAff schema, depicted in Figure 1, as an incomplete first draft of an ontology for comparing architectures. (Sloman, 2001). The schema is intended to support broad, two-dimensional, design- and implementation-neutral characterisations of architectural components, based on information-processing style and purpose. If an architecture is described using the schema, then it becomes easier to compare it directly to other architectures described in this way. This will allow differing architectures to be compared along similar lines, even if they initially appear to have little in common.

4 A Minimal Scenario

For our current research as part of the CoSy project[1], we are working from requirements for a pre-linguistic robot that has basic manipulative abilities, and is able to explore both its world and its own functionality. At a later date we will extend this to add requirements for linguistic abilities. We are approaching the problem in this way because we believe that a foundation of action competence is necessary to provide semantics for language. These requirements come from the CoSy *PlayMate scenario*, in which a robot and a human interact with a tabletop of objects to perform various tasks[2].

In our initial work on this scenario we will focus on the requirements related to the architectural elements necessary to support the integration of simple manipulative abilities with a visual system that supports the recognition of basic physical affordances from 3D structure. We see this as the absolute minimum system for the start of an exploration of PlayMate-like issues in an implemented system [3].

Our requirements analysis has led to the design of a prototype architecture which we believe will satisfy the niche they specify. Space restrictions do not permit a full description of the architecture, but in brief the architecture features multiple concurrently active components, including: a motive generator; information stores for currently active motives, general concepts, and instances of the general concepts; a general-purpose deliberative system; a fast global alarm system; a plan execution system; management and meta-management components; a spreading activation substrate; and closely coupled vision and manipulation sub-architectures.

The high-level design for this architecture is presented in Figure 2, and is in part inspired by our previous work on information-processing architectures (e.g. (Beaudoin, 1994; Sloman, 1998a; Hawes, 2004)). Although this design clearly separates functionality into components, these components will be tightly integrated at various levels of abstraction. For example, to enable visual servoing for manipulation (e.g. (Kragic and Christensen, 2003)), visual and proprioceptive perception of the movement of the robot's arm in space must be closely coupled with the instructions sent to the arm's movement controller.

The information-processing behaviour of the architecture is driven by motives, which are generated in response to environmental or informational events. We will allow humans to generate environmental events using a pointing device. The agent will interpret the gestures made with this device as direct indications of desired future states, rather than intentional acts (thus temporarily side-stepping some of the problems of situated human-robot interaction). Generated motives will be added to a collection of current motives, and further reasoning may be necessary if conflicts occur between motives. The deliberative system will produce action plans from motives, and these plans will be turned into arm commands by the plan execution system. This process will be observed at a high level by a meta-management system, and at a low level by an alarm system. The meta-management system may reconfigure the agent's processing strategies if the situation requires it (e.g. by altering the priorities associated with motives). The global alarm system will provide fast changes in behaviour to handle sudden, or particularly critical, situations.

---

[1] See http://www.cognitivesystems.org for more information.

[2] More information about the PlayMate is available at http://www.cs.bham.ac.uk/research/projects/cosy/pm.html.

[3] Our work on requirements from the PlayMate scenario is presented roughly at http://snipurl.com/cosy_playmate.

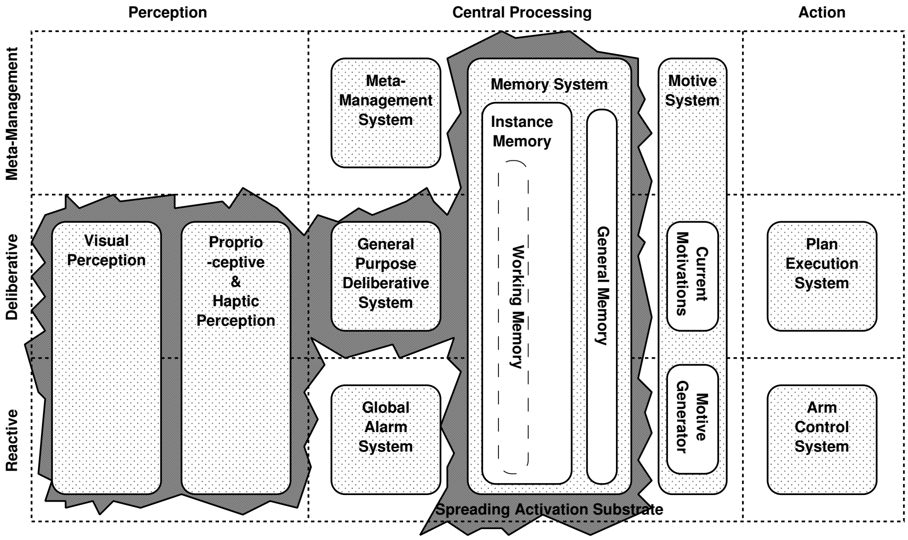


Figure 2: The Proposed Architecture.

Although a spreading activation substrate is featured in the design, we are not currently committed to its inclusion in the final system. Instead we are interested in the kinds of behaviour that such a design choice will facilitate. Information across the architecture may need to be connected to related information, and such an approach may allow the agent to exploit the structure of such connections by spreading activation, which may be based on co-occurrence, recency or saliency. We are also interested in investigating how to combine distributed approaches to representing and processing information with more localised approaches, and what design options this provides. Such a combination of processing approaches can be seen in the work on the MicroPsi agent architecture (Bach, 2005).

5 Architecture Evaluation

The question of whether, and why, our proposed architecture is appropriate for an agent with PlayMate-like requirements is quite hard to formulate in a way that is directly answerable. Instead, we must use our requirements analysis, and our previous experiences of designing architectures for such agents, to derive precise questions and suggest testable hypotheses from this. The following paragraphs present specific questions we could ask about the architecture, and many other architectures.

How can information exchange between architectural components be controlled, and what trade-offs are apparent? For example, should information from visual perception be pushed into a central repository, or should task appropriate information be pulled from vision when necessary, or should this vary depending on the system's information state, goals, the performance characteristics of subsystems, etc.?

What are the relative merits of symbolic and sub-symbolic (e.g. spreading activation) processing methods when applied to collating information across the entire architecture? The proposed spreading activation substrate could interact with various processes and ontologies, and record how information is manipulated. Alternatively, this could be implemented as a central process that must be notified by other processes when certain operations occur. These different approaches could be compared on their proficiency at managing large volumes of multi-modal information, their ability to identify changes of context across the architecture, the difficulty of integrating them with other processes, or the ease with which they facilitate other operations (such as attentional control).

To what degree should the architecture encapsulate modality-specific and process-specific information within the components that are directly concerned with it? Cross-modal application of the early processing results can increase accuracy and efficiency in some processes (c.f. (Roy and Mukherjee, 2005)). In other cases information may be irrelevant, and attempts to apply it across modalities may have the opposite effect whilst increasing the computational load on an architecture. We could explore this notion more generally by asking what types of information should, and should not, be made available by architectural components whilst they are processing it, and what use other architectural components could make of such information.

Given the types of information the architecture will be processing, what are the advantages and disadvantages of having a single central representation into which all information is translated? How do these advantages and disadvantages change when additional processes are added into the architecture?

What role does a global alarm mechanism have in PlayMate-like domains, how much information should it have access to, and how much control should it have? For example, an alarm mechanism may have access to all the information in the architecture and risk being swamped by data, or it may have access to limited information streams and risk being irrelevant in many situations.

Does the architecture need some global method for producing serial behaviour from its many concurrently active components, or will such behaviour just emerge from appropriate inter-component interactions? Approaches to component control include a single central component activating other components, a control cycle in which activity is passed between a small number of components, and other variations on this. Are there particular behaviours that are not achievable by an agent with this kind of control, and only achievable by an agent we decentralised control, or vice versa? If such trade-offs exist, how are the relevant to PlayMate-like scenarios?

Given the range of possible goals that will need to be present in the whole system, how should these goals be distributed across its architecture, and how does this distribution affect the range of behaviours that the system can display?

Obviously there are many other questions we could ask about the architecture, such as whether it will facilitate the implementation of mechanisms for acquiring and using orthogonal recombinable competences[4]. The process of designing and implementing architectures to meet a set of requirements involves the regular re-evaluation of the requirements in light of new developments. Inevitably, this means that other questions will be considered, and the above ones reconsidered, as the research progresses.

Acknowledgements

The research reported on in this paper was supported by the EU FP6 IST Cognitive Systems Integrated project Cognitive Systems for Cognitive Assistants "CoSy" FP6-004250-IP.

References

John R. Anderson and Christian Lebiere. The Newell test for a theory of cognition. *Behavioral & Brain Science*, (26):587–637, 2003.

Joscha Bach. Representations for a complex world: Combining distributed and localist representations for learning and planning. In Stefan Wermter, Günther Palm, and Mark Elshaw, editors, *Biomimetic Neural Learning for Intelligent Robots*, volume 3575 of *Lecture Notes in Computer Science*, pages 265–280. Springer, 2005.

Luc P. Beaudoin. *Goal Processing In Autonomous Agents*. PhD thesis, School of Computer Science, The University of Birmingham, 1994.

Nick Hawes. *Anytime Deliberation for Computer Game Agents*. PhD thesis, School of Computer Science, University of Birmingham, 2004.

Danica Kragic and Henrik I. Christensen. A framework for visual servoing. In Markus Vincze and James L. Crowley, editors, *ICVS-03*, volume 2626 of *LNCS*. Springer Verlag, March 2003.

Pat Langley and John E. Laird. Cognitive architectures: Research issues and challenges. Technical report, Institute for the Study of Learning and Expertise, Palo Alto, CA, 2002.

Deb Roy and Niloy Mukherjee. Towards situated speech understanding: Visual context priming of language models. *Computer Speech and Language*, 19(2):227–248, 2005.

Aaron Sloman. Varieties of affect and the cogaff architecture schema. In *Proceedings of the AISB'01 Symposium on Emotion, Cognition and Affective Computing*, pages 1–10, 2001.

Aaron Sloman. Damasio, descartes, alarms and metamanagement. In *Proceedings of IEEE Conference on Systems, Man, and Cybernetics*, pages 2652–2657, 1998a.

Aaron Sloman. The "semantics" of evolution: Trajectories and trade-offs in design space and niche space. In Helder Coelho, editor, *Progress in Artificial Intelligence, 6th Iberoamerican Conference on AI (IBERAMIA)*, pages 27–38. Springer, Lecture Notes in Artificial Intelligence, Lisbon, October 1998b.

Aaron Sloman and Jackie Chappell. The Altricial-Precocial Spectrum for Robots. In *Proceedings IJCAI'05*, pages 1187–1194, Edinburgh, 2005.

---

[4] http://www.cs.bham.ac.uk/research/projects/cosy/papers/#dp0601

Integration and Decomposition in Cognitive Architecture

John Knapman*
*School of Computer Science
The University of Birmingham, UK
J.M.Knapman@cs.bham.ac.uk

Abstract

Given the limitations of human researchers' minds, it is necessary to decompose systems and then address the problem of how to integrate at some level of abstraction. Connectionism and numerical methods need to be combined with symbolic processing, with the emphasis on scaling to large numbers of competencies and knowledge sources and to large state spaces. A proposal is briefly outlined that uses overlapping oscillations in a 3-D grid to address disparate problems. Two selected problems are the use of analogy in commercial software evolution and the analysis of medical images.

1 Introduction

Debates about cognitive architecture often deal with choices between rival techniques and modalities. By contrast, Minsky, Singh and Sloman (2004) emphasise the importance of integrating components of differing kinds.

A standard method in designing complex systems is to decompose into components and then connect them, simply because it is impossible for individual team members to hold all the detail in their heads. Because of such limitations, there are those who think that it is not possible for human beings to design systems that exhibit general intelligence comparable to their own (Rees, 2003). Even after fifty years of research, such counsels of despair are premature until we have explored more of the possibilities. The prizes are great, in part for a better understanding of ourselves and the enlightenment it will bring in the tradition of the dazzling human progress since the Renaissance in many fields, including medicine and technology. There are also strong commercial benefits in being able to build smarter systems that can undertake dangerous or unpopular work.

To make substantial progress, there are lessons to learn from work on the architecture of computer systems generally (Bass, Clements and Kazman, 2003), which informs us that a key responsibility of the architect is to define how the components fit together and interface with each other. This assumes that there is a degree of uniformity in the style of the components, so that their interfaces can be defined in a form that is commonly understood by the people working on them. Interfaces are then specified in terms, firstly of timing and flow of control (serial or parallel, hierarchical or autonomous, event driven or scheduled, for example), and secondly in terms of the format of data flowing between components.

A less well documented but nevertheless well known experience is that a small team of dedicated experts can achieve wonders compared with large teams of people with mixed ability. A small team of four highly experienced people can often produce efficient, reliable and timely systems that solve problems most effectively. By contrast, many research (and development) teams consist of one experienced person, who is very busy, and several bright but inexpert assistants. To get an expert team together to work full time and hands on for a period of years is expensive and disrupts other activities, but the results can be very exciting.

Even with such a promising kind of team organisation, interfaces have to be defined. In cognitive architecture, components will be of differing kinds. Connectionist methods that emphasise learning and convergence must somehow be combined effectively with symbolic processing. Sun (2002) shows how symbolic rules may be inferred from state transitions in a connectionist network, but that is only one of several ways in which interactions could take place. There are other promising numerical methods, such as KDDA, which has been applied successfully to face recognition (Wu, Kittler, Yang, Messer and Wang, 2004).

The strong emphasis on learning and learnability in connectionist methods is carried over to symbolic rules in Sun's method, but there are other benefits from connectionism, such as approximate matching and graceful degradation, that need to be exploited in a combined system. Benefits like these accrue not just from connectionism but from other soft computing paradigms (see Barnden (1994) for a good discussion in the context of the study of analogy).

One method for linking components that use quite different representations from differing standpoints is to use numerical factors, whether these are interpreted as probabilities, fuzzy or other uncertainty values, ad hoc weights, or arrays of affective measures (e.g., motivation (Coddington and Luck, 2003), duty, elegance). Such techniques can also help to reduce search spaces and large state spaces, although they may sometimes smack of heuristics of the kind favoured by researchers in the 1970s.

2 Abstraction

Many different kinds of abstraction have been identified. In the field of computer system design, there are methods with well-defined levels of abstraction, the most concrete being a set of programs that implements a design. In the B and Z methods (Bert, Bowen, King and Waldén, 2003) there are formal proof procedures to show that a more concrete specification is a refinement of one that is more abstract.

Less formal methods, such as the Unified Software Development Method (Jacobson, Booch and Rumbaugh, 1999), based on the Unified Modeling Language (UML) still have the idea that a high-level design (as an object diagram) can be refined progressively until implementation. Beyond the writing of the programs, a more complete analysis sees the programs and their specifications as abstractions of their actual performance, as recorded in traces while they execute. The diagrams are found to be better for communicating between designers and developers than either formal-language statements or natural-language descriptions.

Even though there is a clear definition of abstraction in these examples, users often feel intuitively that the diagrams are less abstract than the text. Such an intuition throws up the difficulty of defining abstraction in a general way. It seems to depend on fitness for purpose as well as brevity and omission of detail.

In the case of a story told in a natural language, a synopsis is more abstract in the latter sense. In the case of scientific papers, the “abstract” is intended to help readers decide whether to read the main paper. The “management summary” of a business report allows busy senior people to understand enough to be able to trust and defend their subordinates’ recommendations.

A reference to “the report”, “the story” or “the paper” is clearly more abstract than having to repeat the content. Generally, referring to something verbally or symbolically is brief, enables it to be dealt with in its absence, and permits economy of thought, i.e., it leaves mental room, as it were, for other concepts to be introduced and related to it.

Uncertainty representations provide another form of abstraction, and one that is particularly easy to formalise (Baldwin, Martin and Pilsworth, 1995). A fuzzy value can stand for “the hand-written letter that is probably a k”, or “the car that looked like a Jaguar”. Such representations are the clearest candidate for a form of abstraction that can be used to enable disparate sources of knowledge to be combined effectively and rapidly. For example, a moving picture, some sounds, previous experience of steam trains and expectation that the Flying Scotsman will pass through the station at 11:00 a.m. combine so as to interpret the distant approach of the train while ignoring most other details.

3 Large State Spaces

Problems often entail large numbers of possibilities. Although both abstractions and numerical methods can help to reduce the possibilities, there are frequently cases where many possible states must be carried forward before higher abstractions can be used to eliminate some. Sometimes called the “AI-complete” problem, there have been hopes that quantum information processing could address it. However, the breakthrough has not so far come. Apart from special cases where the data has cyclic properties (as in modal arithmetic for code breaking), the main benefit is that large state spaces (having N states) can be searched in time proportional to $\sqrt{N}$ instead of $N/2$. This can be worthwhile in some cases (e.g., reducing 500 billion steps to one million), but must await the availability of suitable hardware. The programming skills required are rather daunting.

Some people have suggested that the unstable periodic orbits (UPOs) of chaotic oscillators (Crook and Scheper, 2001) can represent potentially infinitely many things. It has been observed that the signals in a brain appear to be either random or chaotic before settling rapidly to a coherent state (Tsui and Jones, 1999). In theory, a random signal contains all possible frequencies, but a practical random signal is limited to the bandwidth of the channel and takes a long time to distinguish from a sum of many overlapping signals of different frequencies, an idea taken up in the Proposal section below. A chaotic signal is somewhere in between, and is also indistinguishable in practice (Gammaitoni, Hänggi, Jung and Marchesoni, 1998) from the other two.

4 Requirements

A test bed for the exploration of ideas is needed that supports the following requirements:

1. Combining modalities, especially connectionist, symbolic and affective
2. Combining competencies, including (but not limited to) analogy and structure matching, vision and formal language interpretation
3. Abstraction, with emphasis on combining knowledge sources
4. Scaling to large state spaces, particularly exploring efficient forms of parallelism
5. Scaling to large numbers of knowledge sources

5 Proposal

To explore these issues and requirements, one approach is to design and simulate a programmable signal-processing network capable of both symbolic and connectionist processing, with commitment to as few preconceptions as possible.

A flexible structure is proposed that will allow for the interplay of several loose decompositions. We will allow an element to belong to several groupings, which can be nested. It must identify with which grouping any particular communication is associated. With such a protocol, elements are not confined to a hierarchical organisation, but hierarchies are still possible, and communication channels are more manageable than in a complete free for all.

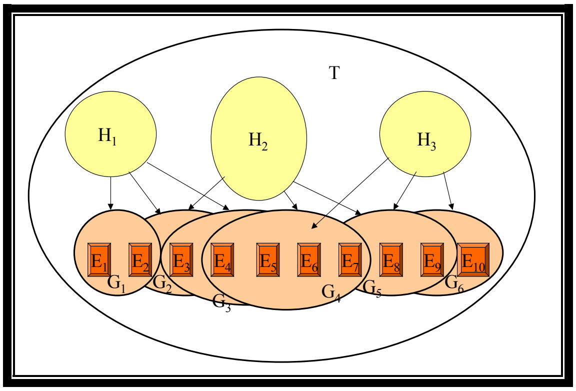


Figure 1: Illustration of Flexible Structuring – containment is shown by nesting or with arrows

A message between two elements has to conform to the representational format of their common grouping at some level of nesting. Then, if the prime method of communication is broadcasting, for example, broadcasting (both sending and receiving) would only take place within a grouping and would follow the representational convention for that grouping. In the illustration of Figure 1, E_1 and E_2 can communicate by the conventions of their containing groupings G_1, H_1 and T. E_1 can communicate with E_3, E_4, E_5 and E_6 by the conventions of H_1 and T, because these four are in G_3, which is in H_1. E_1 can only communicate with E_{10} by the conventions of T or by some form of relay through intermediate subordinate groupings.

Within such a general framework, it is proposed to exploit the parallelism inherent in modulated overlapping signals of many frequencies embedded in a 3-D grid. One such model is described by Coward (2004) as an attempt to emulate aspects of the architecture of the brain. Uncertainty models, including Bayesian nets (Pearl, 1988) and fuzzy logic, are to be accommodated. It must be suitable for both learning and programmed behaviour. Programming provides the flexibility to explore challenging applications, and it allows certain abilities to be built in. For other capabilities, there should always be at least the possibility that the symbols and mappings defined could be acquired through experience or by an indirect process such as analogy, deduction or abstraction.

There must be support for widely differing data types, particularly for image processing and formal language processing. The particular problems under consideration are in two domains:

1. The application of analogical reasoning to commercial software evolution
2. Analysis of medical images

It must be reasonably clear how the framework may be extended to other modalities, e.g., movement control for vehicles or robots.

A particularly elegant form of programming is functional programming, where every program is a transformation from the input parameters to an output value. A function is defined in mathematics as a set of ordered pairs of input and output. Most programmers think of procedures as behaving algorithmically, but the alternative definition sees a function as a transformation from input to output via memory lookup. The effect of a procedure giving multiple results can be achieved by multiple functions that take the same parameters. Functions of several variables can be decomposed into (single valued) functions of pairs of variables.

This view of a function is particularly convenient for the parallel processing of overlapping signals of many frequencies. Such signals can encode ranges or uncertainty in data but can also represent patterns of input, such as image intensities or characters in text.

Frequency can be used to encode data values, with amplitude representing strength. A transformation converts from an input frequency to an output frequency and may adjust the weight. Thus it may transform one pattern to another or may perform simultaneous logical operations on parallel data. The transformation of patterns using weight adjustment can be equivalent to that performed in connectionist networks, with a natural mechanism for incorporating learning. Logical operations may not need to perform weight adjustment, but conjunction and disjunction between two sets of signals are needed. Disjunction can be achieved by summing. Conjunction requires frequency matching.

Some programming models require a global state, for example the query and subgoals in PROLOG, whereas object-oriented (OO) programming localises the state information in separate objects.

A 3-D grid can contain many channels, and the signals can persist for some time, somewhat in the manner of objects in OO programming, though with different dynamics. Together they may encode a very large state space, even though there is a limit to the number of signals that can be carried on one channel because of the constraints of bandwidth. They are well suited to representing data structures such as parse trees or image region classifications.

The interactions are different from those in quantum computing; there, each state is completely integrated but is processed in isolation from all others. In the 3-D grid, a state is distributed, but information from many states can be mixed.

6 Conclusion

After half a century's work, there remain many ideas that can be explored, particularly in the arena of integration. It would be most exciting to see what a dedicated team of four or five experts able to work full time for a period of years could achieve. However, the challenge remains of discovering a small enough set of representations that are general enough for interfacing between the kinds and styles of components identified but are nevertheless succinct enough to be computationally tractable.

Acknowledgements

I'm grateful for valuable discussions with Aaron Sloman.

References

J.F. Baldwin, T.P. Martin and B.W. Pilsworth. *Fril – Fuzzy and Evidential Reasoning in Artificial Intelligence*, Research Studies Press, Taunton, UK and Wiley, New York, 1995, p.54

John Barnden. On Using Analogy to Reconcile Connections and Symbols, in Levine, D.S. and Aparicio, M. (eds.) *Neural Networks for Knowledge Representation and Inference*, Hillsdale, New Jersey: Erlbaum 27-64, 1994

Didier Bert, Jonathan Bowen, Steve King, Marina Waldén (eds.) *ZB 2003: Formal Specification and Development in Z and B: Third International Conference of B and Z Users*, Turku, Finland, June 4-6, LNCS 2651, Springer, 2003

Len Bass, Paul Clements and Rick Kazman. *Software Architecture in Practice (2nd edition)*, Addison-Wesley, 2003

Alexandra Coddington and Michael Luck. Towards Motivation-based Plan Evaluation, in *Proceedings of the Sixteenth International FLAIRS Conference (FLAIRS '03)*, Russell, I. and Haller, S. (eds.), AAAI Press, 298-302, 2003

L. Andrew Coward. The Recommendation Architecture Model for Human Cognition. *Proceedings of the Conference on Brain Inspired Cognitive Systems*, University of Stirling, Scotland, 2004

Nigel Crook and Tjeerd olde Scheper. A Novel Chaotic Neural Network Architecture, *ESANN'2001 proceedings – European Symposium on Artificial Neural Networks*, Bruges, Belgium, 25-27 April 2001, D-Facto public., ISBN 2-930307-01-3, pp. 295-300, 2001

Luca Gammaitoni, Peter Hänggi, Peter Jung and Fabio Marchesoni. Stochastic resonance, *Reviews of Modern Physics*, **70**(1), 270-4, 1998

Ivar Jacobson, Grady Booch and James Rumbaugh. *The Unified Software Development Process*, Addison Wesley Longman, 1999

Marvin Minsky, Push Singh and Aaron Sloman. The St. Thomas Common Sense Symposium: Designing Architectures for Human-Level Intelligence, *AI Magazine*, **25**(2), 113-124, 2004

Judea Pearl. *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*, Morgan Kaufmann, San Mateo, California, 1988

Martin Rees. *Our Final Century*, William Heinemann, London, p.134, 2003

Ron Sun. *Duality of the Mind: A Bottom Up Approach Toward Cognition*, Mahwah, New Jersey: Erlbaum, 2002

Alban Pui Man Tsui and Antonia Jones. Periodic response to external stimulation of a chaotic neural network with delayed feedback, *International Journal of Bifurcation and Chaos*, 9(4), 713-722, 1999

Wu, X.J., Kittler, J., Yang, J.Y., Messer, K. and Wang, S.T. A New Kernel Discriminant Analysis (KDDA) Algorithm for Face Recognition, *Proceedings of the British Machine Vision Conference 2004*, Kingston, UK, 517-526

On the structure of the mind

Maria Petrou and Roberta Piroddi
Imperial College London
Department of Electrical and Electronic Engineering, Exhibition Road, London SW7 2AZ
{maria.petrou,r.piroddi}@imperial.ac.uk

Abstract

The focus of any attempt to create an artificial brain and mind should reside in the dynamic model of the network of information. The study of biological networks has progressed enormously in recent years. It is an intriguing possibility that the architecture of representation and exchange of information at high level closely resembles that of neurons. Taking this hypothesis into account, we designed an experiment, concerning the way ideas are organised according to human perception. The experiment is divided into two parts: a visual task and a verbal task. A network of ideas was constructed using the results of the experiment. Statistical analysis showed that the verbally invoked network has the same topological structure as the visually invoked one, but the two networks are distinct.

1 Introduction

The key to reproducing intelligence and consciousness lies in the understanding and modelling of the organisation of functional areas, the privileged pathways of communication and exchange of information, and their more likely evolution and growth, as in Aleksander (2005). The heart of a biomimetic robot is the pump that sustains the blood-flow of information. The information comes from a number of functional parts which do not necessarily need to be bioinspired: it is possible to produce depth maps without knowing how the visual system produces them. What is important to know is how the depth map is used for navigation and in how many other different tasks it is used. The organisation and evolution of the information, as well as the fusion of pieces of information of varied degrees of uncertainity, are characteristics of a living being and this is what needs to be emulated directly from a biological system.

The Project of the Human Genome Consortium (2003) culminated with the completion of the full human genome sequence in April 2003. Now the research moves forwards, after having named and counted genes. Understanding the organisation and interaction of genes is the new frontier of biotechnology. This endeavour is helped by the development by Barabasi (2002) of a new science of networks, which sheds light into the complexity of organisation of living and artificial mechanisms alike.

The focus of this century will be the brain. Many researchers have already moved to the race for mapping the human brain neurons and understanding their functionalities, for example in Koslow and Hyman (2000). However, neurons are just one element. What about that special products of the human brain, the ideas? Is it possible to gain insight into the world of the ideas? Is it possible to count them? Are ideas organised in a recognisable way?

In the article by Macer (2002) the Behaviourome project was first proposed. In analogy with the genome project aimed at mapping the human genome, here the aim was to count ideas and to find out whether the number of ideas is finite, uncountable or infinite. One of the proposed means of obtaining the final goal was to provide a mental mapping of ideas and their interrelationships.

The organisation of information may be modelled in mathematical terms as a dynamic network. We are intrigued by the possibility that from the lowest possible level of information sources, namely the neurons, to the highest possible level of information products, the ideas, the same structural network may be underling the architectural building of their organisation. Networks have been used for a long time to represent complex systems. They have been popularised recently by Barabasi (2002), who studied self-organising networks. Self-organising networks may be random or scale-free. There is evidence that the organising architecture of a scale-free network underpins the structure and evolution of biological systems (like cells), social systems (like one's circle of friends) and artificial networks (like the Internet).

We designed an experiment in order to study how higher level information invoked by different stimuli, namely visual and verbal, is organised in the mind. The importance of such a speculative attempt lies in the fact that if a symmetry between lower and higher levels is found, then the same mathematical model may be used to bridge the gap between top-down and bottom-up approaches to create artificial brains and minds.

2 Ideas and networks

In order to proceed in our experiment we need to focus on two elements: ideas and networks. Ideas (and their relationships) are the subject of this research. We need to define them and to highlight some elements of the disciplines that study their relationships.

According to Macer (2002) ideas are mental conceptualisations of things, including physical objects, actions or sensory experiences, that may or may not be linguistically expressible.

Self-organising networks, occurring in the natural world and as a development of human activities, may be modelled as being random or scale-free. Both scale-free networks and random networks are *small world networks*, which means that, although the networks may contain billions of nodes, it takes only a few intermediate nodes to move from one node to any other. Let us indicate by k the number of incoming or outgoing links from any node in the network. The difference between a random and a scale-free network is quantified by the probability density function $P(k)$ of a node having k incoming or outgoing links. In the case of a random network, $P(k)$ has a Poisson distribution, $P_1(k) \sim \exp(-k)$. In the case of a scale-free network, the probability may be modelled by a power function, $P_2(k) \sim k^{-c}$, where c is a positive constant.

3 Experimental methodology

We often see something which triggers some thought, which triggers another thought and so on. We colloquially refer to this as a flow of thoughts. We may try to stimulate such a trail of thoughts by showing an image or mentioning a word to a person. Each person will generate their own path of linked thoughts. The collection of ideas that have been thought may be visualised as a network of ideas and analysed as such. So the questions we would like to answer are:

1. How many ideas on average exist between any two randomly chosen ideas?
2. Does this number depend on whether the stimuli for these ideas are presented to a person in a visual or in a linguistic way?
3. If a network of ideas can be generated, which is its topology?

There are serious difficulties in the design of an experiment in order to answer the above questions. The most important difficulties are:

1. How does one define an idea?
2. How does one deal with the enormous number of different ideas?
3. How can one expose a subject to a collection of ideas?
4. How does one count the intermediate ideas between any two ideas?

The brief answers to the above questions are as follows.

1. Since we wish to show the ideas in the form of an image in the visual experiment, we need to restrict ourselves to an idea being an object, e.g. umbrella, flower, car, sun, etc.
2. It is obvious that we need to restrict ourselves to a finite number of ideas. We decided to use 100 nouns. There were various reasons for that, mainly practical, related to the way the stimuli had to be presented to the subjects. The nouns had to be chosen in an objective way, so as not to reflect any prejudices or associations of the investigators. We chose them to be words uniformly spread in the pages of a dictionary.
3. We showed to each subject the full collection of ideas as a matrix of 10×10 images arranged in an $A0$ size poster. Each noun was represented by a clip-art from the collection of Microsoft Office. This ensured that effort had gone into the design of the images so that they were most expressive for the particular object they were supposed to depict. Before use, each image was converted into grey and it was modified so that the average grey value of all images was the same. This ensured that no image would be picked out because of its excessive brightness or darkness.
4. A picture from the collection was picked at random and then the subject was asked to pick the next most similar one, and then the next most similar one and so on. Every time an object was picked after another one, a link was recorded between these two objects.

We conducted the experiments on a sample of 20 subjects. The sequence of actions that constituted the visual experiment are listed below.

- Allow the subject at the beginning to see the whole table of objects for several minutes, to familiarise themselves with what is included in the restricted world.
- Pick up one of the objects at random and highlight it by putting a frame around it. Ask the subject which of the remaining objects is most similar to this. Once the subject has made their choice, cover the highlighted object, highlight the object they had picked and ask them to pick the next most relevant object. Stop when half of the objects are still visible. This ensures that the subject has still plenty of choice when they make their choice of the last object. The subject may be allowed to say that an object has no relation to any other object in the table. Then the particular experiment may stop. Another experiment may follow starting with a different initial object.
- The position of the objects should not be changed during the experiment, as we wish the person to know what is included in the restricted world in which they have to make their choice.

The verbal experiment was aimed at identifying how many intermediate ideas exist between any two ideas presented in a verbal form. This experiment took place several weeks after the visual experiment. The same subjects took part in both experiments. Each subject was presented with a table containing 100 words corresponding to the 100 pictures presented in the visual experiment. The experiment followed these steps:

- Allow the subject at the beginning to see the whole table of words for several minutes, to familiarise themselves with what is included in the restricted world.
- Pick up one of the words at random and highlight it by putting a frame around it. Ask the subject which of the remaining words is most similar to this. Once the subject has made their choice, cover the highlighted word and ask them to pick the next most relevant word. Stop when half of the words are still available. The subject may be allowed to say that a word has no relation to any other word in the table. Then the particular experiment may stop. Another experiment may follow starting with a different initial word.
- The position of the words should not be changed during the experiment, as we wish the person to know what is included in the restricted world in which they have to make their choice.

4 Experimental results

The first point to investigate is whether there are ideas that are selected more frequently than others, i.e. whether there are ideas that are more popular or spring into mind more often, or all ideas are generally selected with the same frequency.

ID	Idea	Frequency	# Connections
42	graduate	16	15
29	doctor	12	11
99	wedding	14	11
2	airport	12	9
33	electricity-mast	14	9
96	tree	14	9
90	teacher	16	8

Table 1: Ideas that manifest a *hub* behaviour in the visual network.

From sociological studies reported by Gladwell (2000), we know that such nodes act as *hubs* and play an important role in a network. We found that only 7% of the ideas are both very frequent and very well connected. This 7% of ideas are connected to 60% of the remaining ideas for the visual experiment, presented in table 1, and to 50% of the remaining ideas for the verbal experiment, presented in table 2. These numbers show evidence of a small world behaviour for the networks of ideas we have built.

ID	Idea	Frequency	# Connections
7	bicycle	8	10
22	circus	7	12
48	house	10	12
69	rain	7	11
71	road	7	9
83	shopping	7	9
99	wedding	8	11

Table 2: Ideas that manifest a *hub* behaviour in the verbal network.

The number of intermediate ideas between any two given ones may be investigated using two measures. The first measure is the *average path*, defined as the average length of the path between any two nodes, if such a path exists. To calculate it, one has first to calculate the *average* distance between any two given nodes and then average all these distances over the whole network. We denote this measure by $\bar{a}$. The second measure is called *mean path*. First, one has to find the *shortest path* between any two given nodes, if there is any. Then the shortest paths are averaged over the entire network. We denote this measure by $\bar{m}$.

The second research question was to discover whether the number of intermediate ideas between any two given ones depends on the way the stimuli are presented to the subject. In table 3 we list the path length characteristics for the two experiments. Taking into consideration the standard deviations of these measures, one notices that the path lengths are statistically the same, irrespective of the method used for hint giving.

Experiment	$\bar{a}$	σ_a	$\bar{m}$	σ_m
Visual experiment	13.3	1.9	11.5	3.5
Verbal experiment	13.9	1.7	12.6	4.2

Table 3: Summary of path length measures.

However, the ideas which act as hubs in the two cases are different, as one may see by comparing tables 1 and 2. This strongly indicates that the ideas are organised in two different networks, one verbal and one visual, which, however, exhibit similar topologies.

To further examine this topology we use the degree density $P(k)$, as this is a measure that characterises a network independently from the number of its nodes. We test here for the following null hypotheses:

- H_1: The data have been drawn from a population with Poisson distribution $P_1(k)$.
- H_2: The data have been drawn from a population with power law distribution $P_2(k)$.

Our analysis showed that we may reject the H_1 null hypothesis at 95% confidence level of rejection, while hypothesis H_2 is compatible with our data.

5 Conclusions

The results show:

- a small world behaviour of the networks obtained both by visual and verbal cues, indicated by the low mean path and low clustering coefficient values;
- a correspondence between the visual and verbal network in the value of the mean path and the value of the possible number of hubs;
- a correspondence in the topology of the networks, the two networks being statistically equivalent in topology;
- a difference between the visual and verbal networks indicated by the concepts that acted as hubs in the two networks;
- evidence that the networks are organised as scale-free ones.

If the mind organises itself in the form of networks, it appears that these networks are strongly influenced by the hardware on which the mind resides, i.e. the brain, otherwise we should not have a different network for the flow of ideas when visual stimuli are used, from that created when verbal stimuli are used. It is possible, therefore, to hypothesise the existence of a hierarchy of organisation of information which flows from the lowest level of electro-chemical information of the neurons, to the highest conceptual abstractions of ideas. The hierarchy is supported by the same underling network framework, which is one of scale-free topology. The network represents the hardware of the organisation, dictating which dynamical modification is plausible in the evolution of the information. What we should look for, in order to create a link between top-down and bottom-up approaches, is a mapping between neural structures and their higher level correlates.

References

I. Aleksander, *The World in my Mind, My Mind in the World: Key Mechanisms of Consciousness in people, Animals and Machines*, Academic Press, 2005.

L. Barabasi, *Linked: The New Science of Networks*, Perseus Publishing, 2002.

M. Gladwell, *The tipping point. How little things can make a Big difference.*, Little, Brown and Company, 2000.

International Human Genome Sequencing Consortium, “Initial sequencing and analysis of the human genome,” *Nature*, vol. 409, no. 6822, pp. 860–922, 2003.

S. Koslow and S. Hyman, “Human brain project: A program for the new millennium,” *Einstein Quarterly Journal of Biology and Medicine*, vol. 17, pp. 7:15, 2000.

D.R.J. Macer, “The next challenge is to map the human mind,” *Nature*, vol. 402, pp. 121, 2002.

Social Learning and the Brain

Mark A. Wood*
*Artificial models of natural Intelligence (AmonI)
University of Bath, Department of Computer Science
Bath, BA2 7AY, UK
`cspmaw@cs.bath.ac.uk`

Abstract

Social learning is an important source of human knowledge, and the degree to which we do it sets us apart from other animals. In this short paper, I examine the role of social learning as part of a complete agent, identify what makes it possible and what additional functionality is needed. I do this with reference to COIL, a working model of imitation learning.

1 Building a Brain

The problem of building a brain is one facing me at this very juncture in my research. I need a brain capable of controlling indefinitely a complete agent functioning in the virtual world of *Unreal Tournament* (*UT*) (Digital Extremes, 1999). As a game domain, clearly UT is not an exact replica of the real world, and much is simplified or omitted altogether. However, it does provide an opportunity to study a very broad range of human behavioural problems at a tractable level of complexity, as opposed to other more realistic platforms which allow only the study of narrow classes of problems.

My research thus far has chiefly been in the area of social learning (particularly imitation), as I believe this is key to survival in a world where there are unfortunate consequences if things are not learned quickly enough. We humans also dedicate a vast amount of brain space to learning, and social learning in particular, compared to other species. In the following section, I will explain what I think the role of social learning is and why it is important. I will then briefly overview COIL, a model extending CELL (Roy and Pentland, 2002) from language learning to social learning in general. I describe both what COIL requires to function and how it would be extended and complemented to form a complete brain system. I conclude with a discussion.

2 The Role of Social Learning

Human infants seem to be innately programmed to imitate from birth (Meltzoff and Moore, 1983). Many animals, particularly the great apes, benefit from similar kinds of social learning mechanisms (Byrne and Russon, 1998), but none to the extent that we do. The speed and accuracy with which we can assimilate goal-directed (ie. task-related) behaviour from others is unique. Of course, communicating via language and the ability to reproduce actions at fine temporal granularity are among the human-specific skills which facilitate this learning. Taking these things into consideration, it would be wise to consider including social learning capabilities in any system designed to function as a complete brain.

Furthermore, autonomous agents need skills: whether ‘basic’, low-level skills such as co-ordinating motor control, or ‘complex’, high-level skills such as navigation. To acquire task-related skills at any level, I believe there are at least four types of things which need to be learned (Bryson and Wood, 2004, see also Figure 1):

1. *perceptual classes:* What contexts are relevant to selecting appropriate actions.
2. *salient actions:* What sort of actions are likely to solve a problem.
3. *perception/action pairings:* Which actions are appropriate in which salient contexts.
4. *ordering of pairings:* It is possible that more than one salient perceptual class is present at the same time. In this case, an agent needs to know which one is most important to attend to in order to select the next appropriate action.

Some of these may be innate, but those which are not must be acquired using a combination of individual and social learning. For example, assume we

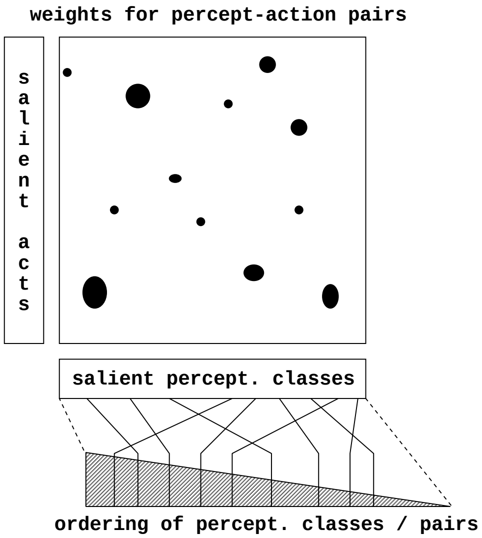


Figure 1: Task learning requires learning four types of things: relevant categories of actions, relevant categories of perceptual contexts, associations between these, and a prioritized ordering of the pairings. Assuming there is no more than one action per perceptual class, ordering the perceptual classes is sufficient to order the pairs.

have an agent which can issue motor commands, but does not initially know the results these commands will have on its effectors. Using visual and proprioceptive sensors (say) to measure these effects, and trial-and-error (individual) learning, a mapping between commands issued and effects produced can be created. This example is deliberately analagous to human infant ‘body babbling’ (Meltzoff and Moore, 1997). However, assuming a reasonable number of ‘primitive’ actions can be learned this way, the set of skills that can be built from these blocks is exponentially larger (and so on as more skills are acquired). To attempt to learn all skills through trial-and-error, then, would be to search randomly through these huge, unconstrained skill spaces — very inefficient.

Social learning can take many forms depending upon the nature of the agents in question: written or verbal instruction, explicit demonstration, implicit imitation, etc. An agent which is part of a society which facilitates such learning can take advantage of the knowledge acquired by previous generations. To do this an agent must be able to relate what it perceives to the actions it can execute; it must solve a correspondence problem between the instruction or demonstration and it’s own embodiment (Nehaniv and Dautenhahn, 2002). For a learning agent in a society of conspecifics, this mapping is simple (although not trivial to learn), but for, say, a robot living among humans, solving this problem amounts to yet another skill that needs to be mastered. Socially-acquired skill-related knowledge can be used to significantly reduce the skill search space, allowing individual learning to merely ‘fine-tune’ new skills, taking into account individual variability within a society. The other alternative is that the ‘instructions’ acquired are coarse-grained enough to perfectly match existing segments of behaviour in the learner’s repertoire.

3 Necessary Components

To better understand the components required for social learning in general, it makes sense to examine the information requirements of a model which is capable of such learning. The Cross-Channel Observation and Imitation Learning or COIL model of Wood and Bryson (2005) is suitable. This system is designed to observe via virtual sensors a conspecific agent executing a task, then in real-time output a self-executable representation of the behaviour needed to complete that task. It achieves this by matching the observed actions of this task *expert* with its observed perceptions of the environment. I now briefly explain the model and identify in general terms what is needed at each stage of processing (see also Figure 2).

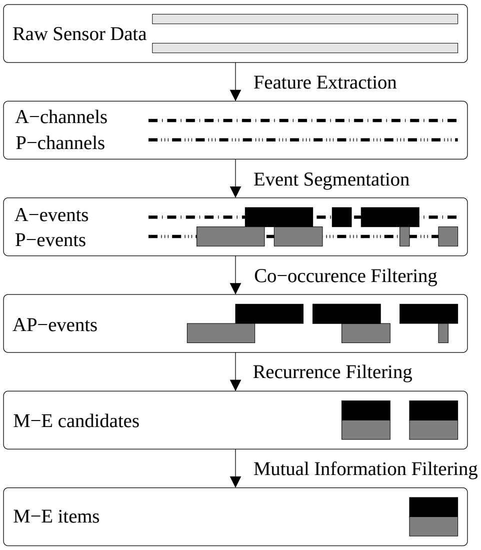


Figure 2: An overview of COIL.

Feature Extraction The inputs to this stage are raw sensory data. Depending upon the task, some of these data are discarded and others are recoded or categorised. Remaining data are diverted into different channels ready for further processing, some specialising in action recognition and others in environmental parsing (perception). This stage therefore suggests a need for selective **attention**, compressed **representations** and **modularity** of processing.

Event Segmentation Following Feature Extraction, the channels containing the data are segmented into action and perception events depending upon the channel type. Events define high-level coarse-grained actions and perceptual classes, and are further divided into lower-level fine-grained subevents. This segmentation requires various **triggers** which are innate in the case of COIL, but could theoretically be learned.

Co-occurence Filtering Action and perception events which overlap in time are paired together and stored in a buffer (called Short Term Memory or STM). This requires **temporal reasoning** and **memory**.

Recurrence Filtering Co-occuring action and perception subevents which are repeated within the brief temporal window of STM are tagged. A chunk called a Motivation-Expectation or M-E Candidate, which represents the set of tagged pairs, is created and placed in Mid-Term Memory (MTM). Here we additionally use **statistical reasoning** and abstract judgments of **similarity**.

Mutual Information Filtering For each M-E Candidate, the maximum mutual information between its component action and perception subevents is calculated. Those which exceed some threshold are stored as M-E Items in Long Term Memory (LTM). COIL currently uses fixed **thresholds**, but again they could be acquired through experience. The LTM is the output of the system.

The innate skills which are necessary for social learning identified above can be provided by the hardware (memory, clock, etc.) and software (statistical algorithms, similarity metrics, etc.) of the agent.

4 Scaffolding COIL

I have looked at the basic components COIL needs in order to function as a social learning system. However, the extent of COIL's role within a complete agent, and the extra pieces which need to be added, remain in question.

There are a number of problems in assuming that a single monolithic COIL system can alone act as the 'brain' of our agent. Firstly the algorithm only learns – it has no capacity for making decisions or acting based upon what it has learned. Our most recent work demonstrates the addition of an extra module for exactly those purposes (Wood and Bryson, 2005). Secondly, a flat COIL system expected to carry out the high-level task of *life* would have to monitor every action and perception channel that could possibly be useful in achieving this task, or any of its subtasks. Even with the innate attentional capabilities of COIL's Feature Extraction stage, the algorithm's complexity is still exponential in the number of channels. Therefore, COIL seems more suited to learning local specialised tasks where the number of channels which need to be monitored can be reasonably constrained.

Let us assume instead that we have a number of COIL systems, each observing a localised task and its associated action / perception channels. We would need a method for discerning which of the following four scenarios is occuring:

1. A known task is being observed.
2. A known task is present[1].
3. An unknown task is being observed.
4. No known tasks are present or being observed.

It may be that scenarios 1 and 2 occur concurrently, in which case a decision would need to be made whether to observe and learn or join in with the execution of the task. Also, the presence of a task does not necessarily imply that the task should be executed. In scenario 4, social learning is impossible, and the product of previous social learning episodes is not applicable. This is where an individual learning module would come into play (see also Section 5).

A complete system that is capable of this arbitration is as yet unrealised. It may be possible to create a 'master' version of COIL which has high-level perception channels monitoring those environmental states which differentiate between local tasks, and action channels which cause lower-level task-specific COILs to be activated. On the other hand, a totally different system designed specifically to coordinate COILs (for social learning), RL (for independent learning) and acting may be more appropriate.

---

[1] A task is present if it is available in the environment for execution by the observer.

5 Discussion

In this final section, I highlight a number of research problems, some of which I will be investigating in relation to the COIL project, but all of which I believe will need to be studied before a complete working brain becomes a possibility. The balance between what is innate and what is learned, for both biological and theoretical robotic examples, has been discussed by Sloman and Chappell (2005). They claim that using a hybrid of the two may prove to be better than using either in isolation. We can further subdivide that which is learned into that which is learned socially, and that which is learned independently. Similarly, the best technique is probably to combine the two, and it is a thorough study of the balance and application of both that may result in significant progress toward constructing a complete agent. This task has at least the following component questions:

- What structures must be present to make independent learning possible? Presumably many of these will be innate, but how can social learning improve these structures / primitives and / or the efficiency of their usage (ie. learning how to learn better from another)?
- What primitives are needed to make social learning a possibility? Must they be acquired through trial-and-error learning, or can they be innate? If acquired, what is the cost of such acquisition? How do they differ from those required for individual learning?
- How does the embodiment of a given agent affect the structures / primitives best suited for both individual and social learning (both *what* is learned and the *way* it is learned)? How does this compare / interact with the affect the required tasks have on these primitives?
- How can individual and social learning be combined at both the practical task level and the abstract memory level? Do different combination strategies result in different levels of efficiency and / or goal accomplishment? Is this ‘meta-skill’ of hybrid learning itself innate, or somehow learned?
- What is the optimal trade-off between individual and social learning for a given task? How does this change with increasing task complexity? How is this affected by the nature of the task relative to, say, the embodiment of the executing agent?
- How can knowledge be consolidated to improve learning (of both kinds) next time? How are conflicts between what is learned socially and what is learned independently resolved? How easily applicable are social skills and their associated knowledge to individual learning situations, and vice versa?

I hope that these proposed research areas, and this paper as a whole, will in some way stimulate others into thinking about social learning in the context of a complete agent system.

References

Joanna J. Bryson and Mark A. Wood. Learning discretely: Behaviour and organisation in social learning. In *Proceedings of the Third International Symposium on Imitation in Animals and Artifacts*, pages 30–37, 2004.

Richard W. Byrne and Anne E. Russon. Learning by imitation: a hierarchical approach. *Behavioral and Brain Sciences*, 16(3), 1998.

Digital Extremes. *Unreal Tournament*, 1999. Epic Games, Inc.

Andrew N. Meltzoff and M. Keith Moore. Newborn infants imitate adult facial gestures. *Child Development*, 54:702–709, 1983.

Andrew N. Meltzoff and M. Keith Moore. Explaining facial imitation: A theoretical model. *Early Development and Parenting*, 6:179–192, 1997.

Chrystopher L. Nehaniv and Kerstin Dautenhahn. The correspondence problem. In Kerstin Dautenhahn and Chrystopher L. Nehaniv, editors, *Imitation in Animals and Artifacts*, Complex Adaptive Systems, chapter 2, pages 41–61. The MIT Press, 2002.

Deb K. Roy and Alex P. Pentland. Learning words from sights and sounds: a computational model. *Cognitive Science*, 26:113–146, 2002.

Aaron Sloman and Jackie Chappell. The altricial-precocial spectrum for robots. In *Proceedings IJCAI’05*, pages 1187–1192, Edinburgh, 2005.

Mark A. Wood and Joanna J. Bryson. Skill acquisition through program-level imitation in a real-time domain. Technical Report CSBU-2005-16, University of Bath Department of Computer Science, 2005. Submitted to IEEE Systems, Man and Cybernetics – Part B.

Biologically Inspired Robotics (Biro-net)

3rd - 4th April 2006

Organisers

Jeremy Burn, University of Bristol
Myra Wilson, University of Aberystwyth.

Programme Committee

Julian Vincent, University of Bath
Alex Ellery, University of Surrey
Mark Witkowski, Imperial College Lond.
Chris Melhuish, UWE
Ulrich Nehmzow, University of Essex
David Bisset, ITechnic
Olga Bogatyreva, University of Bath
Josef Schmitz, University of Bielefeld
Tony Prescott, University of Sheffield
John Hallam, Univ. of Southern Denmark
Ana Sendova-Franks, UWE
Barbara Webb, University of Edinburgh

Contents

Training a Genetic Algorithm and Neural Network Hybrid Pattern Classifier by Population Statistics

Paul Morley
Department of Design & Innovation
The Open University
Milton Keynes, MK7 6AA, England
`pm4958@tutor.open.ac.uk`

Abstract

A method of unsupervised learning is proposed that uses a genetic algorithm to train a neural network to implement a classification function where the only a priori information known about the population to be classified is the relative proportions of the different types.

The genetic algorithm is used to evolve the weights in a network (the structure is pre-determined) in order to provide some classification function that provides the same proportional split as that known, and it is postulated that generally this will provide the classification function required. It is furthermore postulated that with a minimal amount of human guidance the performance of the evolution can be directed, in an analogous manner to a human dog breeder who utilises the mechanics of evolution to a conscious end

Experiments were carried out firstly to establish some operational parameters - number of generations and mutation rate. These were not presented as being definitive optima - simply parameters that pragmatically gave reasonable results in a reasonable time.

Further experiments went on to investigate the effectiveness on this technique, when presented with 50 similar images that had been degraded with either noise, translation, or other such processing. It was found that the technique did give encouraging results, suggesting that it was possible to use this form of unsupervised training to adapt the network to the data. It was also found that some minimal human interaction - simply to make a 'live-die' choice for the members of an evolutionary set - could greatly improve the efficiency of the manner that this system homes in on solutions.

The paper concludes with some questions are suggested avenues for further work.

Autonomous Navigation Based on Optic Flow for ExoMars Rovers

Alex Ellery and Yang Gao
Surrey Space Centre
University of Surrey, Guildford, England, GU2 7XH
A.Ellery@surrey.ac.uk and Yang.Gao@surrey.ac.uk

Abstract

The ExoMars is an ESA flagship mission to search for life signature on Mars. Critical to this objective, the ExoMars rover must provide high mobility across potentially rugged terrain. Furthermore, locomotion must be achieved autonomously with minimal reliance on the Earth ground station. navigation involves four main processes: (a) perception of the environment; (b) self-localisation with respect to landmarks; (c) path planning; (d) path traversal. It is currently perceived that the CNES/LAAS autonomous navigation software will provide this function.

In broad terms, the CNES solution can be considered as a 'traditional approach' to navigation which employs stereovision and A* type path planning to provide a rover with independent navigation capability. Such methods are known to be computationally intensive. This imposes severe limitations given the relatively scarce onboard computational resources on the rover. Research in the area of autonomous navigation has advanced significantly in recent years particularly with the development of optical flow and ego-motion techniques. It seems prudent therefore to revisit the CNES solution to ascertain if performance can be improved in the light of recent advances.

In this study, we propose to adopt the optical flow technique in order to improve localisation and path execution efficiency and robustness of the CNES navigation system. Optic flow based on the navigation system of insects offers a robust obstacle avoidance strategy without the need for extensive vision processing. It uses the 2D motion of points in an image that only normal flow can be computed directly from image measurements. The 2D vector motion field is the perspective projection on the image plane of the 3D velocity field of a moving scene. Optic flow is the estimate of the motion field derivable from the variation in the image brightness pattern over time. The assumption is that the motion field and optic flow coincide as the spatial variations in intensity correspond to physical features on the visible 3D surfaces. The optic flow constraint assumes constant brightness in the direction of image gradient (normal component of optic flow). Therefore, it provides rapid task-relevant image processing without the need for complex 3D object recognition. The optical flow component is plugged into the CNES software and bypassing the computationally expensive steps. This enhancement allows obstacle avoidance on-the-fly without the need to stop regularly.

Bio-inspired Drill for Planetary Subsurface Sampling: Literature Survey, Conceptual Design and Feasibility Study

Yang Gao, Alex Ellery[*]
[*]Surrey Space Centre
University of Surrey, England
Yang.Gao@surrey.ac.uk
A.Ellery@surrey.ac.uk

Mustafa Jaddou, Julian Vincent[†]
[†]Centre for Biomimetic & Natural Technologies
University of Bath, England
en1mj@bath.ac.uk
ensjfvv@bath.ac.uk

Abstract

It is widely acknowledged that the next significant challenge in planetary exploration is to be able to drill deeply (two meters seems the most scientifically valuable and the most technologically reasonable) into the surface of solar system bodies for chemical or physical data. Major limitation of using conventional rotary drills in low gravity environments (such as Mars, asteroids, comet, etc) is the need for high axial force, which suffers from big overhead mass, buckling problem, and power hungriness. Though drills using percussive motion may operate in low mass and power, the drilling rate is generally slow. Drawing inspiration from nature for a lightweight and energy efficient solution, we propose a novel drilling method based on the working mechanism of wood wasp ovipositors. The bio-inspired drill requires no reactive external force by applying two-valve-reciprocating motion. The proposed bio-inspired system indicates enhanced utility that is critical for space missions where premium is placed on mass, volume and power. Biological systems are similarly constrained making biomimetic technology uniquely suited and advantageous as a model of miniaturized systems. As a result of the European Space Agency (ESA) project on bionics and space system design [Ellery, 2005], this paper presents a conceptual design of the bio-inspired drill. Lab-based experiments have shown that the two-valve-reciprocating drilling method is feasible and has potential of improving drill efficiency without any additional overhead force or mass.

1 Introduction

It is widely acknowledged that the next significant challenge in planetary exploration is the ability to drill deep into the surface of solar system bodies. Examples include astrobiological research to search for biomarkers about 2~3 m beneath the surface layer of most solar bodies (such as Mars [Ellery *et al.*, 2003]). Studies at lunar south pole region aim to search for water ice that is likely to exist under 1~2 m depth. The deep drilling system is therefore a crucial on-board instrument that can enable surface penetration, autonomous sample acquisition, and preparation for either *in situ* experiments or sample return procedures. ESA's ExoMars and ESA/NASA's Mars Sample Return missions scheduled for 2011 and 2016 will both require deep drilling capability and a great amount of work has to be performed in this area.

Based on the type of force applied, conventional planetary drills can be classified into two categories, namely rotary and percussive.

- Rotary drilling is the most common terrestrial approach to subsurface penetration. It is an extremely versatile method capable of penetrating cohesive and non-cohesive soils and rock. Examples include the coring tool (i.e. SD2) on ESA Rosetta Lander, and the small drill on NASA DS2 microprobe, etc. The major limitation of using rotary drills in low gravity environment is the need for high axial force, which results in high overhead mass (e.g. use extensive land support structures). Rotary drilling also suffers from bit dulling/breaking/jamming, power hungriness, and long drill string for deep drilling.
- Percussive drills are most viable in terms of mass and power consumption, but it has low penetration rate and difficulty in debris transport. Percussive drilling does not require drill fluid and utilizes short length drill strings. Examples include the ESA Beagle2 Mole [Kochan *et al*, 1999], the JPL Ultrasonic/Sonic Drilling/Coring device (USDC) [Bar-Cohen *et al.*, 2003], etc.

This paper aims to present a novel drill concept that provides a small, light and energy efficient solution to planetary exploration. Inspiration is drawn from working mechanism of the wood wasp ovipositor drill. The rest of the paper is organized as follows: Section 2 introduces the bio-inspired drilling mechanism and its significance; a conceptual design of the wood wasp drill is described in Section 3; Section 4 provides a feasibility study of the proposed drilling mechanism based on the lab experiments; last section concludes the paper and outlines the future directions.

2 Bio-inspired Drilling Mechanism

2.1 Wood Wasp Ovipositor Drill

The wood wasp uses its ovipositor to drill holes into trees in order to lay its eggs. Researchers [Vincent & King, 1995] analyzed the working mechanism of wood wasp ovipositor. As shown in Figure 1, the wood wasp ovipositor is composed of two significant halves: one side is the cutting teeth and the other side is the pockets to remove the debris. The wasp first stabs the surface of the wood in order to stabilize the ovipositor. The initial cut is done by the small proximally facing teeth at the base of the ovipositor, which breaks the cell wall in tension. Once this is achieved the push teeth can be used to cut the wood in compression without the fear of buckling. The push teeth are arranged in a staggered pattern in order to even out the forces required in cutting. The debris from the cutting teeth is deposited into the pockets that then carry it to the surface on the upstroke. Two sides repeat this process in a reciprocating motion. The wasp ovipositor drills at a rate of around 1 mm/min in the initial stage and 1.5 mm/min in the later stages.

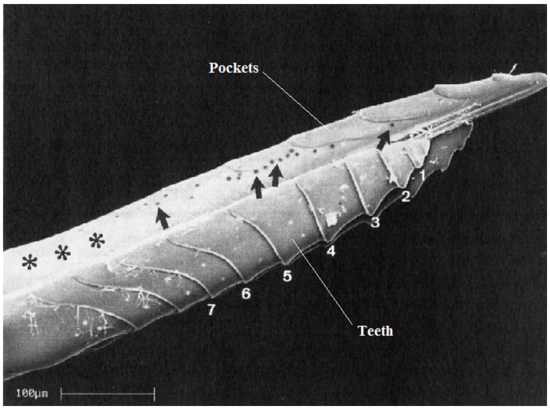


Figure 1: Wood wasp ovipositor [4]

2.2 Two-Valve-Reciprocating Drilling

The ovipositor drill uses two-valve-reciprocating rather than rotatory or percussive motion. The drill is composed of two valves that can slide against each other longitudinally as depicted in Figure 2. The reciprocating drill has backward-pointing teeth that present little resistance to being moved downwards but engage with the surrounding substrate to resist being moved in the opposite direction. Once the teeth are engaged, the tensile force that can be resisted, tending to pull the drill out of the substrate, allows the generation of an equal and opposite force in the other valve tending to push it further into the substrate. The drilling force is generated between the two valves and there is no net external force required. The limit to the drilling ability is the balance between the force required to pass through the substrate, the degree of engagement that the teeth can obtain on the substrate, and the bending strength of the teeth when they are engaged with the substrate. Another intriguing aspect of the two-valve-reciprocating mechanism is the effect to the drilling debris. Since the adjacent valves are moving in opposite directions, the debris is moved up the hole rather than deeper into it.

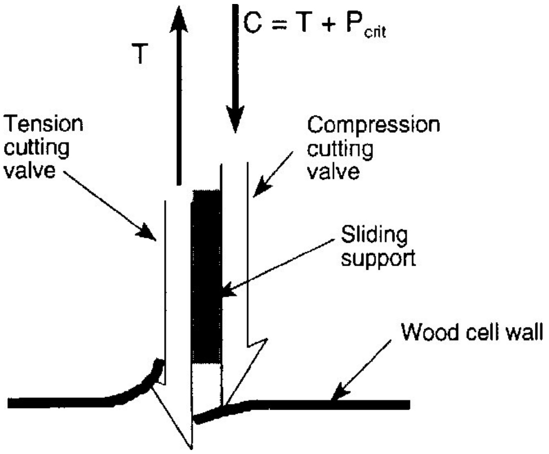


Figure 2: Model of ovipositor drill [Vincent & King, 1995]

A bio-inspired drill based on two-valve-reciprocating could provide a more compact and energy efficient solution. The novel drilling mechanism indicates some enhanced utility that may be incorporated into engineered systems inspired from biological systems. Such enhanced utility is critical for space missions where premium is placed on mass, volume and power. Biological systems are similarly constrained making biomimetic technology uniquely suited as a model of miniaturized systems.

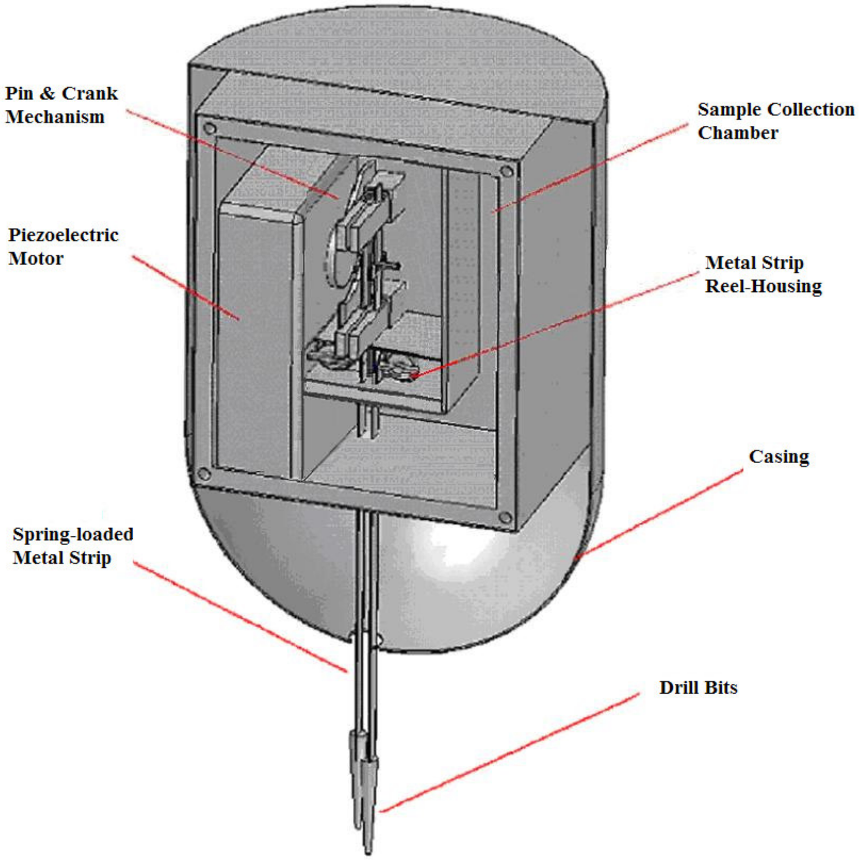


Figure 3: Bio-inspired drill and sampler system

3 Conceptual Design

3.1 Overall Design

The objective of this study is to design the wood wasp drill as a self-contained instrument that can be deployed from any machinery or platform. To represent the advancement in terms of mass, volume and power, the following design requirements are applied:

- Size: 5 (max diameter) x 7.5 (max length) cm
- Mass: 0.5 kg
- Power: 3 W
- Drilling depth: ~2 m

Figure 3 illustrates the preliminary design of the drill. Extra volume budget has been added to ensure sufficient space for all the elements.

3.2 Drill Bit Design

The drill bit is designed in the way to mimic the cutting teeth of the ovipositor drill. As shown in Figure 4 the drill bit is constructed in half cones (increasing in diameter) and the edges of the cones are used for the gripping and cutting action. The sharp pins are attached on the edge to increase gripping ability.

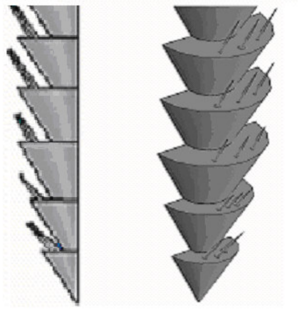


Figure 4: 2D and 3D view of designed drill bit

3.3 Drill Bit Deployment

Drill bits are attached to spring-loaded metal strips, which are reeled into a housing. The design of drill bit deployment is similar to a tape measure design, whereby the metal strip is wound into a reel. Upon reciprocation of the slider bars, the metal strip slides out of the housing (shown in Figure 5). The curved metal strip is free to slide against the slider bar. As the drill digs into the substrate, sample particles will move up to the sample collection chamber (explained in next section) and a hole is created. Once the drill bits are fully deployed a solenoid is activated to push the clip onto the metal strip that press onto the slider bar. This allows the slider bar to pull the metal strip back out of the hole.

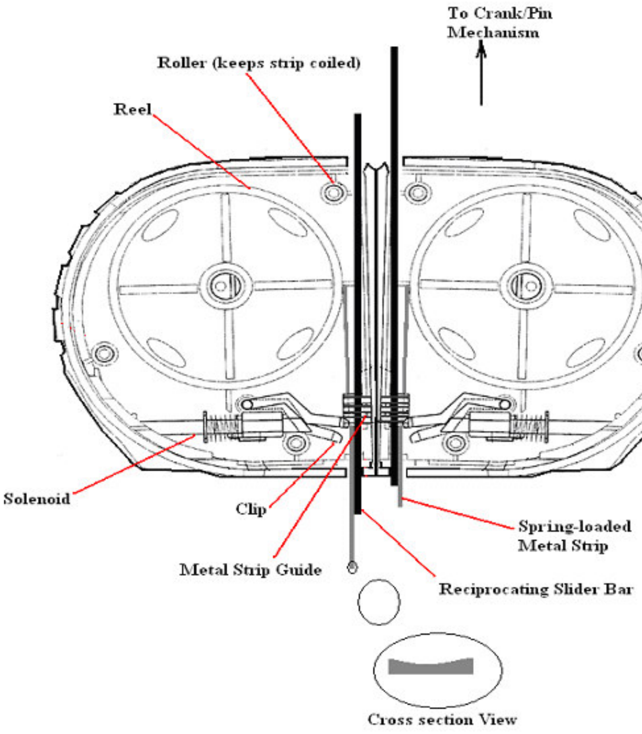


Figure 5: Metal strip reel housing [Jaddou, 2005]


2 Plates with angled fibres

Placed together

Particles at the bottom

Moving one side up & down moves the particles up, ejecting it to the side

Figure 6: Sample extraction mechanism [Jaddou, 2005]

3.4 Sample Extraction Mechanism

Sample extraction method is design based on the debris removal mechanism shown in Figure 6. This mechanism works in two-valve-reciprocating motion same as the drilling mechanism. When one side of the metal strip moves up, the other side moves down the same amount. Angled fibers are placed between the strips connected to the drill bit. Once particles are trapped inside the angled fibers between the metal strips, the fiber at one side lifts the particle and transports it to the opposite side. Consequently the particles can be collected between the metal strips and transport to the top and a hole is created.

3.5 Drive Mechanism

As shown in Figure 3, cam mechanism is used to drive the drill. The pin-crank mechanism is versatile and able to obtain almost any arbitrary specified motion. It also offers the simplest and most compact way to transform motions.

3.6 Actuation Method

In order to meet low cost design in terms of size, weight and power, the choice of actuation source is the piezoelectric motor. The piezoelectric motor shows superior torque and response time. It can generate forces of several 10,000 N over a range of more than 100 µm with sub-nanometer resolution. Other advantages of piezoelectric actuators include: 1) electrical energy is converted directly into motion. Since absorbing electrical energy during

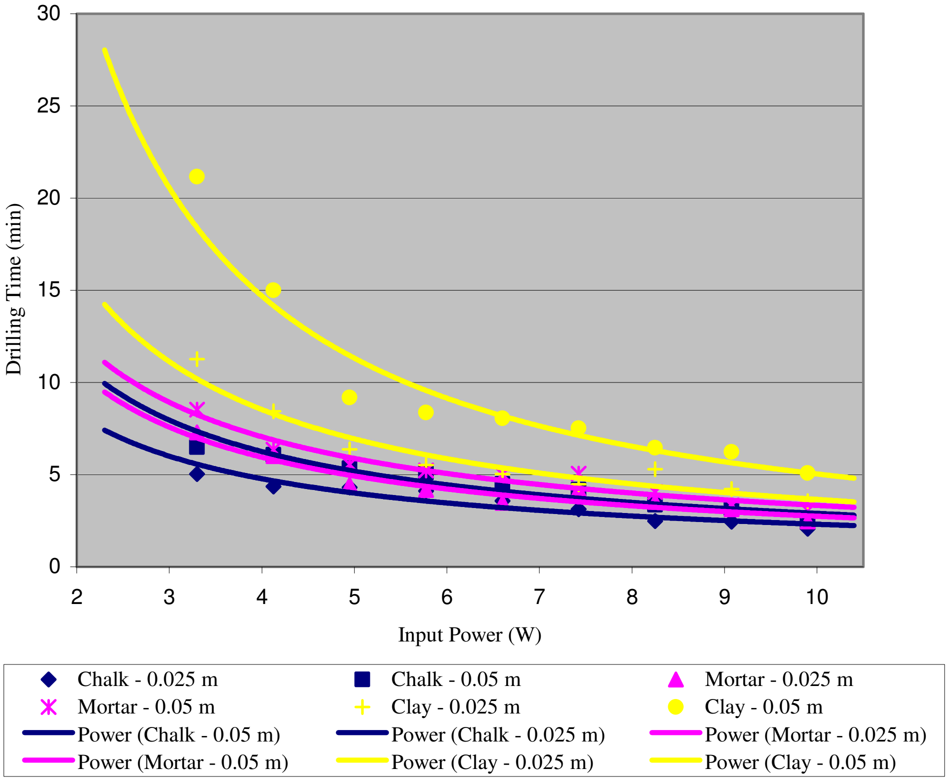


Figure 7: Power versus drilling time of 0.025 m/0.05 m depth

movement only, static operation, even holding heavy loads, consumes no energy; 2) the piezoelectric effect is related to electric fields and do not produce magnetic fields. Piezo-devices are especially well suited for applications where magnetic fields cannot be tolerated as for space applications; 3) the piezo-actuator has neither gears nor rotating shafts. Its displacement is based on solid-state phenomena and exhibits no wear and tear.

4 Feasibility Study

To verify the feasibility of the proposed drilling mechanism, a simplified drill prototype was built based on the design in Figure 3, containing mainly the drill bit, drive mechanism and actuator. The drill bit is about 1.8 cm in diameter and made of metal. It was tested on 3 different substrates: condensed chalk, lime mortar and none-fired clay. For each test, a range of input power from 0 to 10 W was applied to the drill (9 sampling points were taken). Time was recorded using a stopwatch for drilling two holes of 0.025-m and 0.05-m deep, where pre-drilled holes were formed to allow initial gripping.
Figure 7 records the time for drilling into three substrates at two different depths. Harder material like clay (blue lines) takes longer time to drill into than softer materials such as mortar (yellow lines) and chalk (pink lines). The test results tend to show that the drilling speed increases as the drill digs deeper. This could be due to the fact that as drilling deeper the substrate starts to form cracks and hence becomes easier to be chipped off. However, we take the worse case by linearizing the test data to approximate drilling speed at different input powers. Figure 8 plots the approximated relationship between drilling speed and input power for different test substrate. The drilling speed is estimated as a power function of the input power. It is anticipated that a linear relationship between the drilling speed and input power would be too simple to cope with the complexity of the drilling mechanism. Given an input power budget of 3 W, drilling speed was measured as 0.0056 m/min (chalk), 0.0046 m/min (mortar), and 0.0023 m/min (clay). Drilling speed with respect to the substrate compressive strength is predicted for 3W input power as shown in Figure 9.

Ratio of input power over material removal rate, i.e. P/Q, provides a measure of energy efficiency taking into account of power consumption and drilling speed. Smaller value of the ratio implies more energy efficiency of the drill. Table 1 compares the bio-inspired drill with two percussive drills. The proposed drill provides comparable performance especially for handling harder substrates. For conventional rotary drills with similar performance would require high axial force of ~10^2 N.

5 Conclusion

This paper proposed a bio-inspired drill concept for planetary sampling that can be used as an instrument for a generic space mission. The biomimetic drill represents a novel approach of two-valve-reciprocating drilling based on working mechanism of wood wasp ovipositors. It also has technology transfer applications within the terrestrial environment, such as geological drilling, ice coring, in-hole petroleum exploration, etc. This paper presented both conceptual design and feasibility study of the proposed drill. Lab-based experiment showed the potential of improving the drill efficiency within low mass, volume and power budget.

As for the future studies, we need to develop the optimal geometry and material of the drill bit, experiment on drill deployment mechanism and the sample extraction method, derive the empirical drilling model, and eventually build a system prototype.

Acknowledgment

This work was supported by European Space Agency (ESA) under Bionics & Space System Design contract AO/1-4469/03/NL/Sfe [Ellery, 2005].

References

Ellery, A., *et al.*, “Astrobiological instrumentation for Mars – the only way is down”, *Int. J. Astrobiology*, 1(4): 365-380, 2003

Kochan, H., *et al.*, “The mobile penetrometer (Mole): a tool for planetary sub-surface investigations”, *Proc. of the International Workshop on Penetrometry in the Solar System*, Graz, Austria, 213-243, 1999

Bar-Cohen, Y., *et al.*, Ultrasonic/sonic sampler and sensor platform for in-situ planetary exploration, *Proc. IEEE Int. Conf. MEMS, NANO and Smart Systems*, 2003

Vincent, J. F. V. and King, M. J., The mechanism of drilling by wood wasp ovipositors, *Biomimetics*, 3(4), 187-201, 1995

Jaddou, M., Autonomous biomimetic drill based on a *Sirex* wood wasp, project report, University of Bath, 2005

Ellery, A., *et al.*, Bionics & space systems design case study 2 – asteroid micro-penetrator with biomimetic drill, *ESA study contract report AO/1-4469/03/NL/SFe*, 2005

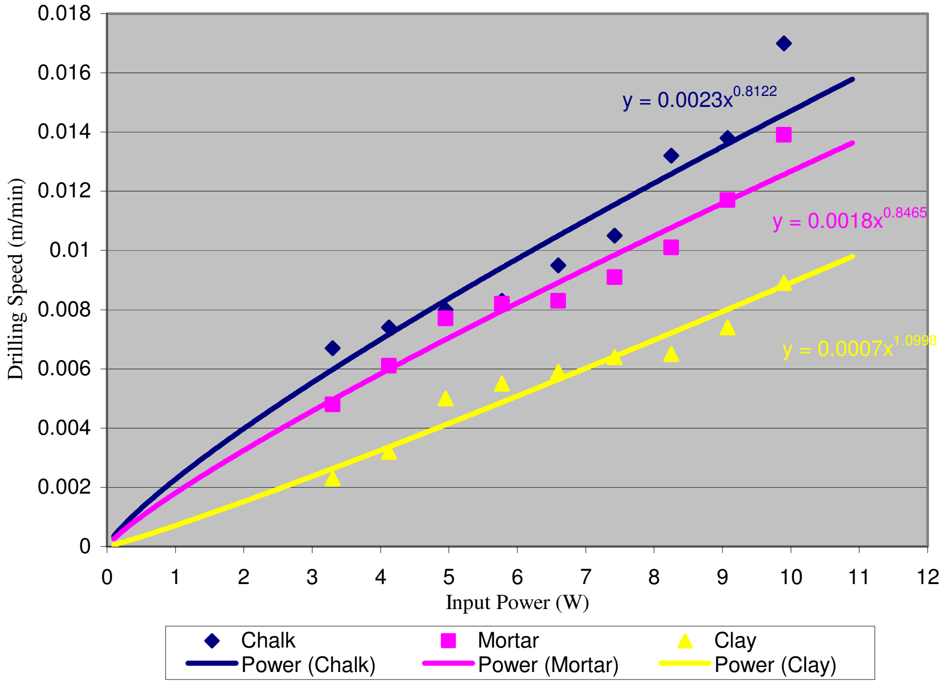


Figure 8: Approximated drilling speed versus input power

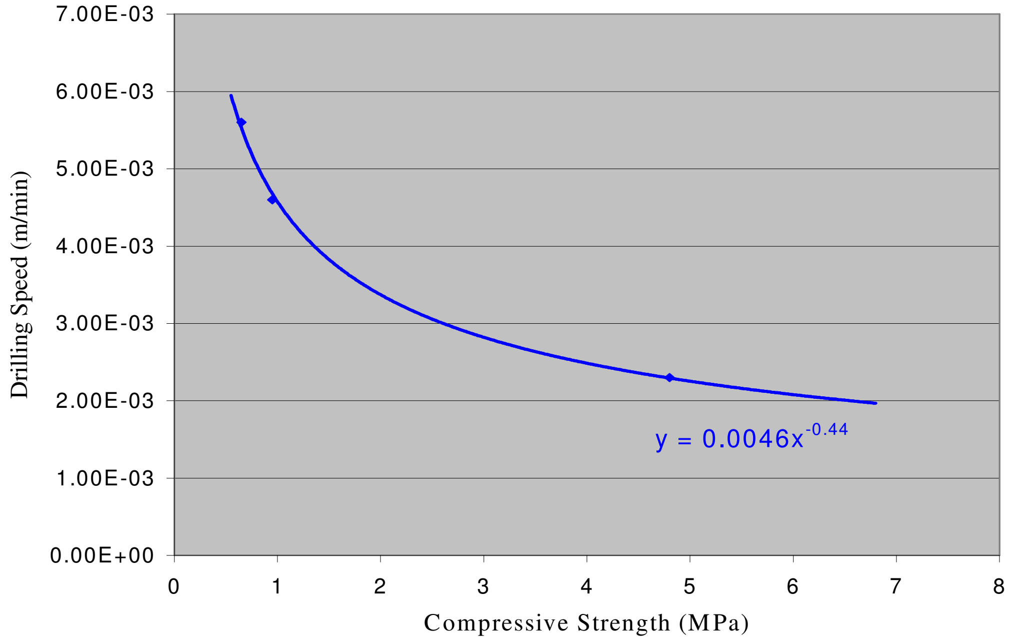


Figure 9: Predicted drilling speed versus compressive strength at 3W input power

Table 1: Comparison of three drills

	Bio-inspired drill	**Beagle 2/Mole**	**USDC**
Drill diameter (m)	0.018	0.02	0.003
Power (W)	3	5(peak)	5
Drilling speed (m/s)	~10^{-4} (soil) ~3×10^{-5} (rock)	~2×10^{-4} (soil)	~10^{-4} (rock)
Q (m^3/s)	$\pi \times 0.009^2 \times 10^{-4}$ (soil) $\pi \times 0.009^2 \times 3 \times 10^{-5}$ (rock)	$\pi \times 0.01^2 \times 2 \times 10^{-4}$ (soil)	$\pi \times 0.0015^2 \times 10^{-4}$ (rock)
Power/Q (J/m^3)	11.7×10^7 (soil) 3.9×10^8 (rock)	6.4×10^7 (soil)	7.07×10^9 (rock)

Neuro-Mechanical Entrainment in a Bipedal Robotic Walking Platform

Paschalis Veskos* and Yiannis Demiris*

*Biologically-inspired Autonomous Robots Team (BioART)
Department of Electrical and Electronic Engineering
Imperial College London
South Kensington, London SW7 2BT, UK
`{paschalis.veskos, y.demiris}@imperial.ac.uk`

Abstract

In this study, we investigated the use of van der Pol oscillators in a 4-dof embodied bipedal robotic platform for the purposes of planar walking. The oscillator controlled the hip and knee joints of the robot and was capable of generating waveforms with the correct frequency and phase so as to entrain with the mechanical system. Lowering its oscillation frequency resulted in an increase to the walking pace, indicating exploitation of the global natural dynamics. This is verified by its operation in absence of entrainment, where faster limb motion results in a slower overall walking pace.

1 Introduction

The traditional engineering approach to robot walking has been one based on control theory both as a methodology and actual solution of the problem. This is an on-line approach, as systems assess their performance and make corrections while walking. At the core of this approach is an internal model of system dynamics, such as the inverted pendulum (Kajita, Kanehiro et al. 2002), that physically interacts with the environment.

The most common stability criterion for dynamical walking makes use of the Zero Moment Point (ZMP). The ZMP is defined as the point in space where the robot's total moment about the ground is zero (Benbrahim 1996). The criterion states that the vertical projection of the ZMP on the ground should be contained within the robot's region of support. As such, the robot cannot topple about any point through which it contacts the ground. A review of several stability criteria can be found in (Garcia, Estremera et al. 2002).

While control theoretical approaches have been successful in creating legged locomoting robots, there are certain disadvantages in this methodology. First, there is a need for the mathematical model, which is specific to the particular system. This makes them inflexible in that they are not portable to different robotic designs and implementations. Additionally, model complexity increases disproportionately with the number of mechanical degrees of freedom. To illustrate the complexity of this task, it is worth noting that even though active embodied bipedal humanoids have existed for more than 35 years (Waseda 2003), a complete dynamic model of five-link (comprising of trunk, thighs and shanks) bipedal walking was only very recently published in 2003 (Mu and Wu 2003).

Much interest has been expressed in exploiting and making use of system dynamics, as opposed to competing against or avoiding them. Natural dynamics can be exploited at several levels; robot and task statics or dynamics can be utilised during the design process, task execution, or both. A review of all these cases can be found in (Williamson 1999).

In many implementations, computation has been reduced by "off-loading" calculations to the mechanical system. Inspired by the corresponding biological systems, designers have made use of natural dynamics to create walking robots with low-complexity control systems. This is termed physical (Lewis, Etienne-Cummings et al. 2003), or morphological (Paul 2004) computation, since leg trajectory calculation is explicitly shared between the robot's controller and mechanics.

The completely passive robots of McGeer (McGeer 1990) can be viewed as an extreme manifestation of this creed: since they do not make use of a controller at all, they completely rely on stable system dynamics to walk. It should be clear that this design methodology displaces the burden of creating proper walking controllers to constructing systems that are inherently able to walk without a controller. In other words, to reduce the control complexity of robots, more time needs be spent designing their mechanical traits. Thus, it has been recognised that control and bodily dynamics are closely linked and in the design process, both have to be considered (Ishiguro, Ishimaru et al. 2003).

1.1 Compliance

A typical feature of human actuator behaviour is mechanical compliance, allowing for misalignment between manipulators and contact surfaces and accommodating the smooth transition of forces from the no-contact mode region to contact with the environment (Dwarakanath, Crane et al. 2000). To achieve this in robotics systems there are two routes. The first requires passive compliance to be built into the manipulator by means of mechanical devices such as springs, as in the compliant arm of (Williamson 1999). Making changes to actuator control loops to achieve active compliance, as described in (Mason 1981) is termed force control. Here the controlled parameter is the force exerted by the motor output shaft and not its position, as in most servomotors.

1.2 Central Pattern Generators

It has been shown that CPGs exist in both lower vertebrates (Grillner, Buchanan et al. 1988) and higher mammals (Rossignol, Lund et al. 1988), generating motion patterns for breathing, heartbeat, mastication and locomotion (Cohen, Rossignol et al. 1988; Kimura, Sakurama et al. 1998). For example, a neural oscillator in the spinal cord of the cat autonomously creates rhythmical locomotive patterns (Shik and Orlovsky 1976).

Artificial neural oscillators try to mimic the biological systems' behaviour and structure. Probably the most popular model used is the Matsuoka oscillator (Matsuoka 1985), consisting of two mutually inhibiting neurons. It can be used to generate a stable bipedal gait as a system limit cycle by global entrainment between the rhythmic outputs of the oscillator and the musculoskeletal system (Taga 1991; Taga 1995b; Taga 1995a). The oscillator and musculoskeletal system are then mutually entrained and oscillate with the same frequency and phase. Matsuoka oscillators have also been used in a physical quadruped to create dynamic walking and running behaviours on irregular terrain (Fukuoka, Kimura et al. 2003).

The van der Pol oscillator (Strogatz 2001) was here chosen for the smaller number of parameters requiring tuning, robustness (Matsuoka is a near-harmonic oscillator that does not feature an asymptotically stable limit cycle) and straightforward computational implementation.

2 Experiments

In this study, we investigated the use of van der Pol oscillators in a 4-dof bipedal robotic platform for the purposes of planar walking. The oscillator controlled the hip and knee joints of the robot to determine its capability of generating waveforms with the correct frequency and phase so as to entrain with the mechanical system.

Lowering its oscillation frequency resulted in an increase to the walking pace, indicating exploitation of the global natural dynamics. This is verified by its operation in absence of entrainment, where faster limb motion results in a slower overall walking pace.

Neither the robotic system or the environment were modelled, but proprioceptive information, i.e. the position of the robot's torso, was physically embedded in the system. The same strategy was followed by (Lungarella and Berthouze 2002) to make use of the distributed control possibilities to developmentally investigate a swinging motion by a small humanoid robot.

2.1 Apparatus

The mechanical setup used in our experiments is shown in Figure 1. The robot is supported by a metal arm and thus restricted to move in the sagittal plane, i.e. vertical to the arm. The arm can rotate freely about a vertical axis, by means of a free revolute joint. The natural elasticity of the arm allows for limited vertical travel. A webcam is mounted on the arm itself, next to the axis of rotation, so that it rotates in the same frame of reference as the robot, yet does not move vertically. The webcam is used to track the position of a coloured marker placed on the robot's body. The marker's ordinate is then used as the feedback signal for the neural oscillators.

The robot features two servomotors on each leg, actuating the hip and knee joints. These motors are position-controlled and hence maintain joint stiffness when idle. To exploit this characteristic, the system was so arranged that the metal arm pushes the standing humanoid downwards. This has the effect of forcing the motors to oppose the disturbance.

While this may appear to cause unnecessary energy consumption, it has the effect of loading the arm with a small force. As a result, the vertical motion of the robot is affected by the natural dynamics of the metal arm, something that the neural oscillators can utilise. Finally, small (~1cm thick) rubber feet were attached to the bottom of the shanks in order to partially absorb impact shocks, filter friction and provide grip against the floor.

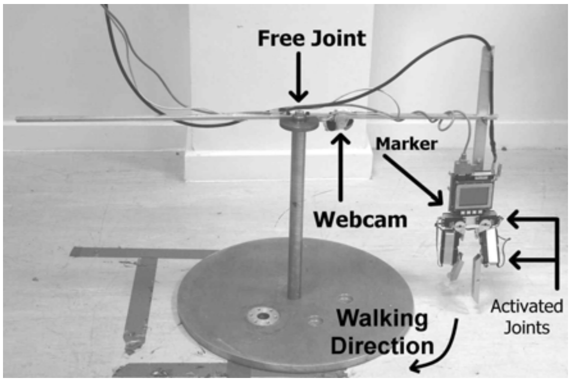


Figure 1: The experimental setup used in this study. The elasticity of the suspending arm is exploited to absorb impact shocks and provide the system with a rudimentary form of compliance. The vertical motion of the robot body as it walks is used as the feedback signal for the neural controller.

A PC host is used to collect sensory data via the webcam and perform the necessary vision processing. This information is then inserted in the nonlinear equations for the neural oscillators. These are solved with a numerical integration library and the appropriate motor commands are generated in realtime. Finally, they are transmitted over a serial link to the robot's onboard controller and executed by the motors.

2.2 Non-linear Oscillator

The equation for the van der Pol oscillator used in our experiments is:

$$\ddot{x}_i + \mu\left(x_i^2 - 1\right)\cdot\dot{x}_i + \omega^2 x_i = G_{in}\cdot fb + G_{i-j}\cdot x_j,$$
$$i, j = \left\{hip, knee\right\} \qquad (1)$$

Where $\mu \geq 0$ is a parameter controlling the damping term, ω is the natural frequency of the oscillator, fb represents the feedback from the vision system, G_{in} is the feedback gain, while G_{i-j} is the cross-coupling term gain, linking joints *i* and *j*. Overall, two oscillators are used, one for the hip and one for the knee joint. The output of each is used for both the left and right joints. Thus, the two sides move in phase with each other, resulting in a 'hopping' motion.

The value of G_{in} has to be sufficiently high to excite the oscillator in entraining to the feedback signal, yet not make it excessively sensitive to spurious inputs such as noise and high-frequency harmonics. To aid this effort, an exponentially decaying moving average was applied to the feedback signal. This way the value of *fb* used at each time-step n was averaged with 30% of the value of the previous sample:

$$fb_n := 0.5\cdot\left(0.7\cdot fb_n + 0.3\cdot fb_{n-1}\right) \qquad (2)$$

The gains affecting the cross-coupling terms control the influence each oscillator has on its counterpart. If they are not equal to each other, a phase difference between the two oscillators is introduced.

The natural frequency was set to $\omega^2 = 10.0$, a high value, to enable quick relaxation oscillations when the system failed to entrain. This should aid the system in avoid getting stuck in stationary local minima.

Unless mentioned otherwise, the following values were used throughout this study: $\mu = 1.00$, $G_{in} = 20.0$ and $G_{hip-knee} = G_{knee-hip} = 0.50$. The initial conditions for the numerical integrators throughout this study were: $\left\{x_{hip}, \dot{x}_{hip}, x_{knee}, \dot{x}_{knee}\right\} = \left\{0.00, 0.00, 0.00, 0.00\right\}$

2.3 Gait Design

The solution of the oscillator equation is not directly fed to the motors, but use of its derivative is made instead. The command activating the position-controlled joints, is:

$$\theta_i = G_{out}\cdot sign\left(\dot{x}_i\right) + C_i \qquad (3)$$

where G_{out} is the output gain and C a constant.

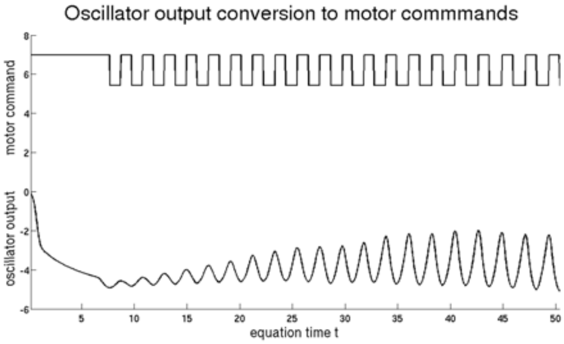


Figure 2: Example oscillator output (e.g. x_{hip}, bottom waveform) and corresponding motor commands (θ, top waveform). Note that the drift in the DC component of the oscillator output does not affect the motor command.

This post-processing is necessary as the van der

Pol oscillator's output varies in amplitude, dependant on the amplitude of its inputs (Williamson 1999). It was also experimentally found that even brief loss of entrainment could produce considerable DC drifts. We thus converted the periodic oscillator signal to a pulse-width modulated square wave retaining frequency and phase information (Figure 2). This worked well with the low-bandwidth model aircraft motors we used.

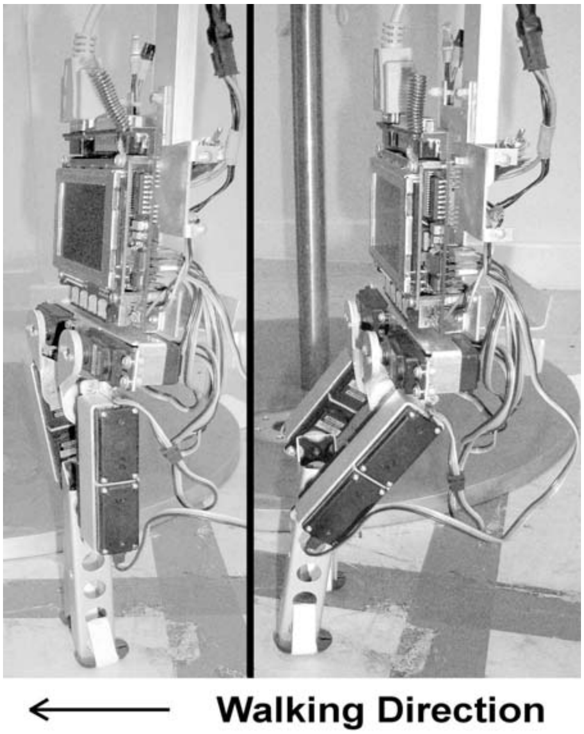


Figure 3: The two robot stances we used: joints fully extended (left) and contracted (right).

The endpoints of the robot's limbs motion was set by choosing suitable values for G_{out} and C_i so as to generate a basic gait. There were a number of desired properties for this gait: apart from propelling the humanoid forwards, an adequate vertical motion of the torso had to be detected by the feedback system. This was achieved by extending the knee at touchdown and hence pushing vertically against the floor to raise the robot's body. At the same time, the limb travel distance had to be limited to control impact intensity due to the restricted compliance of our setup. The resultant postures chosen can be seen in Figure 3.

3 Results

In the absence of a sufficiently large amplitude periodic feedback signal (as in the beginning of an experimental run, when the robot is stationary), the oscillator caused the motors to move at its natural frequency. This was too high for the motors that could not follow the gait trajectories in their entirety. Therefore, the steps the robot made were incomplete and stride length was small. These small and quick steps caused the robot to walk at a very slow pace. However, this caused the feedback signal amplitude to increase (see Gait Design, above), which in turn led to the oscillator entraining to the mechanical system's dynamics. This was indicated by a reduction in the frequency of the neural oscillator and significant increases in walking speed and stride length.

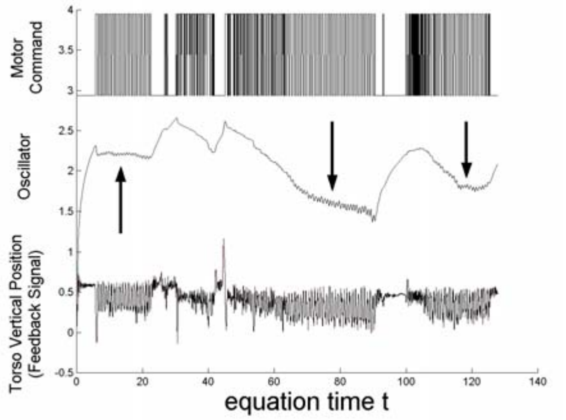


Figure 4: Time series for a typical experimental run showing, from top to bottom, the motor commands, nonlinear oscillator output and feedback signal. The arrows denote the regions where entrainment was achieved.

Data from such an experiment is shown in Figure 4, where the feedback signal, oscillator output and motor commands are plotted against (oscillator) equation time. The three arrows in the figure denote the regions where entrainment was achieved. There the oscillator produced stable, large and generally constant amplitude oscillations, while frequency- and phase-locking to the feedback signal. It becomes clear that in between these regions of stability, the oscillator's output amplitude and DC component show significant drift. However, this phenomenon is totally absent in the first case of entrainment and significantly reduced in the other two.

The oscillator and feedback signals during the above first occurrence of entrainment can be seen in more detail in Figure 5. The frequency- and phase-locking of the neural oscillator to the mechanical signal are evident for t=7 to t=23. The feedback signal resembled a pure sine wave for the first three periods of entrainment. However, harmonics began to manifest themselves after that, distorting the signal. This eventually caused the oscillator to lose its lock on the fundamental frequency and destroyed entrainment.

The phase plots for the above two signals in this experiment are shown in Figure 6. The initial high frequency oscillations occur without forming a closed limit cycle. This is possible because the motor commands are generated by taking the oscillator's derivative, rather than its direct output into account (equation (3)). However, entrainment creates a stable limit cycle, seen as a 'tunnel' in the 3D plot. This type of graph additionally emphasises oscillator DC drift, allowing for a better assessment of entrainment 'quality'. Thus, in its second occurrence, the 'entrainment tunnel' can be seen drifting, indicating an imperfect lock.

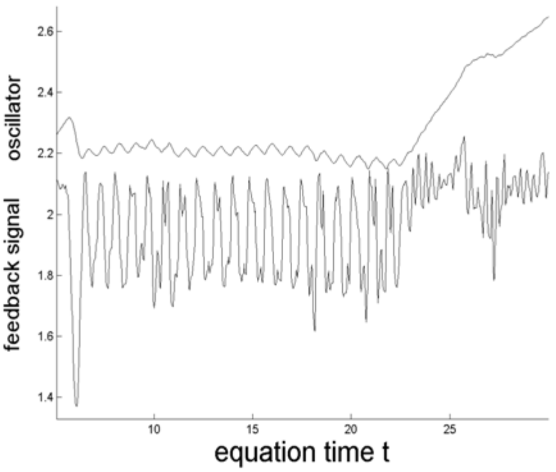


Figure 5: Close-up of the oscillator output and feedback signal in the presence of neuromechanical entrainment. The oscillator has adapted its output frequency and phase-locked to the feedback signal for a number of periods.

This is reflected on the mechanical signal; when the oscillator entrains well with the natural dynamics, the mechanical system increases the amplitude and velocity of its oscillations very rapidly. Since there are very few iterations with increasing amplitude, this is indicated by a clear 'entrance' to the tunnel, as in the first and third cases. In the second case, the tunnel is preceded by a growing spiral and does not maintain constant volume throughout its length. Loss of entrainment is indicated by sharply diminishing amplitude of the mechanical oscillation in all cases.

The system is capable of entraining to the natural dynamics of the combined robot-metal arm system. It cannot, however, maintain the entrained configuration for an extended period of time. The problem appears to be the distortion of the feedback signal by high-frequency harmonics, destroying entrainment. These harmonics are introduced after a small number of periods following the onset of entrainment. In behavioural terms, the humanoid typically makes 2-3 successful steps before motion becoming affected by these strong oscillations throughout the apparatus. They could be caused by the impact shock not being sufficiently absorbed by the compliant mechanism and thus transmitted to the robot's body.

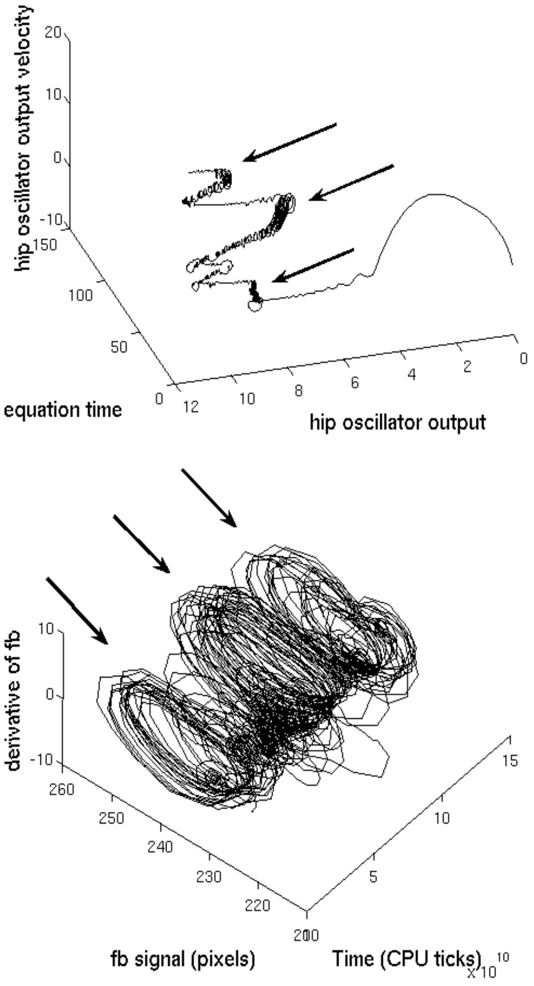


Figure 6: Oscillator output (top) and mechanical feedback signal (bottom) phase plots for the given experimental run. Again, the arrows denote the three regions during where entrainment was achieved; they are characterised by consistency in the oscillator output and large amplitude mechanical oscillations. The mechanical feedback signal has additionally been smoothed in this plot with a 5-

point moving average to reduce high frequency noise and discriminate it from stable, large amplitude oscillations.

4 Discussion

In this study, we have demonstrated the use of the dynamical systems paradigm to realise a walking behaviour in an embodied robotic walking platform. Body and neural oscillator dynamics have interacted with the environment to rapidly entrain to the natural dynamics of the combined robot-metal arm system. The oscillator lowered its output frequency, resulting in an increase to the walking pace, indicating exploitation of the global natural dynamics. This is verified by its operation in absence of entrainment, where faster limb motion resulted in a slower overall walking pace. Although we have made use of a basic planar bipedal platform, the successful application of the same neural architecture in a robot swinging task (Veskos and Demiris 2005) demonstrates the elegance of the coupled dynamical system approach.

Work is currently underway to further stabilise the system by augmenting the mechanical setup to enhance its compliant attributes. More advanced forms of filtering the feedback signal to the same end are also being investigated. Expanding the neural structure with separate van der Pol oscillators for each joint will allow the realisation of an antiphase gait. This has already been achieved in simulation, albeit in a simplified mechanical system without modelling dynamics (Zielinska 1996). The more complex neural system will, however, feature a large number of parameters that will require tuning. This process could be automated by means of a learning algorithm such as reinforcement learning (Sutton and Barto 1998).

Acknowledgments

The authors wish to thank the rest of the members of the BioART team, Anthony Dearden, Matthew Johnson, Bassam Khadouri and Gavin Simmons for their suggestions and constructive criticism.

References

Benbrahim, H. (1996), Biped Dynamic Walking Using Reinforcement Learning, PhD Thesis, Electrical Engineering, University of New Hampshire, New Hampshire.

Cohen, A. H., S. Rossignol, et al. (1988). Neural control of rhythmic movements in vertebrates. New York, Wiley.

Dwarakanath, T. A., C. D. Crane, et al. (2000). In-Parallel Passive Compliant Coupler for Robot Force Control. Biennial mechanisms and robotics conference, pp. 587-594, Baltimore, MD, Asme, in 7A.

Fukuoka, Y., H. Kimura, et al. (2003). "Adaptive Dynamic Walking of a Quadruped Robot on Irregular Terrain Based on Biological Concepts." International Journal of Robotics Research 22(3/4): 187-202.

Garcia, E., J. Estremera, et al. (2002). "A comparative study of stability margins for walking machines." Robotica 20(6): 595-606.

Grillner, S., J. T. Buchanan, et al. (1988). Neural Control of Locomotion in Lower Vertebrates: From Behaviour to Ionic Mechanisms. in Neural control of rhythmic movements in vertebrates. A. H. Cohen, S. Rossignol and S. Grillner (Eds). New York, Wiley.

Ishiguro, A., K. Ishimaru, et al. (2003). How Should Control and Body Dynamics be Coupled? - A Robotic Case Study. Intelligent robots and systems; 2003 IEEE/RSJ international conference on intelligent robots and systems (IROS 2003), pp. 1727-1732, Las Vegas, NV, IEEE, in

Kajita, S., F. Kanehiro, et al. (2002). A Realtime Pattern Generator for Biped Walking. International conference on robotics and automation, pp. 31-37, Washington, DC, IEEE, in

Kimura, H., K. Sakurama, et al. (1998). Dynamic Walking and Running of the Quadruped Using Neural Oscillator. Intelligent robots and systems, pp. 50-57, Victoria; Canada, IEEE, in

Lewis, M. A., R. Etienne-Cummings, et al. (2003). "An in silico central pattern generator: silicon oscillator, coupling, entrainment, and physical computation." Biological Cybernetics 88(2): 137-151.

Lungarella, M. and L. Berthouze (2002). "On the Interplay Between Morphological, Neural, and Environmental Dynamics: A Robotic Case Study." Adaptive Behavior 10(3/4): 223-242.

Mason, M. T. (1981). "Compliance and Force Control for Computer-Controlled Manipulators." IEEE Trans on Systems, Man, and Cybernetics 11(6): 418--432.

Matsuoka, K. (1985). "Sustained oscillations generated by mutually inhibiting neurons with adaptation." Biological Cybernetics 52: 367-376.

McGeer, T. (1990). "Passive Dynamic Walking." International Journal of Robotics Research 9: 62-82.

Mu, X. and Q. Wu (2003). A complete dynamic model of five-link bipedal walking. American control conference, pp. 4926-4931, Denver, CO, IEEE, in

Paul, C. (2004). Morphology and Computation. International Conference on Simulation of Adaptive Behavior; From animals to animats 8, pp. 33-38, Los Angeles, Calif, Cambridge Massachusetts, in

Rossignol, S., J. Lund, et al. (1988). The Role of Sensory Inputs in Regulating Patterns of Rhythmical Movements in Higher Vertebrates: A Comparison Between Locomotion, Respiration, and Mastication. in Neural control of rhythmic movements in vertebrates. A. H. Cohen, S. Rossignol and S. Grillner (Eds). New York, Wiley.

Shik, M. L. and G. N. Orlovsky (1976). "Neurophysiology of locomotor automatism." Physiol Rev 56(3): 465-501.

Strogatz, S. (2001). Nonlinear Dynamics and Chaos: With Applications to Physics, Biology, Chemistry and Engineering, Perseus Books Group.

Sutton, R. S. and A. G. Barto (1998). Reinforcement learning : an introduction. Cambridge, Mass. ; London, MIT Press.

Taga, G. (1991). "Self-organised control of bipedal locomotion by neural oscillators in unpredictable environment." Biological Cybernetics 65: 147-159.

Taga, G. (1995a). "A model of the neuro-musculo-skeletal system for human locomotion. I. Emergence of basic gait." Biological Cybernetics 73(2): 97.

Taga, G. (1995b). "A model of the neuro-musculo-skeletal system for human locomotion. II. Real-time adaptability under various constraints." Biological Cybernetics 73(2): 113.

Veskos, P. and Y. Demiris (2005). Developmental acquisition of entrainment skills in robot swinging using van der Pol oscillators. The 5th International Workshop on Epigenetic Robotics (EPIROB-2005), pp. 87-93, Nara, Japan, in Lund University Cognitive Studies, 123.

Waseda (2003), Waseda University Humanoid Robotics Institute [UoW HRI Homepage], http://www.humanoid.rise.waseda.ac.jp/booklet/booklet2000.html, Accessed: 7 March, 2004.

Williamson, M. (1999), Robot Arm Control Exploiting Natural Dynamics, PhD Thesis, Department of Electrical Engineering and Computer Science, Massachusetts Institute of Technology, Cambridge, Massachusetts.

Zielinska, T. (1996). "Coupled oscillators utilised as gait rhythm generators of a two-legged walking machine." Biological Cybernetics 74(3): 263-273.

Spider-inspired embedded actuator for space applications

C.Menon[*]
[*] Advanced Concepts Team`
European Space Agency,
The Netherlands
Carlo.Menon@esa.int

C.Lira[#]
[#] Centre of Mechatronics
Science Studies and Information, Kaunas
University of Technology, Lithuania
info@VSFproject.com

Abstract

A novel mechanism inspired by spider legs is presented and discussed in this paper. The mechanism has the potential to be used in future space applications, although the harsh space conditions, and in particular outgassing, should be carefully addressed in the design of a space-qualified model, The mechanism, called “Smart Stick”, has one degree of freedom and is actuated by a pressurised fluidic system. The prototype, which has been designed, built and tested, is of compact size and presents a repeatable behaviour. The relation between pressure and rotation is approximately linear when the pressure is less than 1.2 MPa. The mechanism is suitable for a modular configuration in which several Smart Stick modules are joined together. This modular configuration allows large rotations and does not increase the complexity of the actuation.

1 Introduction

This work proposes a novel, integrated joint-actuator for space use. The mechanism takes inspiration from spider limb joints, and can be efficiently integrated in lightweight structures.

1.1 Inspiration from nature

Most animals have opposing muscles to articulate their joints: 1) flexors, which are used to bend the limbs, and 2) extensors, which straighten the joints. In spider legs, however, some joints do not have extensors. Muscles inside the prosoma (see Fig. 1) can increase the fluid pressure inside the spider, acting as a pump. The pressurised fluid inside the limbs can therefore extend some joints of the spider’s legs.

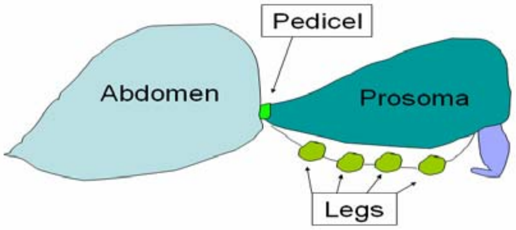


Fig. 1 Sketch of spider parts

1.2 Inflatable systems

Space applications require structures, mechanisms and systems that are able to fulfil challenging tasks, while keeping their volumes and masses to a minimum. Gossamer structures are already a reality in the space field and research is giving new results thanks to the use of innovative materials and technologies. The Echo Balloon, the Inflatable Torus Solar Array Technology (ITSAT), the Inflatable Antenna Experiment (IAE) and the inflated-spherical-wheel rover are just few examples of space inflatable structures which have been tested in the past years (Jenkins, 2001; Yoseph Bar-Cohen, 2000).

The use of inflatable mechanisms has some limitations due to the harsh space environment. Charged and high-energy particles, gas loss, solar ultraviolet and space radiation play an important role in the selection and development of inflatable materials.

In terrestrial use, pneumatic systems are often used for actuation. Pneumatic actuators are generally very fast but difficult to control. Often a ‘bang-bang’ control is used. Double-acting cylinders are advisable for differential pressure control. In this case, however, the complexity, cost and weight are increased. Muscle bio-inspired pneumatic actuators, firstly conceived for use in artificial limbs in the 1950-60's (Knight and Nehmzow,2002) are now used and commercialised for anthroform bio-robotic arm development. This simple device consists of a braided sleeve, usually made of nylon wires, which contains a deformable bladder. Compressed air is used to inflate the bladder, and, since the strand is less extensible, an axial contraction of the actuator occurs. They are characterised by a high strength to weight ratio but the need for storage of compressed air in autonomous systems limits their use.

1.3 Hydraulic mechanisms

Literature on space inflatable systems that use liquid fluids as the inflating means is not as common as that for gas inflatable systems. Hydraulic mechanisms are generally not chosen for on-orbit/planets applications (Benaroya, Bernold, M.Asce and Koon Meng Chua, F.Asce, 2002), for several reasons, e.g.:

1. Liquid fluids outgas
2. Liquids are temperature sensitive
3. Liquids are heavier than gases
4. Hydraulic pumps induce vibrations and cavitations can occur

The use of liquid fluids can, however, lead to new solutions and applications if the space hazards are carefully considered. The use of closed systems (no exchange with the environment) and low outgassing fluids characterized by low sensitivity to temperature changes can lead to the design of hydraulic space-qualified systems. Inflatable systems could also be designed by controlling the elasticity of the liquid tank. Miniaturized hydraulic mechanisms can be of particular interest. In fact, space applications do not generally require high force actuators, while mass must always be minimized. Taking advantage of their high strength-to-weight ratio, miniaturized hydraulic mechanisms could represent a compelling new approach.

1.4 The challenging space environment

A fundamental challenge for the use of inflatable space mechanisms is the space environment. Temperature range, pressure and atmospheric (if any) composition must be considered when a space mechanism is designed. Table 1 summarises salient characteristics of some planets of the Solar Systems (Barik, 2001).

Outgassing and fluid leakage are important issues which must be considered during the design phase of space mechanisms. Considering, for instance, a probe on the Mars surface, the pressure is in the range of 0.7-0.9 kPa, whereas on Venus there is a pressure of 9320 kPa. Mars has an average temperature of 63°C whereas Venus is at about 464°C, which could be prohibitive for conventional inflatable space architectures. The design of inflatable systems must therefore carefully take into account the surrounding operational environment.

In the framework of this study, which is at an early stage, the Earth environment was considered, because prototyping was considered a necessary step for the mechanism synthesis. Modifications must therefore be introduced when a different environment is considered.

Table 1 Planet characteristics

	T (°C)		Pressure	Composition
	Range	Average	(kPa)	
Mercury	[-173,427]	179	None	K (31.7%)
Venus	[-44,500]	464	9320	CO_2 (>96.4%)
Earth	[-69,58]	7	101	N_2 (>78%)
Mars	[-140,20]	-63	0.699-0.912	CO_2(>95.3%)
Jupiter	[-163,/]	-121	>10100	H_2(>81%)
Saturn	[-191,/]	-130	>10100	H_2 (>93%)
Uranus	[-214,/]	-205	>10100	H_2 (>82%)
Neptune	[-223,/]	-220	>10100	H_2 (>84%)
Pluto	[-240,-218]	-229	None	CH_3

2 Leg extension system in spiders

The legs of several arthropods (arachnids, diplopods, chilopods, pauropods) have joints that can be classified as hinge joints (Manton, 1958a, 1958b). The anatomical form of the joint often does not permit the presence of antagonistic extensors, which are often substituted by hydraulic systems. The empty spaces in between muscles and skeleton are usually filled with hemolymph. This pressurised liquid is used as a means to pressurise the spider's joints. Thin channels supply hemolymph to peripheral segments through the leg.

Parry and Brown, in 1959, carefully investigated spider legs and their pressurised mechanism. They showed experimentally that the active extension, which occurs at the hinge joints of the Tegenaria legs, is based on a hydraulic mechanism. They measured the internal pressure in the leg of an intact spider, established an empirical relation between the internal pressure and joint torque, and performed measurements of the actual torque developed when a spider accelerates a mass attached to its leg.

Many methods used to measure the leg inner pressure take advantage of the thin flexible articular membranes at the joints. Parry and Brown, for instance, used a sleeve sealed over the leg. The pressure in the sleeve was slowly raised until the membrane collapsed (Parry and Brown, 1959). Blickhan and Barth, in 1985, used a transducer with a tip smaller than the leg blood channels. The measurement system was mounted on freely moving spiders and also connected to several points of the hunting spider *Cupiennius salei* by means of tethers with negligible weight.

The mechanism for leg extension is now well documented in various spiders and whipscorpions (Sensenig and Shultz, 2003 and 2004). *Aphonopelma* have an inner pressure of about 5.3-8 kPa (Stewart and Martin, 1974) and in walking *Mastigoproctus* the fluid arrives at a pressure up to 9 kPa (Shultz,1991). Blickhan and Barth, in 1985, measured up to 70kPa on *Cupiennius salei* legs (130kPa during autonomy). Several hypotheses have been supported by experiments to explain how the prosoma natural pump works (Parry and Brown, 1959; Shultz, 1991; Stewart and Martin, 1974; Wilson and Bullock, 1973; Anderson and Prestwich, 1975).

Biologists were also able to measure the torque exerted by pressurised limb joints. It was shown that the torque increases approximately linearly with pressure (Sensenig and Shultz, 2003). The tibia–basitarsus joints of the tarantula *Aphonopelma* exert 20–74 mN mm when a pressure over the range 2.5–9.8 kPa is provided. Recent research reported high-resolution images, using transmission electron microscopy (TEM), which show cross-sections of spider legs. By studying the legs of *Leiobunum Nigripes*, Guffey (Guffey, Townsend and Felgenhauer, 2000), e.g., observed the presence of a single tendon, within a hemocoelic space, connecting the tarsal claw to the claw-flexing musculature of the tibia. The spider mechanism is also able to provide explosive force. High pressure can extend the rear legs, allowing spiders to jump (Parry and Brown, 1959).

There have not been many attempts to design engineering models of spider joints. One of the most remarkable works was done by Schworer, Kohl and Menz (1998). Their mechanism was actuated by nitrogen and was able to lift a weight of 8.2mN. The mechanism was built using extremely expensive equipment and processes (e.g. LIGA) suitable only for micro-systems.

3 Problem definition

Fig. 2 shows a flow diagram of the mechanism to be designed. A command is given to an actuator, which compresses the working fluid. The induced reversible deformation of an inflatable structure produces a rotation of a flexible joint. The joint rotation is fed back by a sensor in order to correct the command.

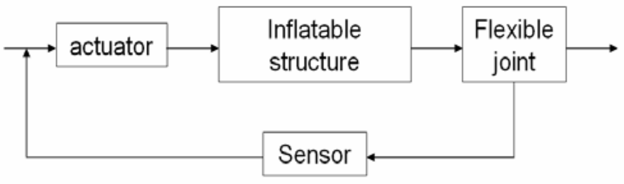


Fig. 2 Flow diagram

The system should have a closed fluid system in order to reduce leaking problems and outgassing phenomena, which are compelling challenges for space mechanisms. There are no limitations on the working fluid, which could be a gas or a liquid.

The synthesis of the novel bio-inspired mechanism described in this paper focuses on the inflatable structure embedded in a flexible joint. The sensor unit and the mechanism used to compress the working fluid are not analyzed at this stage. The mechanism is intended for operation in the space environment. However, the breadboard described in this paper was designed for terrestrial experiments.

Besides space and robotic applications, the proposed fluidic actuator could also be used in wearable equipment, e.g. smart bra (see patent document Lira, Angrilli and Debei, 2003).

4 Smart Stick

The novel conceived mechanism, called “Smart Stick”, is based on the use of a miniaturised tube (outside diameter 1 mm). The tube can be embedded into flexible structures to obtain integrated systems. Fig. 3 shows the shape and dimensions of the miniaturized pipe obtained by plastic deformation. The part of the micro-tube having elliptical shape constitutes the “fluidic actuator” of the system. Fig. 4 shows the tube embedded in the “Smart Stick” structure. The actuator acts upon hydraulic principles and the effects of the pressurisation are stiffness variation and bending force generation in the structure.

The actuator manufacturing process can be repeated along the whole tube. The repeated process allows easy fabrication of a series of miniaturised fluidic actuators. By folding and deforming the micro-tube, mechanical connectors are avoided making the system simple, reliable and light.

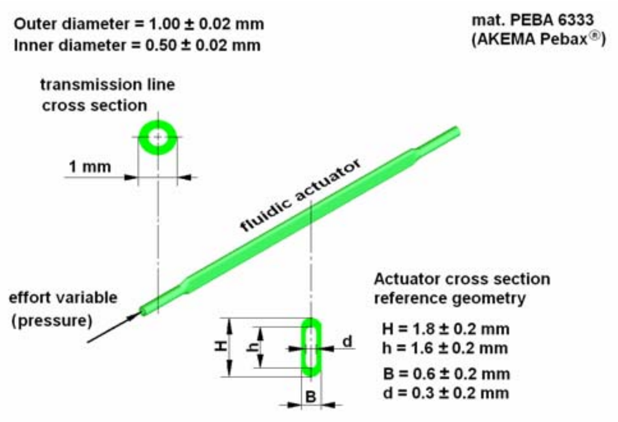


Fig. 3 Fluidic actuator

The design was conceived taking into account the manufacturing process and the robustness of the actuator. The case of a bi-phase fluid was also considered. The system is suitable for a closed loop control, which regulates the pressure *p* (effort variable) in the actuator.

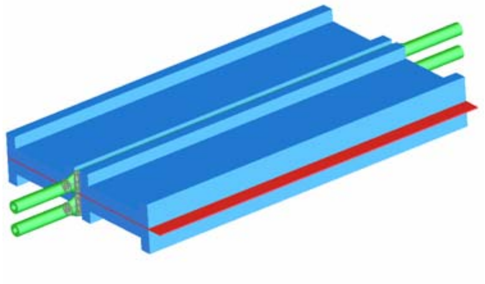


Fig. 4 Elastic joint with two Joint-actuators

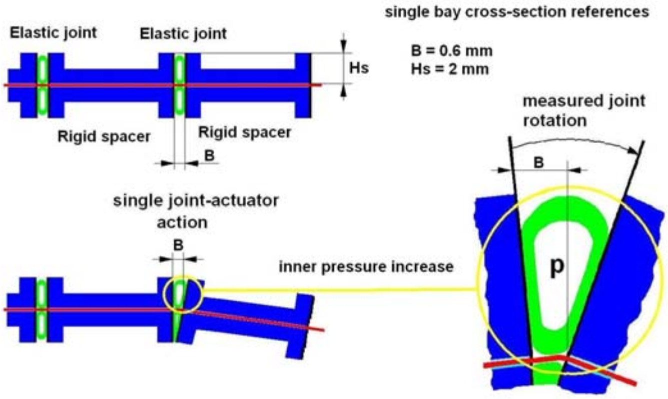


Fig. 5 Smart stick module, rigid spacer (blue), metal foil (red), fluidic actuator (green)

5 Prototype design

The length of the actuators which were designed was 30 mm. The elliptical section of the micro-tube (see Fig. 3) was repeated along the tube every 15 mm. The tube was bent between each two elliptical sections in order to convey the working fluid to different joints. Fig. 5 shows the rotation of a mechanism module induced by the deformation of the pressurised tube.

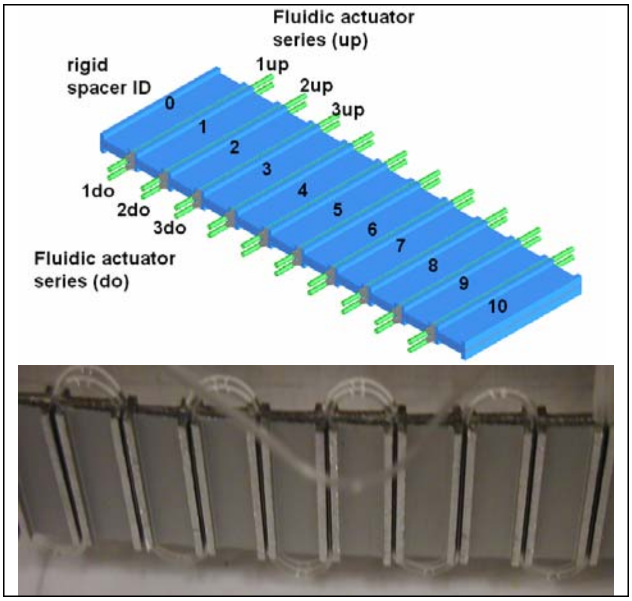


Fig. 6 Prototype with two series of fluidic actuators and 10 elastic joints.

The pressurised tube expands, bending the inner metal foil along the bay. Eleven rigid spacers are glued to each side of the metal foil to ensure that each joint-actuator works properly. In order to guarantee a controllable bending variation of the joint in both directions, the smart-stick module is symmetrical with respect to its longitudinal axis.

Multiple modules of the smart stick can be embedded into one structure. Fig. 6 shows both the design and the prototype which was built. This prototype has 10 elastic joints and 20 fluidic actuators.

The action of actuators on the same side of the mechanism make it bend with a constant curvature when no external loads are applied. By changing the position of simultaneously-actuated joint actuators, different shapes of the smart stick can be obtained.

6 Fabrication

The prototype was built using the components described in Table 2.

The procedure to fabricate the prototype is as follows:

1. The spacers, which had a “C” shape, were accurately glued to a flexible metallic joint (steel foil).
2. The elliptical shape of the fluidic actuators was obtained by compressing the micro-tubes between two parallel surfaces.
3. The fluidic actuators were positioned between two consecutive spacers. Friction and

compressive forces were sufficient to keep the actuators in their positions.

Table 2 Characteristics of the components of the smart stick system

Component	*Material*	*Dimensions*
Flexible joint (foil)	Steel	Length 94 mm Width 30 mm Thickness 0.1 mm
Spacers	Aluminium alloy Al-Mg series 6000	“C” shape: 8mm x 2 mm Thickness 1 mm
Micro-tubes (fluidic actuators)	AKEMA Poly Block Ammide (Pebax® 6333)	Outer diameter: 1 mm Inner diameter: 0.5 mm (see Fig. 3)

7 Experimental characterization

A smart-stick module was tested in order to obtain the relation between fluid pressure and joint rotation (water was used as the working fluid). An empirical model is useful to predict the behaviour of the smart stick in the control algorithm. Fig. 7 shows the elastic joint in vertical configuration obtained by fixing the rigid spacer (n-1). A vertical configuration was used to reduce the influence of the gravitational force.

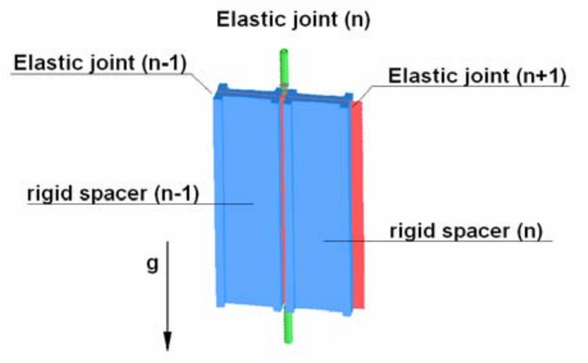


Fig. 7 Elastic joint in vertical position with only one fluidic actuator in the bay

Gravitational forces do not cause appreciable deformations as the inner metal foil has high flexural stiffness around an axis perpendicular to the axis of joint rotation.

During the experiment no external loads were applied to the elastic joint (except gravity). The free evolution of the system was analysed considering the action of only one fluidic actuator (Fig. 5). Eight complete cycles, obtained by increasing and decreasing the pressure in the fluidic actuator (range 0-1.2 MPa), were carried out. A linear model can be used when the relative pressure is between 0.1 MPa and 1.2 MPa.

8 Optical test bench

The first pair of rigid spacers (Fig. 7) is fixed and referred to the incoming ray trace. Fig. 8 shows the circular mirror fixed on the second pair of rigid spacers. In the nominal horizontal position, the mirror is perpendicular to the rays coming from a laser source fixed to a screen frame. Depending on the mirror angle, the reflected ray trace moves on the screen, by a measured distance M.

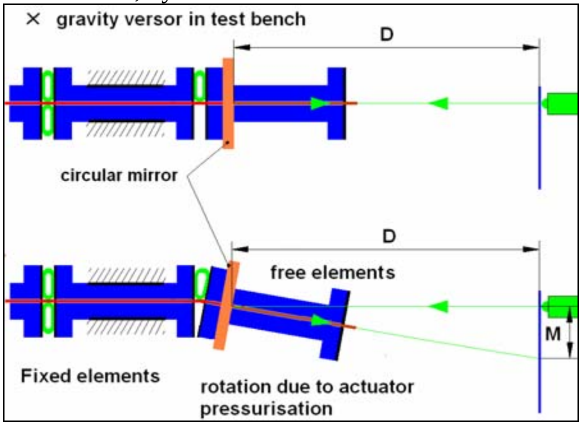


Fig. 8 Sketch of the test bench

8.1 Test bench overall dimensions

The distance (D) between the mirror and the laser source (see Fig. 8) is 2 m. The uncertainty associated with D, with a level of confidence of 68.3 %, is $\delta_D = 1.5$ mm. The uncertainty in the direct measurement of deflection distance (M), with a level of confidence of 68.3%, is $\delta_{\mathrm{M}} = 0.5$ mm.

The rotation of the smart stick module was computed by the following equation:

$$\theta = \frac{1}{2} arctg\left(\frac{M}{D}\right) \quad (1)$$

For small angles the previous equation can be simplified:

$$\theta \approx \frac{1}{2}\left(\frac{M}{D}\right) \quad (2)$$

The uncertainty associated with the angle θ (δ_θ) can be computed as follows:

$$\delta_\theta = f(\theta, M) = \sqrt{\left[\left(\frac{\partial\theta}{\partial M}\right)\cdot\delta_M\right]^2 + \left[\left(\frac{\partial\theta}{\partial D}\right)\cdot\delta_D\right]^2} \quad (3)$$

By substituting equation (2) in equation (3):

$$\delta_\theta = \sqrt{\left[\left(\frac{1}{2}\frac{1}{D}\right)\cdot\delta_M\right]^2 + \left[\left(-\frac{1}{2}\frac{M}{D^2}\right)\cdot\delta_D\right]^2} \quad (4)$$

the uncertainty of $\delta_\theta \approx 0.08°$ results (68.3% level of confidence).

Considering the assumption of the model and additional variables that were neglected, the

uncertainty i_θ of the angle θ can be assumed to be equal to 0.3° (99.7% level of confidence).
Fig. 9 shows the experimental setup (two fluidic actuators are represented).

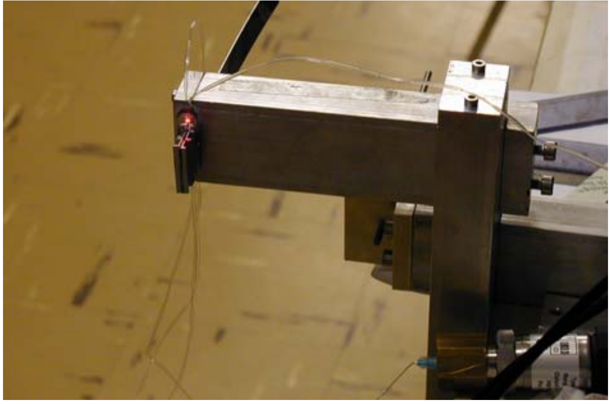


Fig. 9 Elastic joint in test bench

The uncertainty in the effort variable *p* (pressure of the working fluid) was i_p=0.013 MPa (99.7 % level of confidence) as declared in the calibration certificate of the pressure gauge which was used.

9 Test results

Several tests were performed in order to characterize the behaviour of one module of the smart-stick. Fig. 10 shows that the rotation of the joint can be approximated with a linear function (blue line). Fig. 10 concerns the use of one module of the smart stick having embedded only one fluidic actuator.
Fig. 11 shows experimental results for eight cycles using the same joint module. A fitting of the experimental data was performed in order to obtain an empirical equation that could be used for control purposes:

$$\theta = \theta(p) = 3.57\,p + 0.13 \qquad (5)$$

where the angle is in degrees and the pressure in MPa. The uncertainty (99.7 % level of confidence) associated with the slope of the function is ±0.13 and that associated with the intercept is ±0.24°.
A more accurate measurement procedure is required to appreciate the non-linear behaviour of the Smart Stick which is mainly caused by the fluidic actuators made of plastic (Pebax® 6333). Improvements are foreseen by using material with better elastic performance for the miniaturized tubes.

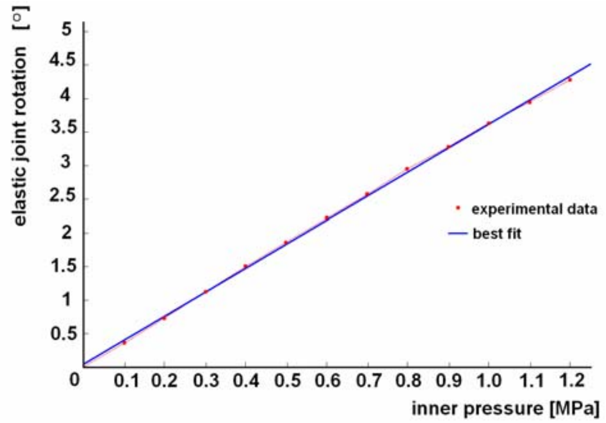


Fig. 10 Experimental results of one smart stick module with only one fluidic actuator. Red line: experimental results. Blue line: linear fitting curve.

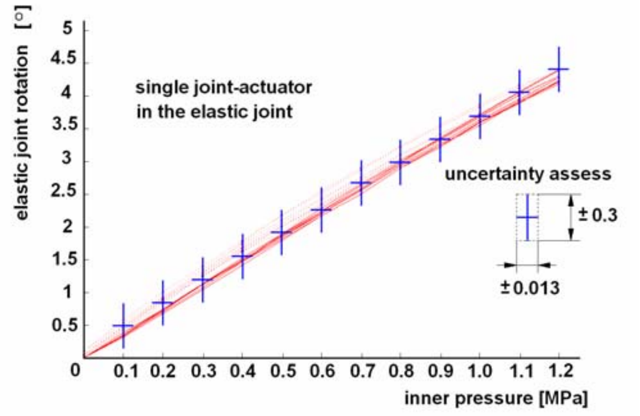


Fig. 11 Multiple cycles of one smart stick module (only one fluidic actuator)

10 Future improvements and designs

In order to increase the performance of the smart stick, a careful selection of the material employed is needed. The use of aerogel and lighter and more flexible materials for the joints is being considered. Improvement in the design will focus on multiple parallel mini-tubes which can bring more flexibility to the system.
The fundamental next step necessary to obtain a complete mechatronic breadboard for space applications will be the realization of the multi-module smart stick incorporating a closed fluid loop. Fig. 12 shows a sketch which presents the concept idea of a closed fluid loop.

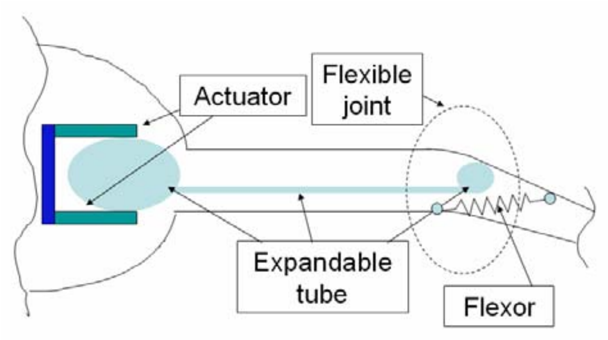


Fig. 12 Sketch of the closed fluid loop

The working fluid is confined inside a closed tube with expandable parts. The actuation is performed by squeezing one end of the tube. The actuation mechanism could include smart materials such as piezoelectric materials, shape memory alloys and electro-active polymers (Rossi, Carpi, Jeronimidis, Gaudenzi, Tralli, Zolesi, and Ayre, 2004). The use of a closed fluid loop, which allows us to overcome outgassing issues, is considered to be a critical point for a successful hydraulic space mechanism. Magnetorestrictive fluids could also be used in future work to improve torque performance.

The realization of a compact mechatronic system made of an electronic unit and a closed loop smart stick, integrated with the actuation unit, will also be promising for commercial applications including the toy market.

11 Conclusions

A novel flexible joint with an embedded actuator is presented in this paper. The novel mechanism is inspired by spider joints and hydraulic closed system. The proposed mechanism can be fabricated using traditional and inexpensive processes and methodologies. The modularity of the mechanism can be used to design joints for large displacements. The miniaturized prototype which was built and tested using an optical test bench showed the feasibility of the design and suggested improvements for a future design.

Acknowledgements

The work was carried out thanks to a NATO grant joint to the Italian Research Council (NATO-CNR 215.36). Thanks also to Susanna Valpreda for her valuable support in the bibliographic review.

References

Anderson, J. F. and Prestwich, K. N. (1975). The fluid pressure pumps of spiders (Chelicerata, Araneae). *Z. Morphol. Tiere 81, 257-277.*

Yoseph Bar-Cohen, Electroactive polymers as artificial muscles - capabilities, potentials and challenges, *Robotics 2000 and Space 2000. Albuquerque, NM, USA, February 28 - March 2, 2000*

M.E.Barik, Planetary Handbook , *Cambridge, 2000*

Haym Benaroya; Leonhard Bernold, M.Asce; and Koon Meng Chua, F.asce, Engineering, Design and Construction of Lunar Bases *Journal of Aerospace Engineering / April 2002.*

Blickhan and Barth, Stains in the exoskeleton of spiders, *Journal of comparative physiology, 157:115-147,1985.*

Cary Guffey, Victor R. Townsend, Jr., Bruce E. Felgenhauer External morphology and ultrastructure of the prehensile region of the legs of leiobunum nigripes (ARACHNIDA, PILIONES), *2000. The Journal of Arachnology 28:231–236*

Ato Kitagawa Hideyuki, Tsukagoshi Mitsuru Segawa, Proposal of a connective type active hose with many degrees of freedom for the rescue operation, *Mechanical and Control Eng., Tokyo Institute of Technology 2-12-1 Ohokayama Meguro-ku Tokyo 152-8552, Japan*

C.H.M. Jenkins, Gossamer spacecraft: membrane and inflatable structures technology for space applications, *Progress in Astronautics and Aeronautics, 2001.*

Rob Knight and Ulrich Nehmzow Walking Robots, A Survey and a Research Proposal" *Technical Report CSM-375 University of Essex October 2002*

C.Lira, F.Angrilli, S. Debei, Variable Structure Fabric and uses for it, *patent document WO02099172*, 2003.

Manton, S. M., Evolution of arthropodan locomotory mechanisms , *Pt. 6. J. Unn. Soc. (Zool.), 43, 488-556, 1958.*

Manton, S. M. Hydrostatic pressure and leg extension in arthropods *Ann. Mag. Nat. Hist. Series 13, 1, 161-82, 1958.*

D. A. Parry and R. H. J. Brown The hydraulic mechanism of the spider leg *Department of Zoology, University of Cambridge* (February 1959)

Parry, D. A. and Brown, R. H. J. The jumping mechanism of salticid spiders *J. Exp. Biol. 36, 654-664, 1959.*

D. Rossi, F. Carpi, G. Jeronimidis, P. Gaudenzi, A. Tralli, V. Zolesi, M. Ayre, Electroactive Polymers for actuation and sensing in space applications, *IAC, AIAA, Vancouver, Canada, 2004.*

Andrew T. Sensenig and Jeffrey W. Shultz, Mechanics of cuticular elastic energy storage in leg joints lacking extensor muscles in arachnids *The Journal of Experimental Biology 206, 771-784 © 2003 The Company of Biologists Ltd doi:10.1242/jeb.00182*

Andrew T. Sensenig and Jeffrey W. Shultz, Elastic energy storage in the pedipalpal joints of scorpions and sun-spiders (ARACHNIDA, SCORPIONES, SOLIFUGAE)" 2004. *The Journal of Arachnology 32:1–10*

Schworer, M., Kohl, M., Menz, W., Fluidic Microjoints Based on Spider Legs, *Conference on New Actuators, Bremen, 1998.*

Shultz, J. W., "Evolution of locomotion in Arachnida: the hydraulic pressure pump of the giant whipscorpion, Mastigoproctus giganteus (Uropygi). *J. Morphol. 210, 13-31. 1991.*

Stewart, D. M. and Martin, A. W., Blood pressure in the tarantula, Dugesiella hentzi. *J. Comp. Physiol. 88, 141-172, 1974.*

Wilson, R. S. and Bullock, J. The hydraulic interaction between prosoma and opisthosoma in Amaurobius ferox (Chelicerata, Araneae) *Z. Morphol. Tiere 74, 221-230, 1973.*

Whiskerbot: A Haptic Sensor Array for Mobile Robots

Martin J. Pearson*
Chris Melhuish*
Anthony G. Pipe*
Ben Mitchinson†
*Bristol Robotics Laboratory (BRL)
Bristol Business Park
University of the West of England
Coldharbour Lane
Frenchay
Bristol BS16 1QY
`Martin.Pearson@uwe.ac.uk`

Ian Gilhespy*
Mokhtar Nibouche*
Tony J. Prescott†
†Adaptive Behaviour Research Group
Department of Psychology
University of Sheffield
Western Bank
Sheffield
S10 2TN
`b.mitchinson@sheffield.ac.uk`

Abstract

This short paper summarises the aims, aspirations and progress to date on the Whiskerbot project, run jointly by Engineers at the Bristol Robotics Laboratory and Neuroscientists at the Adaptive Behaviour Research Group.

1 Introduction

Rodents can orient themselves and discriminate between surface textures using their array of mystacial vibrissae, or facial whiskers (Carvell and Simons, 1990). The vibrissae are the visible ‘front end’ of a sensory system that involves the interaction of numerous neural structures throughout the Central Nervous System (CNS) of the animal (Kleinfeld et al., 1999). Ultimately the mechanical deformations of the vibrissal shaft, as it interacts with the environment, are translated into information which the CNS can interpret and upon which generate appropriate action selections. The potential applications for such a sensor array in the field of mobile robotics includes the rapid orientation in confined, dark or visual occluded environments, for example, to assist with search and rescue operations following the collapse of buildings or mines. It also does not require the illumination of the environment to extract information, therefore it has a low power consumption and the ability to move covertly. The approach taken in this work differs markedly from previous whisker based robotic sensor systems (Russel, 1992), (Kaneko et al., 1998) which adopted a more abstract engineered interpretation of the whiskers.

2. Artificial vibrissae and Active whisking

Our artificial vibrissae have been formed using composite materials and have the ability to be actively moved in a manner analogous to natural whisking. The sensory apparatus of real vibrissae has been modelled and implemented using micro

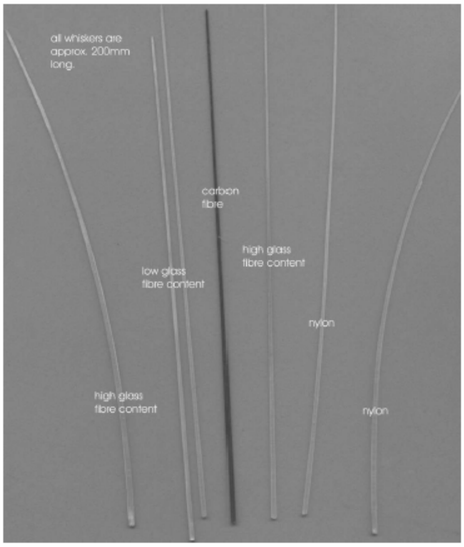


Figure 1: A selection of the artificial composite vibrissa formed using a curved mould and a straight mould. Experimenting with different materials

strain gauges and Digital Signal Processors. The primary afferents have been modelled using empirical data taken from electrophysiological measurements, and implemented in real-time using a Field Programmable Gate Array. Pipelining techniques were employed to maximise the utility of the FPGA hardware. The system is to be integrated into a more complete whisker sensory model, including neural structures within the central nervous system, which can be used to orient a mobile robot.

We have included the characteristic curvature and tapering observed in rodent vibrissae by machining an aluminium mould to form stereotypical composite vibrissae (see Fig.1). Two pairs of diametrically opposing micro strain gauges were bonded to the periphery at the base of the artificial vibrissae. The configuration of the gauges is such that deflections of the vibrissal shaft, when clamped at the base, will generate a proportional 2 dimensional strain measurement vector.

The ability of rodents to actively move their vibrissae, a behaviour known as whisking, has been suggested as highly instrumental in extracting both textural and spatial sensory information (Carvell and Simons, 1995). The ability to whisk the artificial vibrissae has also therefore been implemented in this model, a photograph of the first prototype is shown in Fig.2. The intrinsic muscles of the rodent facial musculature have been implemented using a wire, shape metal alloy called BioMetal©.

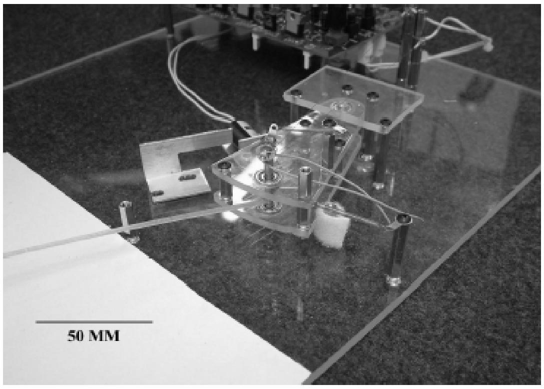


Figure 2: Prototype active whisking module using BioMetal to protract an artificial vibrissae

We intend to build multiple instantiations of the actively whisking artificial vibrissae units and place them onto specialised sensor chassis. Each chassis will hold 6 vibrissae, arranged as 3 pairs of opposing units representing the vibrissae protruding from opposing mystacial pads. Ultimately we plan to build 3 such chassis and stack them upon each other to form two 2 dimensional arrays of 9 vibrissae (3 X 3) projecting in each direction.

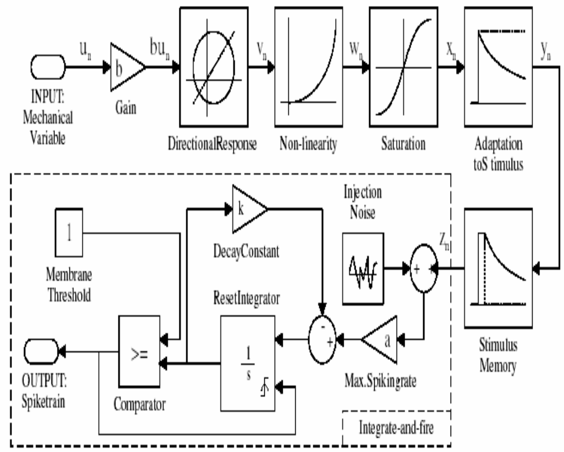


Figure 3: Functional block diagram of the proposed mechanoreceptor model (Reproduced from (Mitchinson et al., 2004))

3. The Follicle Sinus Complex (FSC) Model

The mechanoreceptor models themselves are simplified functional representations of large groups of actual cells. Indeed, what are actually being modelled are the Primary Afferents that are excited by the mechanoreceptors, translating this excitation in the form of spike trains to the trigeminal sensory complex (Brain stem). However, for clarity we refer to a single model as a single mechanoreceptor of a particular species with an associated Most Effective Angle (MEA) of sensitivity to the direction of mechanical deformations in the vibrissal shaft. When the vibrissae bends in a certain direction, the mechanoreceptors with MEAs 180^o to the direction of bend will become maximally excited due to the pivot point at the sinus. This directional sensitivity has been modelled along with a number of other features, as shown in Fig.3, such as adaptation to stimulus, saturation of activity and a simple leaky-integrate-and-fire model to translate the cell activity into a train of discrete spike events.

Fig.4 shows an abstract block diagram of the inter-processor architecture that implements the processing structure above. The TMS320F28xx series Texas Instruments DSP on each chassis samples the 6 vibrissae (using the on-chip peripheral ADC module) and computes the IIR filtering of the FSC model. The filtered mechanical variables (16 bit) from each chassis are passed, via separate SPI buses, to the FPGA, which updates all 720 mechanoreceptor models and sends the resultant spike trains to the 'brain stem' using a single SPI bus. The brain stem, in this case, consists of a model of the trigeminal sensory complex implemented on a matrix of real-time spiking neural network processor FPGAs presented elsewhere (Pearson et al., 2005). The central FPGA, modelling all 720 mechanoreceptors has been named the MechanoProcessor (see inset of Fig.4). It consists of a main sequencer module, 3 Mechanoreceptor Processing Elements (MPE) (each of which incorporates an SPI input bus) and a single SPI output module. The internal update period of the processor is 100µS to match the DSP sample rate, whilst the output update period is 500µS to synchronise with the 2KHz neural processors modelling the brain stem. The sequencer module, therefore, requires 2 separate synchronisation lines, 10Khz and 2KHz, in order to correctly coordinate the activity of the various concurrently operating modules of the system.

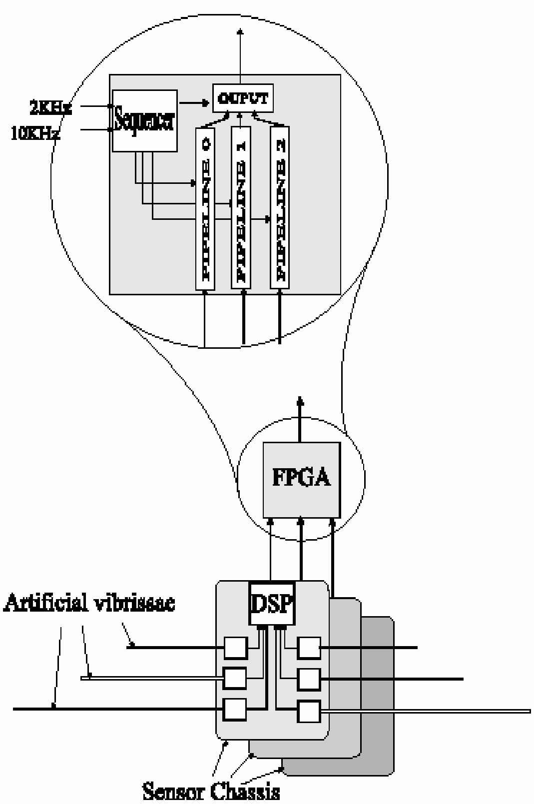


Figure 4: Block diagram of inter-processor architecture. (Inset: functional diagram of the Mechano-Processor)

4. Conclusion

The periphery of a biologically inspired whisker based sensory system has been designed and implemented. The form of the mystacial vibrissae has been scaled up and replicated using moulded glass-fibre. The artificial vibrissae have been mounted onto a platform that can sweep them in a manner analogous to the whisking behaviour observed in rodents. To extract sensory information from the vibrissae array, strain gauges were used that can derive a stable, 2 dimensional displacement vector of the vibrissal shaft. This information was used to drive an empirically based model of the follicle sinus complex found at the base of each vibrissa. This model has been developed using software and implemented in real-time using a combination of DSP processors and reconfigurable hardware (FPGA). The biologically plausible output from this system will be passed to an established spiking neuron model of the trigeminal sensory complex (brain stem) for further processing and feature extraction. Ultimately this will become a more complete, biologically inspired, sensory system which will be used to orientate a mobile robot in a real-word environment.

Acknowledgments

This work is funded by the EPSRC (Grant number GR/S19639/01). Our thanks is extended to Dr. Kevin Gurney and Prof. Peter Redgrave for their contribution to the overall project.

References

Carvell, G., Simons, D. (1990). Biometric analyses of vibrissale tactile discrimination in the rat. J. Neuroscience, 10:2638-2648.

Carvell, G. and Simons, D. (1995). Task and subject related differences in sensorimotor behaviour during active touch. Behavioural Brain Research, 12(1):1-9.

Kaneko, M., Kanayama, N., Tsuji, T. (1998). Active antenna for contact sensing. IEEE trans. on rob. and auto., 14(2):278-291.

Kleinfeld, D., Berg, R., and O'Connor, S. (1999). Anatomical loops and their electrical dynamics in relation to whisking by rat. Somatosensory and motor research, 16(2):69-88. Krichmar, J. L. and Edelman, G. M. (2005). Brain-based devices for the study of nervous systems and the development of intelligent machines. Artificial Life, 11:63-77.

Mitchinson, B, Gurney, K, Redgrave, P, Melhuish, C, Pipe, A, Pearson, M, Gilhespy, I, Prescott, T. (2004). Empirically inspired simulated electro-mechanical model of the rat mystacial follicle-sinus complex. Proc R Soc Lond Biol Sci, 271:2509-2516.

Pearson, M., Gilhesphy, I., Gurney, K., Melhuish, C., Mitchinson, B., Nibouche, M., and Pipe, A. (2005). A real-time, fpga based, biologically plausible neural network processor. Proceedings: International Conference of Artificial Neural Networks.

Russel, R. (1992). Using tactile whiskers to measure surface contours. In press., I., (Ed.), Proceedings of the 1992 IEEE International conference on robotics and automation, pages 1295-1299.

A Mechatronic Testbed for VOR Inspired Neuro-Control Experiments

Alexander Lenz
Bristol Robotics Laboratory (BRL)
Bristol Business Park
University of the West of England
Coldharbour Lane
Frenchay
Bristol BS16 1QY
Alex.Lenz@uwe.ac.uk

Thanushan Balakrishnan
Bristol Robotics Laboratory (BRL)
Bristol Business Park
University of the West of England
Coldharbour Lane
Frenchay
Bristol BS16 1QY
Thanushan.Balakrishnan@uwe.ac.uk

Abstract

This short paper describes our work to date, carried out as a close collaboration between Engineers, Neuroscientists and Biologists, in implementing a biologically plausible testbed for neuro-control experiments. The testbed consists of a high speed motorised camera-platform playing the part of the eye and surrounding musculature, and a motion system playing the part of the head in which the eye is mounted. The result will be a biologically plausible gaze-stabilisation system.

1 Biological Background

The vestibulo-ocular reflex (VOR) causes eye movements in the direction opposite to the head movement. It therefore serves to stabilise vision. It is closely associated with the flocculus, an evolutionarily old part of the cerebellum (J. Voogd and M. Glickstein, 1998), and the medial vestibular nucleus (MVN) of the brainstem. Three semicircular canals and two otolith organs in each labyrinth (this structure exists in each ear) act as a sensory input in addition to the vision system. The sensory signals can be broken down into yaw, pitch, roll (as well as linear motion or static tilt). The output is the horizontal, vertical and torsional movements of the eyes. Under 'real-life' conditions this comprises the concerted operation of numerous parallel pathways linking the ten (2 × 5) sensors with the twelve (2 × 6) muscles of both eyes. (M. Ito, 1998) One important concept to realise is that the VOR's stabilisation task is not driven by visual feedback since this would be much too slow (typically delayed 50-100ms) to compensate for fast head movements. Rather, the extraocular muscles are driven by vestibular output in a feed-forward manner, whereas the vision system acts as an error feedback for the necessary calibration.

2 Neuro-Control Architecture

It is currently understood that the cerebellar microcircuit takes simple motor commands and elaborates them to detailed and precise muscle control sequences. On the basis of error feedback from the motor performance, this elaboration is constantly fine tuned. In the case of the VOR, and with some abstraction, this can be looked at as an adaptive filter where the input is a copy of the motor command sent to the oculomotor plant and the output signal is added to the feed-forward filter already present in the brainstem. The training signal for the filter is the climbing fibre input carrying a retinal slip signal, representing the sensory consequence of inaccuracy in the VOR fine tuning (J. Porrill, P. Dean, and J. V.

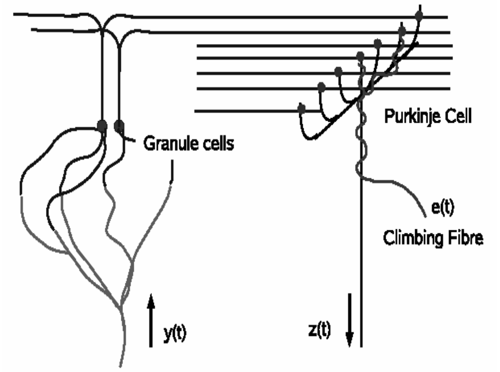


Figure 1: Schematic of cerebellar micro-circuit

Stone, 2004). The structure of the cerebellar microcircuit is shown in figure 1 and the control context within which it sits relative to control of eye movement is shown in figure 2.

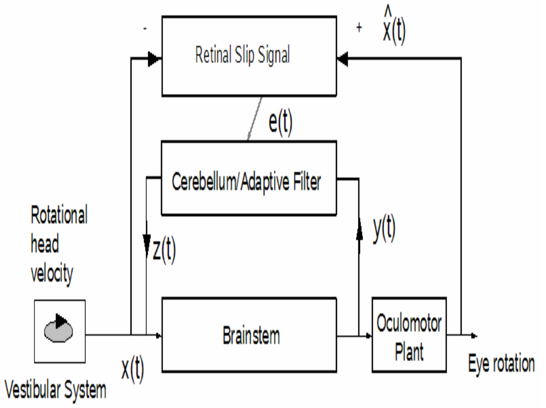


Figure 2: Architecture of cerebellum and brainstem control components driving the oculomotor plant

3 Engineering Equivalent of the Oculomotor and Control System

In order to test the proposed neuro-control algorithms we have been working on an engineering equivalent of the oculomotor system. Although stereo vision and paired sensors/actuators are typically found in animals, we limit our artificial eye mechanism to one drive system that can rotate a camera with 3 degrees of freedom; namely yaw, pitch and roll. We are not concerned with linear motion and tilt, concentrating only on the rotational VOR. Consequently, we also only employ one set of three gyros to emulate the sensing capabilities of the two labyrinths found in mammals.

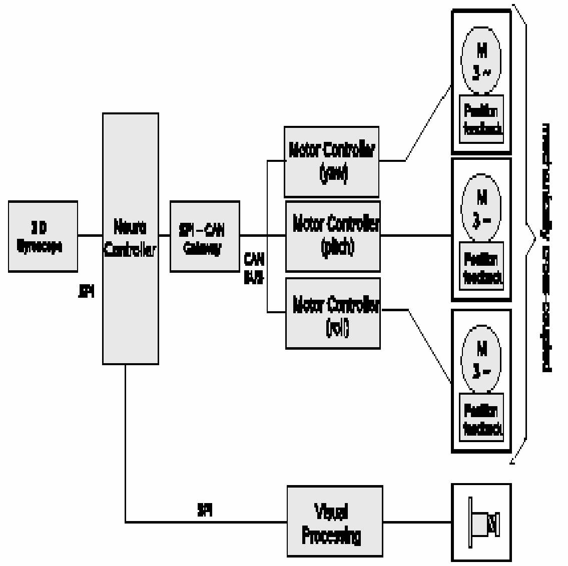


Figure 3: Overview, 3D Mechatronic "Eye"

The six extraocular muscles are substituted by three brushless DC motors, each driven via a dsPIC® microcontroller including position feedback via incremental encoders. This allows for accurate control and for us to change the dynamics of each axis, via filter algorithms executed on the embedded processor. Consequently, varying eyeball dynamics can be emulated using the same mechanical setup. This arrangement is illustrated in figure 3.

’Head movement’ is generated in a controlled manner via a 3D rotation platform that the mechatronic eye rests on. Here we also employ brushless DC motors and embedded microcontrollers. The platform allows us to rotate the motorised camera in a controlled manner and at the same time providing feedback about its current state, crucial for the evaluation of the performance of the control algorithms.

References

J. Voogd and M. Glickstein, “The anatomy of the cerebellum,” Trends in Cognitive Sciences, vol. 2, no. 9, pp. 307–313, September 1998.

M. Ito, “Cerebellar learning in the vestibulo-ocular reflex,” Trends in Cognitive Sciences, vol. 2, no. 9, pp. 313–321, September 1998.

J. Porrill, P. Dean, and J. V. Stone, “Recurrent cerebellar architecture solves the motor-error problem,” Proceedings of the Royal Society of London, 2004.

Fractals of Stability and Adaptability in Swarms: Thresholds for Emergent Effects

Olga Bogatyreva
The University of Bath
Mechanical Engineering Department
ensob@bath.ac.uk

Alexander Shillerov
ensob@bath.ac.uk

Abstract

Adaptability and sustainability are the most desirable features of many engineering products, especially for complex distributed systems where optimal and robust control of dynamic systems is required. Sociality in nature can be seen as a good prototype for investigating the distributed system adaptability. The model for diagnosis and prediction of complex distributed system behaviour is suggested. We have discovered a generating algorithm for the behavioural fractals of a social group of agents. It is based on unequal personal contributions to group behaviour and agents' differences in motivation and intention (which could be the subject of programming in artificial "swarms" in robotics). The diversity-generating algorithm of the social behaviour fractal is chaos/order balance that can be estimated by the normalised entropy index. We argue that the thresholds for emergent effects are defined by a combination of the two most general laws of nature – golden ratio and mirror symmetry.

1. Introduction

In the life sciences, a commonly held point of view is that death creates evolution towards species adaptation. Natural selection causes a lot of deaths, which supposedly aid the prosperity of the species. All living creatures try to avoid death. Engineers do not like the deaths of their products as well. In life of natural distributed system (for example the society) individual and group deaths are not necessary interconnected. The group can die or 'dissolve', but its members are sometimes accepted by other related groups (adoption of children is very common amongst social animals, although adults are rarely afforded the same treatment), or can create a new group instead. For example, one way to form a new ant colony is by splitting, known in ethology as sociotomy or social fragmentation. This process is very common amongst bees, ants and termites: workers, soldiers, and nymphs (amongst termites) migrate or march to a new nesting site where the fragment develops. Therefore, the death of a group does not necessarily mean the death of its parts. On the other hand, as the death of individuals is a very common event and individuals are easily replaced in a society, their death can even benefit group sustainability. So, in social life, adaptive management evolves guided by its own laws where death does not play a significant role. If death is no longer a creator in societies what drives the development of a society and provides group adaptability?

In the simplest scenario, adaptation can be seen as a thermostat or single-loop governor. Single-loop adaptation refers to adjustments made to maintain progress along a predetermined course. However, in order for organisations to successfully manage change, double loop adaptation is necessary. In an organisational context this sort of adaptation refers to an agent reacting to a situation by re-evaluating priorities, looking beyond the existing modes of operating and querying whether a more fundamental change is necessary instead of simply fixing problems so that everything returns to normal.

2 Social attractors and hierarchy

Interaction is the central point for our model of social behaviour. The main actor in the arena of life is an organism. All organisms exist in uncertain conditions: an unpredictable ecological environment, the variable behaviour of a neighbour, partner or the society they are in. This is where *uncertainty* and individual *intention* meet. So, the complexity in life is a result of not only actions but also intentions (awareness of behaviour of others), we assume that social "attractors" are produced by the mind and intentions. Organisms' (agents') individuality creates a chaotic component in their interactions; different organisms might follow the rules in a different way depending on their state of mind and previous experiences. From the chaos of individual interactions and behavioural diversity emerge social attractors through bottom up processes: goals that are common to all agents in a group.

Let's have a look how social attractors emerge and what changes they bring to individual minds and behaviour. In the case of the simplest level of consciousness, when creatures react only to the behaviour of others without understanding their intentions, interactions are built around the direct competition and dominance, which arise from the fight for resources. As the group grows larger the chains of direct competition become longer and the relationships between these creatures become increasingly distant. Uncertainty in society is caused by lack of each other understanding. One's own behaviour always seems determined and not at all chaotic, but the behaviour of others who must be interacted with can only be predicted with some degree of certainty. To reduce the amount of uncertainty for the sake of safety and comfort, agents need to be aware of not only the explicit behaviour of others but also their implicit intentions and motivations: agents need to know what others *think:* it is useful to explain our own goals and intentions to others to prevent conflict. If the group stays together for a relatively long time, the need for communication and flow of information increases. The more agents that control the flow of information, the more active the exchange of information about intentions: this makes a group more coherent. By chance some agents will have the same intentions and this will form a social "attractor" (fig. 1).

A group of agents with a common goal attracts other agents with similar intentions: the group becomes a team, which steadily grows. To realise the goal as a group is potentially much more efficient; sometimes it can be the only way to achieve a specific aim – group hunting, for example. Attractors emerge with coherence of intentions so goal synchronisation is a bottom-up process. Once the group has developed a hierarchical structure for the flow of information, attractors can be prescribed by the top-down way, as happens with ideology, fashion or philosophy in our society.

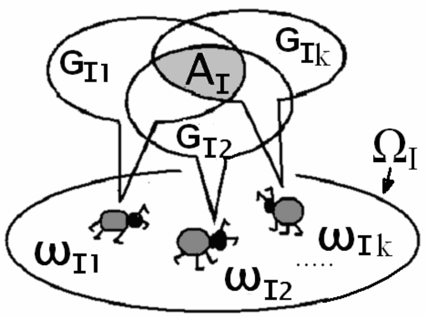


Fig. 1. How a social attractor (a common goal) forms the group.
Ω_I is a group of the first complexity level (such groups can join together and form a group of the second complexity level by the same principle of attractor formation); ω_{IK} –individuals, G_{IK} – goals/intentions; A – social attractor.

There are some ethological mechanisms to attract other agents to a group that has already formed around a social "attractor". These mechanisms are well known and widely used in everyday life. The most general is called 'social facilitation'; the more people that are involved in an activity, the more attractive this activity becomes to others. This could be described by the saying "monkey see, monkey do". This "monkey effect" is based on fundamental principles of behaviour such as imitation and the effects of being influenced by a crowd. It is not solely a feature of human society as ants also demonstrate it. For example, if a moderately active ant works with slow-working ants, the first ant reduces its level of activity. If the same ant were to work with extremely active ants, its activity level would increase (Wallace, 1979). Imitation is mostly behavioural patterning and requires mental activity, self-awareness and empathy; however 'crowd effect' is mainly an automatic physiological reaction and for this effect it is sufficient just to exchange with the censor information – to see what is happening without thoughts and analysis. Agents lose sight of their individual motivations and become driven by the crowd: for example gangs of young elephants storm human settlements without a visible reason, behaving just like football fans running onto the pitch after a game. War dances, parades, mass demonstrations and meetings dramatically increase individual aggression, leading people to do things, which they would never have done on their own.

How do agents with this common goal (the social attractor) regulate their interactions? Each person has his/her own ideas about how best to achieve a goal. This creates concurrence and direct suppression in their interactions and reduces the efficiency of their behaviour. One option is for an agent to pursue his own plans and force or attract others to join him. The success of such magnetism depends on personal will and motivation. The more different and highly motivated agents with a strong will, the more uncertainty in a group; it becomes harder to decide upon the best strategy for achieving a goal – potential leaders will be in a permanent quarrel. Ideally only one leader directs a group; he suppresses the will of others, showing them the path to follow. Others more or less support the leader, forming the hierarchical structure of their interactions dependent on their tolerance to their commander. Direct suppression and competition for the goal (which can be any desirable state: for example money, food, space, a mating partner) is replaced with the hierarchical framework that regulates access to resources and distribution of information in the society.

So, the process of hierarchy origin is top-down, rather than social attractor formation, which necessarily requires bottom-up support.

If this group acts as a separate unit, we will call it a first level social group. Such groups can cooperate with other groups to achieve some common goal (second level social attractor) forming second level social units. To make things easy to understand we will only describe the groups of the first level of complexity (that consist of individuals), and regard the second level group consisting only of the first level groups, but not the individuals. Such rule can be easily extended to any level of this "nested doll" hierarchy. Each person in a tribe (second level group) can perform different function if necessary and become a member of different groups of the first level: hunters, farmers or babysitters. When an organisation has already developed its structure, new comers are distributed throughout this system according to the social demands of the organisation. This phenomenon is known in socio-biology as 'social request'. In human society this social request is often driven by technological developments; amongst other animals it is the result of agents trading energy amongst themselves.

It is obvious that attractors cause centripetal forces in a group. The other goals and intentions that agents could express might contradict with their colleagues' goals, thus creating conflict. These goals are not in the interest of the attractor and are seen as spontaneous behaviour that brings uncertainty into agents' interactions. So, individual goals provide the centrifugal forces in the society. These forces are essential for the process of new attractors developing and therefore provide desirable adaptability despite their continuous threat to cause the group to disappear. To avoid the obliteration of the group there is a need to compromise with the use of different additional micro-attractors. This means that the hierarchy can create sub-attractors in a top-down manner which will entice deviant agents back to the group. This is the underlying process that provides the possibility for professional specialisation within a group: we all are different and perform different jobs, but are all still part of the same society.

As we mentioned above, hierarchy is based on the behavioural and mental diversity of agents and their differing motivations to achieve a goal. An agent's hierarchical rank is generalised characteristic of an individual as well as of the entire hierarchy statement, such as the style of management. In a second level hierarchy, the first level group itself has its own rank: we will show how to calculate it in the next section of this paper. From our calculations we discovered that the group rank is larger than the sum of its agents' ranks; this is the emergent effect. On the other hand, an agent has different ranks in different hierarchies (this is role-dependant hierarchy, the most complex sort of social stratification seen at least amongst primates, ants and hyenas). Hence, every agent within a given society carries information about all the hierarchies that exist within that society. This gives us hope that by studying a set of elements, it is possible to gain an idea of the group picture. As it is originally a bottom-up feature, an agent's rank can also be modified in a top-down manner (Bogatyreva, Shillerov, 1998).

3 What and how to measure in agent behaviour to judge the state of a system?

Any interaction of agents is asymmetrical simply because of agents' unique individuality. Dominance is the behavioural expression of such individuality. The agent position in the hierarchy reflects (and determines) its personal contribution and role in the information flow regulation within a society. Agent interaction in a group can be described hierarchy, where each agent has its rank according to its social activity and motivation.. In fact rank describes the intensity of agent coercion on the other agents.

Any social group has its structure – predictable interactions of agents – and has a lot of uncertainty in its behaviour as well. So, we need a model that can be used to describe a system whose behaviour is determined by both structural and random components.

Uncertainty is well described with informational entropy for the random variable with frequency p_i (Shannon, 1948):

$$H = -\sum_{i=1}^{n} p_i \log p_i \qquad (1)$$

However, we cannot use this approach directly as 1) the entropy method is only suitable for the description of random processes and we have a structural component to consider and 2) the entropy index depends on the number of elements in a system. This number can vary; for example agents can die or leave a group.

To enable us to use entropy for the uncertainty description for the random variable with numerical values r_k and probabilities p_k ($\sum_{k=1}^{n} p_k = 1$) we need to reduce it to the random variable with nominal values (Bogatyreva, Shillerov, 2005):

$$q_k = \frac{r_k p_k}{\sum_{k=1}^{n} r_k p_k} \quad (k = \overline{1,n}) \qquad (2)$$

The equation (2) estimates the contribution of each class of equivalent agents into the whole picture of interactions in a group. As $\sum_{k=1}^{n} q_k = 1$ we are able to calculate the Shannon's entropy :

$$H = -\sum_{k=1}^{n} q_k \log q_k \qquad (3)$$

The entropy is maximal when all the probabilities are equal or $q_k = \frac{1}{n}$ where n is the number of agents (Khinchin, 1957). Hence $H_{\max} = \log n$ characterises maximum uncertainty in the system. The index of current uncertainty varies $0 \le H \le \log n$ and obviously depends on n. This creates difficulties when we need to investigate a system during its development accompanied with the changing of agents' numbers, or when comparing two different systems. To avoid this obstacle it is useful to normalise H to yield a **Normalised Entropy Index (NEI)**:

$$h = \frac{H}{H_{\max}} = \frac{H}{\log n} \qquad (4)$$

This index varies in the interval [0;1] with any number of agents in a group and for any probability distribution and measures the amount of uncertainty in a system per one event. describes The maximal potential variety in agents' behaviour is when $h_{\max}=1$ therefore maximal **potential** order in a system (all possible knowledge about it), h – describes the current variety or residue uncertainty (unknown) .The redundant information in the system, is given by:

$$i = h_{\max} - h = 1 - h \qquad (5)$$

This is the measure of current order in the system structure of interactions (our knowledge).

Both indexes: normalised entropy (which describes the unknown, uncertainty) and normalised redundant information (which describes known, certainty, order) vary in the interval between 0 and 1 (fig. 2).

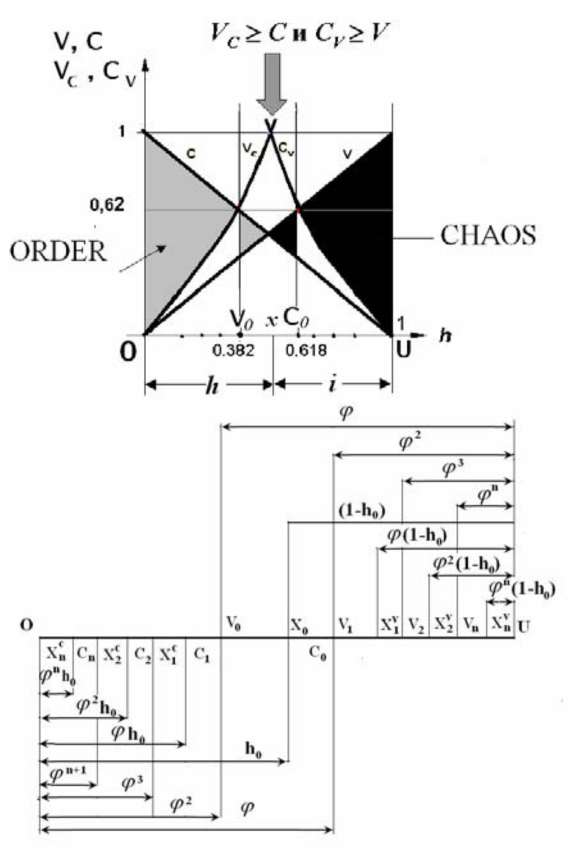


Fig 2. Diagram of relative measures of order and uncertainty in a system

On the segment OU (fig 2) there are two special points: C_0 (h=0.618) and V_0 (h=0.382) which happened to be golden ratio with its mirror symmetry – from the left and from the right.. In the first case (when the larger segment is on the left) the ratio (C_V) of certainty (C) to uncertainty (V) equals the amount of uncertainty in a system. Uncertainty (which describes unknown) is the geometrical mean of all system states (all potential knowledge) and order (certainty, actual knowledge). In the second case the ration (V_c) of uncertainty (V) to certainty (C) equals the certainty (C) in a group. Certainty, predictability (known) is a geometric mean of all system state (all possible knowledge) and uncertainty (what is unknown).

4 Fractal of interactions and qualitative emergent effects

Golden ratio is a singular geometrical proportion that saves the similarity of object shapes while it's successive division into smaller parts. It possesses the additive property according to which parts can be brought together into the wholeness without leftovers. It is the simplest linear fractal (Voloshinov, 2002). All behavioural interactions can be seen as combinations of chaos (uncertainty) and order (goal directedness). So, in what cases behavioural interactions can be seen as fractals?

Let's take arbitrary point X_0 (fig.2) on the segment OU with normalised entropy index h_0. This point divides the segment *OU* onto two segments. Dividing the segment which is on the left from the point X_0 by golden ratio with the following iterations with contractivety factor φ we will get the set of segments $OX_k^C\,(k=\overline{1,n})$ with length $\varphi^k h_0$ accordingly (fig. 2). These segments have the common point O, which allows expressing the sum

$$S_n = X_0U + \sum_{k=1}^{n} X_{k-1}^C X_k^C \qquad (6)$$

as the sum of the differences:

$$X_0U = OU - OX_0 = 1 - h_0$$

$$X_0X_1^C = OX_0 - OX_1^C = h_0 - \varphi h_0$$

$$X_1^C X_2^C = OX_1^C - OX_2^C = \varphi h_0 - \varphi^2 h_0$$

....

$$X_{n-1}^C X_n^C = OX_{n-1}^C - OX_n^C = \varphi^{n-1} h_0 - \varphi^n h_0$$

Hence, the sum $S_n = 1 - \varphi^n h_0$. Searching for the attractor of the dynamic fractal of stability we need to find the limits:

$$\lim_{n\to\infty} S_n = 1 - \lim_{n\to\infty} \varphi^n h_0 \qquad (7)$$

In case if $h_0 \le \varphi$ and the point X_0 either overlap with or is situated on the left of the point C_0 , $\varphi^n h_0 \le \varphi^{n+1}$. So $\varphi < 1$ we gain:

.

$$\lim_{n\to\infty} \varphi^n h_0 \le \lim_{n\to\infty} \varphi^{n+1} = 0 \qquad (8)$$

.

So, the limit of the sum $\lim_{n\to\infty} S_n = 1$ exists, therefore the attractor of the stability fractal exists– it is absolute order, when nothing changes and everything is known and predictable.

The key fractal property is self-similarity: when the geometric figure in which the same fragment repeats itself while scale diminishing. In the case of golden section each of the segments $X_0X_1^C, X_1^C X_2^C, ...X_{n-1}^C X_n^C$ is a redundant information which describes the patterns of agent behaviour that stabilise a group and it evolves towards the absolute order – the attractor of the stability fractal. Agents know more and more about the system and their expectations more and more match the reality providing the desirable predictability and order. But after extremely ordered state of development in life we should expect chaotic one simply because the system is disabled to adapt and change itself. Death appears to be unavoidable stage for any perfectness.

When $h_0 > \varphi$ and the point X_0 is situated on the right side of the point C_0 we gain:

$$\lim_{n\to\infty} \varphi^n h_0 > \lim_{n\to\infty} \varphi^{n+1} = 0 \qquad (9)$$

.

So, the limit of the sum $\lim_{n\to\infty} S_n$ does not exist, therefore **fractal of stability breaks down – this range is a kingdom of chaos. This is the emergent effect of the system at the current state and the point C_0 is the upper limit of fractal of stability existence.**

If we now make a golden section of the segment on the right hand side from the point X_0 with the following iterations with the contractivety factor φ, except for the segment UX_0 which has length $(1\text{-}h_0)$ we will get the consequence of segments $UX_k^V\,(k=\overline{1,n})$ having lengths accordingly $\varphi^k(1-h_0)$ (fig 2). These segments have a common point U, which allows expressing the sum

$$S_n = X_0O + \sum_{k=1}^{n} X_{k-1}^V X_k^V \qquad (10)$$

as the sum of the differences:

$$X_0O = UO - UX_0 = 1 - (1 - h_0)$$

$$X_0X_1^V = UX_0 - UX_1^V = (1 - h_0) - \varphi(1 - h_0)$$

$$X_1^V X_2^V = UX_1^V - UX_2^V = \varphi(1 - h_0) - \varphi^2(1 - h_0)$$

....

$$X_{n-1}^V X_n^V = UX_{n-1}^V - UX_n^V = \varphi^{n-1}(1 - h_0) - \varphi^n(1 - h_0)$$

Hence, $S_n = 1 - \varphi^n (1 - h_0)$. Searching for the attractor of the fractal of adaptability we need to find the limits:

$$\lim_{n\to\infty} S_n = 1 - \lim_{n\to\infty} \varphi^n (1 - h_0) \quad (11)$$

In case if $1 - h_0 \leq \varphi$ and the point X$_0$ either overlap with or is situated on the right side from the point V$_0$ we gain:

$$\lim_{n\to\infty} \varphi^n (1 - h_0) \leq \lim_{n\to\infty} \varphi^{n+1} = 0 \quad (12)$$

.

So, the limit of the sum $\lim_{n\to\infty} S_n = 1$ exists, therefore the attractor for the fractal of adaptability exists – it is absolute chaos, when nothing repeats itself and behaviour of a system is unknown and unpredictable.

Just like in stability fractal in the case of adaptability fractal the each segment $X_0 X_1^V, X_1^V X_2^V, ...X_{n-1}^V X_n^V$ is negative meaning of surplus information which we call surplus entropy – our wrong knowledge, guesses. It describes the process of loosing patterns of the stabilising behaviour, the process that benefit to a group adaptability. So, the fractal fragments iterate as redundant *entropy* and in fact describe the imperfection of knowledge (false predictability) and therefore are result of deviant behaviour patterns that are against the rules. The more patterns of stabilising behaviour breaks the more unpredictable the situation in a group is and the more uncertainty agents get after each experience, the less known becomes the global picture of a group. A system moves to the absolute chaos (attractor of the fractal of adaptability) and as the fractal of stability breaks down it becomes more and more clear that a system needs the new stabilisation inputs – new decisions and new goals and rules. When $1 - h_0 > \varphi$ and the point X$_0$ is situated on the left side from the point V$_0$ we gain:

$$\lim_{n\to\infty} \varphi^n (1 - h_0) > \lim_{n\to\infty} \varphi^{n+1} = 0 \quad (13)$$

.

The limit of the sum $\lim_{n\to\infty} S_n$ and therefore the attractor does not exist, so the **fractal of adaptability loses its fractal properties**. **This is the emergent effect of the current system state and the point V$_0$ is the low limit of adaptability fractal existence.**

While the development a social system confronts with two mutually exclusive demands: to safe its identity (or to be stable) and to evolve (or to be adaptable). This means that agents should provide group stability in social attractor realisation on one hand and to respond to the environment changes on the other hand. While social attractor realisation the behaviour patterns of agent interactions are stable and predictable. This expresses in the hierarchy of agents in a group: the more the difference between agents ranks the more stability and certainty in their interactions. Geometrically this can be shown with dynamic fractal of stability (fig. 3).

On the other hand environment is never stable and a system should evolve accordingly to survive. The adaptive behaviour patterns are provided by the dynamic fractal of adaptability and can be seen as the behaviour patterns that contradict to the social attractor realisation. For example a group is crossing the road, but one driver did not notice the red signal of traffic lights. The group stops and lets the car passing by, expressing the behaviour pattern "to stop" which contradicts with the main social attractor pattern "cross the street" being very adaptive pattern. The less is rank difference and therefore the ability to make a decision between agents the more uncertainty in their interactions and more adaptable group is. In previous example the group will definitely stop if each agent will act as an individual (with equally high ranks), rather then all of them rely on the single leader who might not notice the danger. Apparently, chaos is not a source for new stability fractals and therefore order. It is a signal of death, which might be desirable as well in the case when the group death does not mean the deaths of individuals. In case the group death is not desirable agents find another reason for their "friendship and unity". The points C$_0$ and V$_0$ on the segment OU (fig 3) are the thresholds for the current point X$_0$ (that has the meanings of NEI) on which fractals of stability and adaptability are defined accordingly. Therefore when h$_0$ is in the interval [0.382;0.618] both fractals describe the behaviour of a system (interactions of its agents): a system is stable and adaptable at the same time and its development can be called sustainable. We will call this stage as quasi-equilibrium.

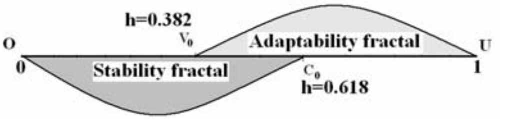


Fig.3. Thresholds for emergent effects

In the interval of h$_0$ [0;0.382) fractal of adaptability breaks down and only the process of ordering behaviour patterns can be described with the fractal of stability and the group moves towards the attractor which is an absolute order. Ordering pattern of behaviour achieve the emergent qualities and increase dramatically. This is the conservation interval. A group stagnates as it is too organised to evolve. The adaptability of such a system decreases as h →0, but the system becomes extremely adapted only to particular living conditions and absolutely "deaf" to environmental changes. It exists for the sake of itself, as a social attractor (the main purpose of agent gathering into the group) becomes a fetish. This position is unsustainable in a crisis as the system is vulnerable to any change and often dies.

In the case when h_0 is in the interval (0.618;1] the only acting fractal is the fractal of adaptability (fig.3). Fractal of stability collapses and this gives another emergent effect – uncontrollable disappearing of stabilising behaviour patterns this leaves systems non-adapted but still adaptable. The system looses the ability to control the agent interactions and the social attractor as a common goal expires and it is ready to undergo dramatic change. The group either dies or reorganises itself to "reincarnate" the fractal of stability on the different goals and stabilising behavioural patterns.

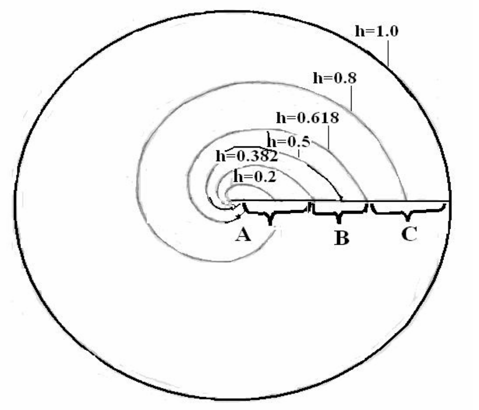


Fig. 4. Dynamic fractals of behaviour – 'trajectories' of goal directed behaviour in a phase space. Normalised Entropy Index indicates three intervals for three system qualities depending on the amount of uncertainty within it – interval of stability (A), corridor of quasi-equilibrium (B) and zone of bifurcation (C). Social attractor is in the origin (0,0).

So, as we see the most desirable situation for a group is when both fractals work together (fig. 3). The fatal change in the fractal either stability or adaptability brings crucial change in a system state and can be seen as emergent effect.

Any system realises its social attractor (common goal) via interactions of agents. Less uncertainty in the interactions takes a group to the social attractor quicker than the interactions do not have certain patterns (rules). On the other hand all interactions happen in time and the more time agents spend on interactions the closer they are to social attractor realisation. Let's consider the following generalized coordinates: h – measure of uncertainty of agents behaviour patterns in a group, Θ – time, ρ - "the distance" to the goal (attractor) realisation. The process of common goal realisation can be seen as a family of logarithmic spirals, those polar equation is given by:

$$\rho = he^{-(1-h)\Theta} \qquad (14)$$

Each logarithmic spiral describes the social attractor realisation under the particular conditions of uncertainty of agents' interaction patterns in a system. In the real life a group can change the amount of uncertainty in its elements interactions, but these changes are not continuous as correspond to the stability and adaptability fractals and therefore are discrete. It means that the real trajectory of social attractor realisation consists of the bits of logarithmic spirals (fig 4). The switch from one spiral to another happens in a blink of an eye when agents' ranks in their hierarchy of interactions change.

4 How to calculate emergent effects of stability?

Being part of a team, we believe that our "outside" rank is larger than if we were alone. Let's calculate the team rank for a group, which is sustainable enough so that the Normalised Entropy Index (*h*) is in the quasi-equilibrium range – h meanings are in the interval [0.382; 0.618] (fig. 2). Group rank is directly proportional to the sum of the agents' ranks but inversely proportional to NEI (Bogatyreva, Shillerov, 1998):

$$r_i = \frac{\sum_{k=1}^{n_i} r_{ik}}{0{,}382} \quad \text{and} \quad r_i = \frac{\sum_{k=1}^{n_i} r_{ik}}{0{,}618}$$

When a group is functioning optimally the group rank is larger than the sum of all agents' ranks between 1.618 and 2.618 times:

$$2{,}618\sum_{k=1}^{n_i} r_{ik} \ge r_i \ge 1{,}618\sum_{k=1}^{n_i} r_{ik}$$

The more chaos in a group (h→1), the less powerful the group is (when h=1 there is no emergent effect of stability at all). When a team is more organised, the more its rank differs from the sum of all its agents' ranks. In the case of maximal order, a group rank → ∞. This group or society becomes very dangerous and aggressive towards its environment (a favourite slogan in communist Russia was "We must not wait for a welcome from Nature, we should take everything from it") and also towards other societies and groups ("If you are not with us – you are against us!”). This happens in everyday life with religious confrontation and less frequently with clashes of ideology. An example of ideological confrontation is the case of fascism and communism. As it is too organised to accept innovations and react to a changeable environment, such a society becomes completely deaf, closed and expansive: becomes a real social and ecological disaster. The way such systems are dealt with unfortunately has a little effect on their strategy – we fight with them. There are apparently are some different methods to deal with deaf groups; we can dissolve them, increasing the normalised entropy index. We can use this index as a thermometer that indicates how close a society comes to the dangerous point of deafness and aggression. This will give the opportunity for quick thinking politicians to prevent conflicts and even wars.

5 Conclusions

It appears as though we have found the behavioural pattern generating algorithm, which is responsible for the diversity of social interaction and structure fractals (analogical to hands, arms and fingers, which are the result of interactions between cells during the development of the organism) that we see in nature (Prusinkievicz, Lindenmayer, 1996) and human organisations (Bogatyreva, Shillerov, 1999; Mitleton-Kelly, Papaefthiimious, 2000). The golden ratio is a geometric proportion that produces the similarity of forms, illustrating the Heraclitus principle of harmony: from everything to wholeness (bottom up), from wholeness to everything (top-down). The golden ratio is the single mathematical proportion that has an additive property, which allows a part to be completely subsumed into the whole. This is the main algorithm for structural and behavioural fractals in nature.

We suggest the algorithm for generating of any parameter diversity in a system, which includes interaction as an elementary ‘actor’. The polarisation of the intensity of action of different agents affects the group phase trajectory between three ranges. All the possible stages of a system can be realised and described through interaction, providing different pictures, golden ratio mirror symmetry defines the thresholds for emergent effects. When a system jumps into the bifurcation range, the fractal of stability pops up and switches on the emergent effect – old attractor (common goal) destruction and new social attractor appearance. If agents only “forget” the social attractor that kept them together the group destroys. It might reassemble again according to a new social attractor, but this will be a system with a new quality – qualitative emergent effect.

When a system jumps into the conservation interval, the fractal of chaos breaks down and another emergent effect switches on – the order becomes the attractor itself. The result is “deafness” and “blindness” of a system. It becomes like a cancer tumour – growing for the sake of growth. A system becomes an attractor for itself and all changes get frozen.

Only when both fractals exist a system is flexible enough to adapt without losing its identity. As none of the fractals breaks, there are no qualitative emergent effects at this stage. But there are quantitative ones, which can be calculated.

Acknowledgements

We are very grateful to Professor Julian Vincent for his interest in our work and always fruitful discussions.

References

Bogatyreva O.A., Shillerov A.E. (1998) Social Synergetics. –Novosibirsk, “Nauka”, – 292p.

Bogatyreva O. A., Shillerov A.E. (1999) Social process simulation: the way of human tribe origin. Humanitarian Science in Siberia, p14-20

Bogatyreva O., Shillerov A. (2005) Robot swarms in an uncertain world: controllable adaptability. Int.l J. of Advanced Robotic Systems, V. 2, Number 3, - p. 187-196.

Khinchin A.I. (1957). Mathematical Foundations of Information Theory. Dover Publications Inc.

Mitleton-Kelly E., Papaefthiimious (2000) Co-evolution of Diverse Elements Interacting Within a Social Ecosystem. FEAST 2000, Intern. Workshop “Feedback and Evolution in software and business processes”; Imperial College, London, UK, July 10-12.

Prusinkievicz R., Lindenmayer A. (1996) The algorithmic beauty of plants. Springer, 228 p.

Shannon, C.,*Bell System Tech. J.,* 1949. Reprinted in Shannon CE & Weaver W *The Mathematical Theory of Communication,* University of Illinois Press, Urbana, US.

Voloshinov A.V. 2002 About fractal aesthetics and fractals in art. In “The Synergetic Paradigm” Moscow, - 213-246 p.

Wallace Robert A (1979) Animal Behaviour: Its Development, Ecology and Evolution. – Santa Monica, California: Goodyear publishing company INC., - 590p.

Adaptable language understanding: can robots understand context bypass grammar rules?

Olga Bogatyreva
The University of Bath
Department of Mechanical Engineering
ensob@bath.ac.uk

Chris McMahon
The University of Bath
Department of Mechanical Engineering
enscam@bath.ac.uk

Abstract

Language is the only way for us to communicate with robots. It can be programming language, language of commands or ordinary speech. To achieve adaptable autonomous functioning of a robot while understanding commands within different contexts is strongly desirable. We suggest a biomimetic approach to the problem of context understanding evoking the model of animal behaviour within a social environment, where context plays crucial role. We propose a quasi-semantic model for a context which allows to take into the account words values as well as word frequency

1 Introduction

The field of natural language interpretation (a branch of Artificial Intelligence (AI) and of human-computer communication (HCI)) is mainly devoted to the automated processing of texts. This branch includes machine translation, semantic search for information, and speech processing. Here, purely linguistic methods are implemented for the analysis of the text semantics. The results of the analysis are represented in the form of a semantic network displaying a list of the most important words from the text and relations between them. A semantic network is a convenient text representation object often used in cognitive sciences. But a set of the created linguistic rules works well only with texts within the subject for which these rules have been developed. Thus the performed analysis depends strongly on the background knowledge of the analyzed field. This also implies that a human expert must be involved at the stage of the development of linguistic rules for a subject. Such an approach works well for the creation of systems that are utilized only in a single application field – therefore a single context. In order to successfully analyze texts from arbitrary fields, one needs to employ more general algorithms.

Another approach applicable to processing unstructured texts, artificial Neural Networks (NN), was developed with the hope that a homogeneous artificial processing media made out of connected elements similar to the neurons could indeed process information similarly to the human brain. Again, it has been demonstrated that systems based on this approach are capable of successfully solving simple analysis tasks. However, in general, a homogeneous processing media is not suited well for the analysis of linguistically structured information. Developing a new type of structured processing media is required for tackling this task. Summarizing, both AI and NN approaches provide important insights into the problem but demonstrate only limited success in practical applications and huge computational logistics. In fact, the most promising techniques for the analysis of natural language texts reside in the overlap of the two fields. To provide such overlapping we need the mathematical background for the procedure of the description of interaction between structural and random language characteristics. We suggest for such purpose a method that was developed initially for social systems behaviour description (Bogatyreva, Shillerov, 2005).

2 A sense recognition tool: normalized entropy index.

Human interaction with a computer while searching information in dialog contains both bottom-up and top-down processes. The human interaction with a computer while searching the information in dialog contains both – bottom-up and top-down processes. Methods of automatic clustering and classification – the way information is stored – and the ways the information is searched and retrieved are not totally compatible. If we want to implement a semantic search tool we should expect a semantic based storage of it. So, in computer modelling information clustering can be seen as top-down process and searching or information understanding (if we are dealing with a robot) mechanisms as bottom-up.

The concept of entropy is very popular in information theory. As an example consider some English text, encoded as a string of words and punctuation (string of characters). Since some characters are not very likely while others are very common the string of characters is not really as random as it might be. On the other hand, since we do not know the context and cannot predict what will be the next word, there is some subjective randomness for us. Claude E. Shannon suggested entropy (1949) as a measure of this randomness and reflects what regularities are still unknown. Shannon's formula can be derived by calculating the mathematical expectation of the *amount of information* contained in the information source. Shannon's entropy measures the uncertainty about the realization of a random variable. It thus captures the quantity of information contained in a message without taking into account of part of the message that is strictly determined (hence predictable) by structure which in fact represents a context. Examples include redundancy in language structure or statistical properties relating to the occurrence frequencies of letter or word pairs, triplets etc.

So the entropy method is the statistical method developed only for processes that do not include structural components and depend on the number of characters (n) -increasing this number always increases the entropy. The method of Conditional Maximum Entropy has been successfully applied to language modelling. The development of Maximal Entropy method occurred via two lines of research: Statistical Inference (Bernoulli, Bayes, Laplace, Jeffreys, Cox) and statistical modelling of problems in mechanics, physics and information (Maxwell, Boltzmann, Gibbs, Shannon). The objective of the first line of research has been to formulate a theory/methodology that allows understanding of the general characteristics (distribution) of a system from partial and incomplete information. In the second line of research, this same objective has been expressed as determining how to assign (initial) numerical values of probabilities when only some (theoretical) limited global quantities of the investigated system are known. E.T. Jaynes (1957) aided the Maximum Entropy (ME) formalism. The ME formalism is based on the philosophy of the first line of research and the mathematics of the second line of research. But Maximal Entropy methods are often *too demanding computationally.* Furthermore, the conditional framework does not allow expression of global sentential phenomena.

In our research we followed the first line using a mathematical model developed for social systems (Bogatyreva, Shillerov, 1999, 2005). The method of Normalized Entropy Index captures the thresholds when the sequence of signals has a particular sense and when the text (signal) changes the meaning.

It is important to estimate the *value of information*, which is a key point in sense recognition. If information has no value for us it might as well not exist: we will not notice something if we did not expect to see it or it does not hold any value for us. A maths textbook, left on the grass in the park, holds no meaning *as a book* for an ant: an ant will see it merely as an obstacle to be climbed or avoided. However, for a detective this book could be a clue to a crime which happened in the park; for a maths student, finding this book before an exam could be of great use. He who searches will find: this refers to an intention to notice or find as an individual has the goal of obtaining certain information that is of value to him. In other words, all interactions happen in a certain global context. We should include the phenomenon of value into the process of information exchange with robots/computers. Without this phenomenon it is not possible to create a context, which is absolutely essential. We are only able to find new information if we know that it is there for us to discover. What a system is able to recognise is limited by its past and current situation, so how is a system ever able to find new information that lies outside the boundaries of experience?

The problem of the Semantic Web is still a philosophy with a lack of practical issues. And it will remain the philosophy for ever because we cannot imbed in the computer the phenomenon which is only specific for a living system – intention. This phenomenon is the key one and without it the idea of artificial intelligence will never come to an engineering product. But the phenomenon of intention is already built into our language grammar. You can see this on the example of a language grammar where the word order defines the importance of the word. “A cat is sitting on the fence” is a sentence about a cat (context 1), and “A fence on which the cat is sitting” is about the fence (context 2). So, if you type the word “cat” and “fence” the search engine will retrieve you both sentences. In the conditions of great information abundance this is not what we want. We can use completely different words to express the same context and a computer should be able to “sense” this. This is a random component of a context – the frequency we meet a word or its synonym with the same meaning in the text. To deal with structural and random component in one method in fact means the ability to manipulate with a contexts. But the language is not the only way to encode sense and to teach the computer to extract sense from language might be not very optimal way to go. To manipulate with contexts as separate from language entities with embedded intentions might be a solution.

The intention phenomenon is the main issue that distinguishes human and computer "thinking" – a computer deals with data (all data are of equal value for the computer), we are dealing with information that has values for us. Information as the main actor on this scene arrives only when we start to search for it – our intentions provide values for the words we type in the space for search in Google. Sense does not exist without values and intentions because only the set of values define the contexts. We are trying to embed language understanding by building the Semantic Web, but we should in fact embed some language independent sense recognition system. This will sufficiently speed up the process of automatic language understanding.

3. Mathematical method for context recognition

The problem of semantic search is both – task formulation (question) and information retrieving problem (answer). So it requires methods of language independent semantic problem formulation and language independent information retrieving. Ideally we would like to have the search engine to recognise and match the contexts in unstructured environment. We used a method that can estimate the borders of any context and retrieve the information accordingly. It allows effective information retrieval without preliminary text/information mark up.

For our method preliminary classification is not essential and can even limit the search of requested contexts with the CONTEXT under which the classification was done. Signal of linguistically expressed patterns that represent the context can be seen as a fractal of stability, described in the paper presented at the same proceedings of this conference (Bogatyreva, Shillerov, 2006).

In the context of the information theory, let us consider the search problem as a formal system A, where the key words $a_i\,(i=\overline{1,n})$ are its "elements". Before searching we need to define the subject of our interest. We suggest that this subject is in fact the problem definition. We can describe the problem using different words and their synonyms. Each word has different value in contribution to the whole context of the problem and this contribution can be described by its rank. We propose to arrange words in a specific quasi-semantic structure, which can be expressed as "things do things somewhere somehow". This formulation was adopted successfully from the innovative basic framework for the classification of biological data for engineering use (Vincent, Bogatyreva, Bogatyrev, Pahl, Bowyer, 2005). The value of each word is ascribed by rank in the context of the current search by the searcher. We ask the searcher to mark each key word as extremely significant, or very significant, or significant or less significant; in mathematical model these marks are transformed into scores 100, 50, 25 and 1 accordingly. Such score range was developed as a result of expert problem definition of his own paper. So, each word a_i will has its rank x_i evaluated from the user's point of view. The probability of a word appearance in the document is defined as

$$p(a_i)=\frac{v_i}{\sum v_i},$$

where v_i represents each word frequency in the tested document. As a result we have for the random variable with a numerical value which includes the target word value rank and the word frequency in a tested document and a random component of a context. To enable us to use this value within an information theoretic approach it is necessary to reduce the variable with numerical value it to the variable with a nominal value (Felinger, 1985; Bogatyreva, Shillerov, 2005). In this way we can effectively remove the barrier to merging structural and stochastic components of a context. As a result we have a finite scheme of events/words associated with system A:

$$\mathrm{A}=\begin{pmatrix} a_1, & a_2, & \ldots & a_n \\ q(a_1), & q(a_2), & \ldots & q(a_n) \end{pmatrix}$$

where $\{q(a_i)>0, i=\overline{1,n}\}$,

$$q(a_i)=\frac{x_i\,p(a_i)}{\sum_{i=1}^{n} x_i\,p(a_i)} \text{ and } \sum_{i=1}^{n} q(a_i)=1$$

To measure the quantity of uncertainty connected with the scheme we can use the information entropy index as it is used in information theory to represent the measure of uncertainty in the transmission of information in the message:

$$H=-\sum_{i=1}^{n} q(a_i)\log q(a_i)$$

To avoid the dependence of the entropy index on the number of key words found in the document, we used the normalized version of entropy (NEI)*:*

$h=\dfrac{H}{H_{\max}}$ For any nominal value of word distribution $q(a_i,)$ h varies in the range [0;1] and is the measure of uncertainty of the document value (relative measure of lack of interest) to the purpose of the search. In case when $q(a_i,)=q$ in the scheme A $H_{max}=logn$ or h_{max} = 1. So, the redundant information 1-*h* is the measure of document value for the purpose of the search. In fact this index is able to estimate the balance between randomness and order in a text. Karlheinz Stockhausen who studied information theory with Werner Meyer-Eppler discovered chance as an artistic problem, when he used random generators to decompose texts and computed the redundancy of these texts. (R /Hmax * 100%, H being the information, R - redundancy = Hmax - H). Stockhausen then started to compose music based on statistical criteria. What he wanted to achieve was to create a synthesis of random processes and deterministic processes as developed by serial compositional methods. He thought he would be able to find an "offspring of rich and lively new music, which would allow to experience order on a wide scale from 'out-of-control' and utmost organisation." (Werner, 1968). This is just what we want with context recognition. We now introduce two linear functions: a function of relative lack of interest or value -worthlessness (*V*=*h*) and function of positive usefulness or value - valueability (*C*=1-*h*) to the document for a user (these two values always sum to 1). While the normalized entropy index increases, the ratio between worthlessness an valueability is increasing

$$V_C=\frac{V}{C}=\frac{h}{1-h}.$$

At the same time, the opposite ratio is decreasing:

$$C_V=\frac{C}{V}=\frac{1-h}{h}.$$

As we can see from the diagram (fig. 1) the NEI meanings corresponding to equations $V_c=C$ and $C_v=V$ divide the normalized entropy interval into three ranges: [0; 0,382), [0,382; 0,618], (0,618;1] .

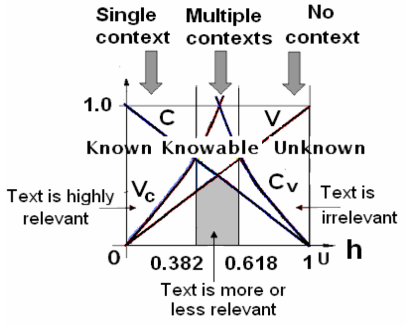


Figure. 1. Normalized Entropy Index ranges define the “borders” of a context we search in a text.

Table 1. Quasi-semantic units and context matching

Quasi-semantic unite		Rank of value	Word frequency in the text		
			Text 1	Text 2	Text 3
Things	Robot	100	0.6	0.7	0.02
	Swarm	60	0	0	0
Do things	Adapt	80	0.2	0.02	0.08
	Repair	20	0.05	0.03	0.7
Somewhere	Space	50	0.1	0.1	0.1
	planet	5	0	0.05	0
Somehow	interaction	50	0.05	0.05	0
	Decision-making	10	0	0.05	0.1
NEI (h)			0.55	0.29	0.8

If the estimated NEI is within the range of (0,618; 1] the document is worthless for the user (irrelevant to the subject of the search). If NEI is in the range of [0,382; 0,618], the document might be useful to consider, and if NEI is in the range of [0; 0,382) the subject of the document is highly relevant.

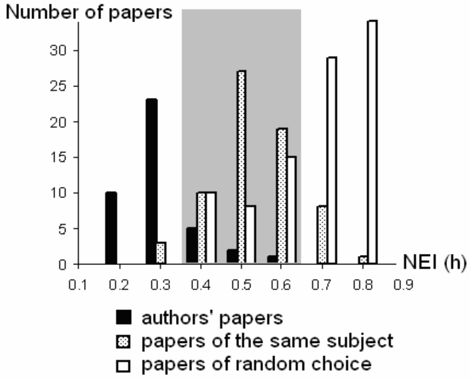


Fig. 2. Measurable boundaries between the known (easy recognisable context, knowable (allows variations and depicts the situation with multiple contests) and unknown (that does not match the contexts we expect).

For example, we need to find documents relevant to the subject of our interest (Table 1). We will formulate the goal as a semantic unit based on the description framework: “Things do things somewhere somehow”: For example “robot swarm adapts and repairs itself in space or planet by interaction and decision-making” (table 1).

In this example, we will allow at least two synonyms to each category of words. The structural component of our model will be defined by the rank of value in a context of our search that we give to each word. The random component in this case is the probability of finding the data in the electronic databases (word frequency in the tested documents). By calculating the Normalized

Entropy Index we can easily detect the relevance of documents to our semantic scheme. In the example shown text 1 is more or less relevant to the purpose of our search – the *h* index is in the quasi-equilibrium range of certainty and uncertainty of the document value. Text 2 is highly relevant – the *h* index is in the zone of high certainty. Text 3 is irrelevant to the semantic scheme of our interest – the *h* index is in the zone of high uncertainty (Fig.1). We also can use one of many independent semantic schemes (Table 1) to describe our problem. Likewise we can combine them. For example in a scientific geological approach it might be appropriate to use "Pumping water into the oil well increases its productivity" (scheme A); likewise for an ecological approach (minimisation of environment pollution) it might be appropriate to use "Water pollution reduction in the oil industry" (scheme B). To combine these two approaches into one semantic scheme, we need an estimate of NEI by multiplication of A B, $h(AB)$ to find the relevant documents.

We have tested our method on the example of the paper collection of the Department of Mechanical Engineering of the University of Bath. We analysed 125 papers from 15 people in the Department and 116 papers from a random selection from the Internet using the same key words as experts defined. Before the procedure of the method evaluation we adjusted the ranking system (giving the rank of minimum word significance = 1; max rank of value = 100) to such a way that NEI will take the author's (experts) own papers into the interval of high relevance; papers from the same scientific subject into the middle range of relevance and papers we choose randomly in the Internet to the range of irrelevance. After this we did the evaluation of the method the result of which is shown on figure 2. Papers tested by NEI formed 3 groups according to their relevance to a context described by the searchers (figure 2). The actual relevance of the papers retrieved was verified by experts as well to check the accuracy of the NEI method.

4 Conclusions

The conclusions from this work are as follows:

I. NEI method is as simple and cheap in document processing as the most popular key-word search. Although simple it possesses all the possibilities that all the most progressive methods for context recognition can offer.

II. The additional possibilities of the method compared with others are that it does not require preliminary marking or clustering and needs minimum document pre-processing.

III. Quasi-semantic structure is universal and only words that fill it (as well as their synonyms) define the specific context – a hierarchy of words values. Linguistically structured information can be transformed in to the hierarchical structure of each word semantic significance weight rather than looking for the word weight in the interaction with other words according to grammar. It can be used as a translator of language grammar dependent context into an entity for a computer or robot to work independently from the language structure. The refinement and further developing of this model is the actual challenge. It can also be combined with problem definition and problem solving tools.

IV. There is no single method that uses the same mathematical background, which gives the possibility to achieve the advantages of both the AI and NN methods and to get rid of shortcomings (AI – a set of the created linguistic rules works well only with texts within the subject for which these rules have been developed; NN – homogeneous processing media is not well suited to the analysis of linguistically structured information).

Acknowledgements

The research reported in this paper is undertaken at the Engineering Innovative Manufacturing Research Centre (E-IMRC) in Bath University funded by EPSRC under the grant GR/R67507/01. Thanks are due to our colleagues in the Department of Mechanical Engineering, especially to Mansur Darlington and Steven Culley, for helping to adjust the model to the linguistic domain.

References:

Bogatyreva O. A., Shillerov A. E. (1998) Social Synergetics. Novosibirsk, Russia, 291p.

Bogatyreva O., Shillerov A. (2005) Robot swarms in an uncertain world: controllable adaptability. Int.l J. of Advanced Robotic Systems, V. 2, Number 3, - p. 187-196.

Bogatyreva, Shillerov, 2006. Fractals of Stability and Adaptability in Swarms: Thresholds for Emergent Effects. Proc. of AISB'06 Conference on Adaptation in Artificial and Biological Systems, in press.

Werner Meyer-Eppler (1968) Musical Communication as a Problem of Information Theory, *die Reihe*, VIII, Bryn Mawr: Theodore Presser Company, p. 10.

Felinger A.F. (1985). Statistical Algorithms in Sociological Studies. "Nauka", Novosibirsk.

Jaynes, E. T., 1957, Information Theory and Statistical Mechanics, *Phys. Rev.,* 106, 620.

Shannon, C.,*Bell System Tech. J.,* (1949). Reprinted in Shannon CE & Weaver W *The Mathematical Theory of Communication,* University of Illinois Press, Urbana, US.

Vincent J., Bogatyreva O., Pahl A-K., Bogatyrev N., Bowyer A. (2005) .Putting Biology in TRIZ: Biological effect database. The J. of Creativity and Innovation Management, V.4, N 1

Biological and Engineering Parts of Biomimetics: “Software” and “Hardware” Adaptability

Nikolay R. Bogatyrev[†]
[†]Division of Visual Science, Institute of Ophthalmology,
University College London, London, UK
n.bogatyrev@ucl.ac.uk

Abstract

Biological and technological systems often follow different evolutionary pathways. Both of them – biology and technology – have advantages and limits. The relatively young biomimetics tries to use the prototypes from living nature to design the new technological products. But as there is no one single ultimate theoretical and logical methodology for this transfer it is very important to keep in mind the profound differences between these realms – biology and technology. Taking in account these differences, biologist and engineer will be able to make the correct and relevant prototype to solve the problem. Also to find the best or the optimal paragon from the nearly endless variety of living systems for biomimetic design is a great challenge. At the same time, if biomimetics makes the really profound and serious integration of biological principles to the design of artificial engineering systems, it will give hope that human technology will be more sustainable and less damaging to the environment of our planet.

1 Introduction

Natural evolution of biological systems (biological macroevolution) clearly shows growing importance of informational (psychical, communicational, behavioural, “soft-ware”) part of life. Those systems, which failed to have the advanced “soft-ware” eventually appear to be defeated. Gigantic animals armoured with huge offensive and defensive “weapons” lost their competition with the less armoured, but much more advanced behaviourally animals. More intellectual systems give not only instant response, but arrange long-term adaptation of *itself to the environment* and/or adaptation of the *environment to itself*! Any morphology (“hard-ware”) evolution without comprehensive “soft-ware” support finishes a dead-end.

Adaptation of the environment (as a whole, its parts, various objects and items) eventually has lead to the appearing of tools, constructions (nests, burrows, beavers’ dams, etc.) and finally to human technologies with the most sophisticated artefacts. This artificial “nature” is the contemporary environment at least for humans in the modern society. If we consider the evolution of human technologies throughout the history, we will also notice that hyper-development of any “hard-ware” (material side of any artefact) which was not supported by the relevant and adequate “soft-ware”, finally lost the competition and gave way to the smarter and more “informationally saturated” technologies. (There are numerous examples from the history of armours and weapons, energy-supply technologies, etc.).

Currently we observe incredibly fast development of various computer and telecommunication technologies. But the gap between this “soft”-part of the existence of our civilization and “hard”-part of it is growing. One of the most impressive examples of this is that even the recent super-fast trains still employ the rail-road track, which has been designed in the XIX century and the width of it is equal to the standard width of the Roman chariot’s wheel span. And this wheel span was dictated by the size of the horse’s croup!

Thus the fast evolution of the informational side of the society demands the adequate "hard-ware". That is why we observe the dramatic advance in biochemistry, nano-technologies, genetic engineering, pharmacology, etc. Most of these technologies operate at the smallest scale. But we notice that the level of organism and various and numerous super-organism levels were still staying in the shadow until the most recent times.

Thus appearing of biomimetics within this niche (organism and super-organism biological engineering and engineering biology) was inevitable (Bogatyrev, 2000, 2004). Now we see the roots and causes of biomimetics' evolving: it is the necessary step in the evolution of systems, which often takes place as an alternating process of "hard"- and "soft"-stages. It is easier to imagine now the position and role of biomimetics among other sciences and engineering disciplines. To create the comprehensive theory of biomimetics it is necessary to take in account the peculiarities of the

1) laws of non-living nature development (e.g., physics, chemistry);

2) laws of living nature development (biology *sensu lato*);

3) laws of super-systems and mixed systems development (e.g., distributed living systems and conjugated living and non-living systems which are studied by population genetics, ecology, ethology, evolutionary biology, etc.).

4) laws of non-material systems development (e.g., language, culture, art, etc.)

We should also keep in mind that biomimetics first of all is the field of engineering and thus it should follow the laws of engineering development (for the logical framework of many branches of engineering it is worth employing the TRIZ system (Altshuller, 1973), which is appeared to be applicable not only to traditional branches of engineering, but to biomimetic field as well). Taking in account the laws of development (not only *evolution*![1]) of living and non-living artificial systems within one engineering domain is the real challenge! It should be also mentioned that there are many deviations of various types of evolution, which are totally out of rational ground – numerous cultural patterns, e.g. fashion, social traditions, etc. In biological systems we can observe the same situation – very bizarre acoustic and/or visual signals in the system communication displays in courtship, for instance. Now it is obvious that biomimetics should comprise the most advanced features from the biological principles and vast engineering experience. Let us consider and compare some examples of the typical and the most general features of natural living and artificial non-living systems (see table 1). It is not completely clear and obvious what trends and sides of natural living and artificial non-living systems should biomimetic devices comprise. But for the first instance it is useful to extract the most essential and advantageous characteristics from our table. This is a list of examples (A and B are the columns in table.1):

1-B: space and military robots;

2-A: closed artificial ecosystems, microcosms, permacultural ecosystems;

3-B: recent progress in soft-and hard-ware, telecom technologies;

4-A: low energy permacultural artificial systems;

5-A: slow tillage, grass cutting, transportation economise energy, can be more convenient for customers/passengers, reduce ecological damage to wildlife, eliminate idle periods and other losses;

6-A, B: both trends can be important – cheap short-living numerous disposable units always must go parallel with more expensive, but durable, robust and reliable ones;

7-A: local alternative energy supply must ensure the centralized energy supply. This local energy is to be produced and consumed on site in the exact amounts according to the concrete demand;

8-A, B: the proportion of participation of robots in industry should include not only economical effect, but also cultural and social consequences – unemployment, solitude of elderly people, etc.;

9-A, B: mono-, bi- and poly-systems could have advantages under different conditions – distributed systems of sensors can substitute the mobility of a system, but large poly-system can create mono-system at the super-level – any structure made of the solidifying foam);

10-A, B: traditional fragmentation in engineering improved management and control, but the recent processes in technology mimics the live reproducing, this can be very helpful within the remote autonomous robotic missions;

11-A: diversification of energy supply, road net-work, means of transportation, resources supply improve the reliability of a system;

12-A, B: impulse effectors can be appropriate, but as a poly-system they can provide smooth action and/or even combined with the rotation.

Currently there is no one unified universal biomimetic method of design. But as a

---

[1] There are at least three more types of active transformation within the biological systems – ontogenesis, succession and history, which are often forgotten and omitted.

powerful tool for this task it is possible to employ TRIZ with the appropriate "consulting" at every step with the relevant (analogous or homologous) mechanisms from the realm of biology (Bogatyrev, 2000; Bogatyreva, Pahl, Bogatyrev, Vincent, 2004; Vincent, 2002).

Initially it is necessary to formulate *the basic result* (or Ideal Final Result according to TRIZ), which will ultimately satisfy all our requirements and try to solve the appeared problem(-s) …without any special devices, machines, technologies, etc. only by different arrangement and mutual positioning of the existing objects, items and processes in space and/or in time. If that does not help, one should formulate *the most essential function*, which we require to obtain the desired result. Then step-by-step we must adjust this function to the surrounding media, more distant environment, to the size scale, to the scale of operating (time and space limits), ergonomical, economical, ecological, cultural demands. And at every step we must compare the existing technological solutions with the way living systems solve the problems of that kind. Often the most optimal solution is some sort of amalgamation of the both – technological and biological strategies. And this is the promising sign for the future of biomimetics.

Table 1 Comparison of natural living and engineering non-living systems.

Living natural systems **A**	**Non-living technical artificial systems** **B**
1. Operate in relatively narrow conditions of temperature, pressure, chemical environment, etc. Utilisation of high-energy electromagnetic fields, radiation is absent.	1. Operate within sufficiently wide conditions, which are beyond the limits of living creatures' tolerance. Utilisation of high-energy electromagnetic fields, laser, radiation, extreme temperatures, and pressure is widespread.
2. Balanced complex living systems tend to create homeostasis or homeorhesis due to closed cycles of energy and substance.	2. Most of human technologies are open-ended, which causes most of problems in various types of misbalance and lack of sustainability.
3. Relatively slow rates of evolution.	3. Very fast and accelerating development of technologies.
4. Long range of sustainability.	4. Short range of effectiveness ("here and now by any price")
5. Slow processes are widespread.	5. Slow processes are considered as a shortcoming.
6. Complex ecological systems tend to drift from *r*-to *K*-mode.	6. Economical forces make steady shift from *K*-to *r*-mode in manufacturing and life cycle of any product.
7. Biological systems mostly avoid long-range transportation.	7. Contemporary industrial systems are unthinkable without massive global transport flows.
8. Living system as a rule participates (in all the processes concerning it) as a central agent.	8. Technology evolution goes from mechanisation via automatisation towards replacement of human-operator in the manufacturing process
9. Morphological trends in evolution: oligomerisation of effectors and metamerical parts of body.	9. Morphological trends throughout the evolution: mono-, bi-, poly-systems (i.e. polymerisation of monomerical parts).
10. Replication, reproducing, cloning, metamerisation.	10. Fragmentation.
11. The newly evolved systems do not substitute the old ones, often show the parallel existence.	11. A new technology substitutes the old one to maximum extent.
12. Common types of locomotion and manipulation: oscillation, reciprocation, impulse.	12. Typical types of physical movements for locomotion and manipulation: rotation.

Acknowledgements

I would like to thank my colleagues from the Centre of Biomimetic and Natural Technologies (University of Bath, UK) who always gave me inspiration and were attentive to my ideas.

References

Altshuller G. S. Algorithm of invention. Moscow, "Moscow worker", 1973. 296 p.

Bogatyrev N.R. Ecological engineering of survival. – Publishing house of SB RAS, Novosibirsk, 2000, 184 pp.

Bogatyrev N.R. A "living" machine (2004). Journal of Bionic Engineering, v.1, No.2, p.79-87.

Bogatyreva O.A., Pahl A.-K., Bogatyrev N.R., Vincent J.F.V. Means, advantages and limits of merging biology with technology. Jour. of Bionic Engineering, 2004, vol.1, N.2, p.121-132.

Vincent J.F.V. Smart biomimetic TRIZ. – TRIZ future, Pro ETRIA World Conference, Strasbourg, 2002, p. 61-68.

The Anthropomimetic Principle

Owen Holland and Rob Knight
Department of Computer Science
University of Essex
Wivenhoe Park
CO4 3SQ
owen@essex.ac.uk, rrk@essex.ac.uk

Abstract

Most humanoid robots are essentially conventional robots that fit within the morphological envelope of a human. However, for robots that are intended to help in the understanding of human cognition and action, a much higher level of biological inspiration may be necessary. This paper sets out some of the requirements for an anthropomimetic robot – one which imitates not just the human form, but also the biological structures and functions that enable and constrain perception and action – and describes the design, construction, and initial performance of such a robot. The findings to date indicate that the combination of a realistic skeleton, series-elastic actuators, and a foveated vision system gives a unique insight into the problems the human brain has to solve in the areas of perception and action.

1 Introduction

At present, there are approximately 70 major humanoid robot projects being undertaken around the world [1]. It might be expected that such a high level of activity would generate a constant stream of new findings relevant to the field of biologically inspired robotics, but unfortunately this does not seem to be the case. Perhaps the main reason for this is that the aim of the typical humanoid development programme is simply to engineer a mobile robot that fits within a broadly human envelope, and employs a broadly human range of movements; there is no intrinsic requirement to draw inspiration from the ways in which humans actually produce and control their actions.

This is not intended as a criticism of these projects, because there are many different reasons for adopting a predominantly morphological perspective. For example, NASA [2] is interested in robots that can undertake maintenance and repair tasks on spacecraft designed to be worked on by humans, and so a human morphology is the logical choice. Many Japanese humanoid robots, such as Honda's Asimo [3], are intended to assist humans in normal domestic or work tasks, and a human-like morphology facilitates human-robot interaction and cooperation, as well as being matched to the human-centred environment. Some entertainment robots, such as the Sony QRio [4], are designed to charm by imitating humans. All of these robots are actuated and controlled using conventional engineering techniques of an extremely high standard, and all meet the specified requirements, but none of them draw on any further biological inspiration.

Of course, there are humanoid programmes that do go beyond mere morphological similarity. ATR's DB (Dynamic Brain) project [5] uses a hydraulically powered humanoid as a test-bed for movement control algorithms inspired by the neural structures and processes thought to be used by the brain [6, 7]. A more recent theme has been the development of several walking robots that use explicitly biologically inspired techniques to produce extremely efficient and natural-looking bipedal locomotion [8]. Such robots apply the ideas of passive dynamic control – the exploitation of resonances, oscillations, foot shape, and passive compliance. These and other examples, taken together, point us in a direction that may enable both the design of better robots, and an increased understanding of human movement control: *why not build a humanoid robot that faithfully copies the essential physical structure of a human, and attempt to control it using the same methods as the brain does?*

We propose to call such machines *anthropomimetic* robots. We believe that this shift of emphasis from the outward form to the nature of the internal mechanisms carries the promise of transforming not only humanoid robotics, but also the way in which robotics is perceived by the lay community. However good the cosmetic appearance and however soft the flesh-toned latex of the latest Japanese robotic receptionist, people are always aware of the artifice, knowing that the apparent humanity is only skin deep. But if the robot's internal form and function is also close to our own, the question of the boundary between the natural and the artificial will become more acute, and the debate about the nature of the relationship between robots and ourselves may take on a new urgency.

This paper describes the progress of an attempt to build a truly anthropomimetic robot. The next section sets out the background to the project; section 3 describes the technologies used in building the robot; sections 4 and 5 describe the construction and behaviour of the first two prototypes, and section 6 discusses the main findings so far.

2 Approaching anthropomimesis

The robot described in this paper is a spin-off from a larger project aimed at building a robot with at least the potential for some form of consciousness. Several writers on the subject (e.g. [9], [10]) have identified the existence of an integrated internal model of the self as being a key component of consciousness. Such models are thought to include many aspects of the body, including how it is controlled. The central idea behind the parent project was to build a robot that could develop such a self-model, and this led to the question: what sort of body would the robot need to form the kind of internal model that might support consciousness? For want of any better information, the obvious answer is: a body similar to that of a human, since human consciousness is the only consciousness about which we have any reliable information. However, when we reviewed the state of the art in humanoid robotics, it rapidly became clear that almost all existing humanoids had bodies that were only superficially similar to humans, and that they were moved and controlled in ways very different from those of humans. It was necessary for us to start from scratch.

3 Materials and components

The technical problems we faced in the construction of the robot centred around two key problems: the skeleton; and the musculature. Of course, these could not be treated independently, because each imposed constraints on the other. In the event, however, the first strategy we tried worked very well: it was simply to copy the skeleton as best we could – at life size – using purely passive elastic elements to represent the musculature, and then to investigate possible ways of constructing and installing suitable powered muscle analogues.

3.1 The Skeleton

The adult human skeleton is at first sight extremely complex, containing 206 bones. (Interestingly, we have 275 bones at birth, but many have fused by maturity.) However, since more than half are in the hands and feet, and since our bilateral symmetry means that most bones have a mirror image bone with identical structure and function, the problem of building a working skeleton may just be very difficult rather than completely intractable.

Our first problem followed directly on our decision to model the bones of the skeleton: how could we model bone-like structures? In a conventionally engineered robot, the actuators are built into the joints, and the only constraints on the links between the joints are those of rigidity, clearance, and weight. However, as is clear from any anatomy textbook [11, 12], or more spectacularly from the plastination preparations of Gunther von Hagens [13], bones must also provide the points of attachment for the tendons, and this can be critical in determining how the mechanical advantage of a muscle-tendon-joint system changes as the joint moves. (Indeed, in many cases the bone will form also surface over which the tendon moves.) In addition, the joints are not limited to simple hinges or universal joints, but may accommodate rolling or sliding movements. To machine, fabricate, or cast a large number of different such components by conventional methods would be difficult and expensive.

The solution was to use a new type of engineering thermoplastic known in the UK as Polymorph, and in the US as Friendly Plastic [14]. Technically a caprolactone polymer, it is polythene-like in many ways, but when heated to only 60 degrees C (for example, by plunging into hot water) it fuses (or softens, if already fused) and can be freely hand moulded for quite some time, finally resetting at around 30 degrees. It has a distinctly bone-like appearance when cold. Since it is a true thermoplastic, it can be reheated and remoulded as many times as is necessary; it is possible to soften it locally, which makes it particularly easy to use. It is readily moulded around other components and materials – for example, it can be used to form a ball and socket joint by moulding it around a metal sphere mounted on a rod. Its slight contraction on cooling can be used to ensure tight joints when it is moulded around other components.

In practical engineering terms, it is tough and springy. Its tensile strength is good – Polymorph has the highest tensile strength of all the capralactones, at 580 kg/cm^2. It can be further strengthened (and stiffened, if necessary) by adding other materials, such as wire, or metal rods or bars.

3.2 Muscles and actuators

Although there are many different types of muscle in biology, the 650 or so human skeletal muscles are fairly stereotyped. A muscle consists of a number of muscle fibres (or cells) arranged in parallel, and connected at each end to a common tendon, the elastic connection to the skeleton. When a muscle fibre is stimulated by its associated motoneuron, it fires and contracts momentarily, exerting force on the tendon. A given motoneuron innervates only a single muscle, but controls a number of muscle fibres within that muscle, typically between ten and a hundred; the combination is known as a motor unit. A given muscle is innervated by a number of motoneurons, in many cases by hundreds of them. A sustained muscular contraction is achieved by repeatedly stimulating individual motor units, and the strength of the contraction is modulated by varying the number of motor units activated. Muscle is elastic tissue, and the force exerted is a function not only of the motor unit activation but also of the length of the muscle, which of course changes if the associated joints change position as a result of the balance between the load and the effort.

In many animals there are reflex and auxiliary subsystems in place to enable more sophisticated closed loop control of muscular systems. The level of complexity varies, but mammals are at the top of the tree, with sensor systems for measuring muscle tension (via the Golgi tendon organ), and the effective length of a muscle (via the muscle spindle). These are involved in various feedback systems, and the sensitivity or gain of some of these (e.g. the stretch reflex) can be centrally controlled.

The essential nature of many skeletal muscle systems derives from two factors: muscles (and tendons) are elastic, and so can only pull and not push; and most degrees of freedom are controlled by antagonistic arrangements of muscles, where the effect of one muscle is opposed by that of one (or more) others. This has two consequences. First, if a muscle and its antagonist are stimulated together, the affected joint will move, changing the lengths of the muscles, until their effects balance the imposed load. This position is known as the set point. Second, the resistance offered to an externally imposed disturbance at the set point – the impedance – is primarily a function of the tension and elastic properties of the muscles involved. Both of these factors make skeletal muscle systems very different from conventional robotic actuation arrangements; as will be seen, these differences have far-reaching effects.

The typical robotic actuator is very different from muscle. In most applications, precise control of trajectory and/or position is of paramount importance, and robotic actuators tend to be extremely stiff to enable this precision. This has two main consequences. First, any unplanned impact with environmental obstacles can impose a shock loading on the transmission (typically a gear train) that may lead to failure or degradation. Second, an unplanned impact with a human can represent a serious safety hazard. The standard ways of dealing with these problems (strengthened transmissions, safety cages) are unsuitable for mobile humanoid robots. As it happens, one technology for dealing with these problems can be adapted to mimic many of the desirable properties of muscle.

3.2.1 Series-elastic actuators

The solution of interest is to use a conventional high impedance actuator, but to place it in series with a source of compliance. At its simplest, this can merely be a rubber buffer to protect gear teeth from the peak shock. However, in many applications, there is also a requirement for good force control, and this offers a further challenge. An excellent compromise is offered by the series elastic actuators first developed at MIT [15], and later commercialised [16]. In these, the source of compliance is a spring; by simply measuring the extension or compression of the spring, the force can be accurately known, and any deviation from the required force can be compensated by using a high gain position controller for the conventional high impedance actuator. Unfortunately, it seems to be impracticable to use these high-specification actuators on a life-size humanoid, for reasons of expense, size, power, and weight.

What other options are available? One possible approach would be to use commercial pneumatic actuators. These can be relatively small and light, although their characteristics are in many ways undesirable, as instead of exploiting their intrinsic compliance, they are usually engineered to reduce it to obtain reasonable position control. An art project [17] has investigated the use of such actuators for an anthropomimetic device in a very different context, but with some success. Like us, they aimed to build a life-size humanoid with a fully articulated skeleton, but their concern was driven solely by artistic considerations, in that they wanted to reproduce human movement patterns rather than to produce an effective mobile robot. Using movement scripts derived from humans, they demonstrated what appear to be very fluid and smoothly coordinated movements, but did not undertake any functional analysis of the source of the observed characteristics.

Our eventual decision was to investigate the use of the series elastic technique, but to use a much lower level of engineering sophistication. Our approach was driven by considerations of cost, size, weight, performance, and power. The cost and size had to be as small as possible - the torso alone would require at least forty powered degrees of freedom. Maximum performance was critical - some of the actuators would be required to generate forces of the order of 1000N. In order to avoid problems with the distribution of power, an actuator with a built-in power source was highly desirable.

Our solution took advantage of the mass production of a common domestic device – the electric screwdriver. These are designed to produce torques of around 3Nm in from NiCad battery packs of 6V nominal voltage, and the direction of rotation is electrically switchable; backdriving against the set direction is prevented by a sprag clutch. The elastic element is provided by marine grade shock cord – a sleeved natural rubber core available in a number of thicknesses. For light loadings we use a 5mm type, and for heavier duty a 10mm version. The shock cord is terminated at each end by 3mm thick braided Dyneema kiteline with a working breaking strain of 250kg. This material, also known as Spectra, is a heavy molecular weight polyethylene (HMWPE), 40% stronger than Kevlar, and with negligible stretch. By winding the kiteline round a 10mm spindle driven by a standard good quality screwdriver motor and gearbox, we can achieve tensions in excess of 520N; by overdriving a rather better motor, we can increase this to around 860N. The maximum current draw is of the order of 20A, giving reasonable endurance from the custom 7Ah battery packs.

Many of the techniques developed can be seen in Figure 1, an early investigation of an unpowered knee joint. The upper half of the joint contains two large chrome balls embedded in Polymorph – one is clearly visible on the left. When the knee is straightened, the two balls rest in the moulded cups in the lower part of the joint, partly locking the joint to give some stability. As the joint rotates during flexion, the load is partially transferred to the moulded ball housing, and the joint becomes a rolling joint. The kneecap, or patella, is continuous with the lower part of the joint, but is thin enough to bend under the load of the tendon from the extensor motor. The termination of the flexor tendon can be seen on the left. The joint is held together by the tension transmitted through the tendon connecting the lower leg to the upper joint housing, and passing through a hardened insert moulded into the lower leg. It is clear that there are enormous differences between this type of joint and the conventional geared hinge joint forming most robot knees, and it is overwhelmingly likely that the functional characteristics of the joints will also be different. It may well be that the anthropomimetic knee is inferior to the engineered knee in many respects, but the point is that the anthropomimetic knee is a better representation of the control problem that the brain has to solve.

Figure 1: An early knee design using Polymorph

4 The first prototypes

Before construction began on the final prototype, we carried out a number of design studies and partial prototypes. Figure 2 shows the first prototype of the head and neck. It is clear that the neck is made up of several vertebrae – four in all – but they are much longer than their human equivalents. There are two reasons for this. First, it would be impossible to accommodate all the motors and tendons in a strictly faithful copy, and so reducing the number of vertebrae at least maintains some degree of qualitative fidelity. Second, it is not yet possible to build a hand able to expose held objects to different points of view by manipulating them, and a possible remedy for this is to provide the head with rather more than the usual range of movement to enable a more through visual inspection. This was done by increasing the length of the vertebra to enable the head and neck to crane and rotate in an exaggerated manner. In this version, the neck has two motors fitted – they are clearly visible just below the single eyeball. Each motor and gearbox is moulded into a Polymorph housing which provides the fixing points, and also supports the spindle onto which the kiteline is wound.

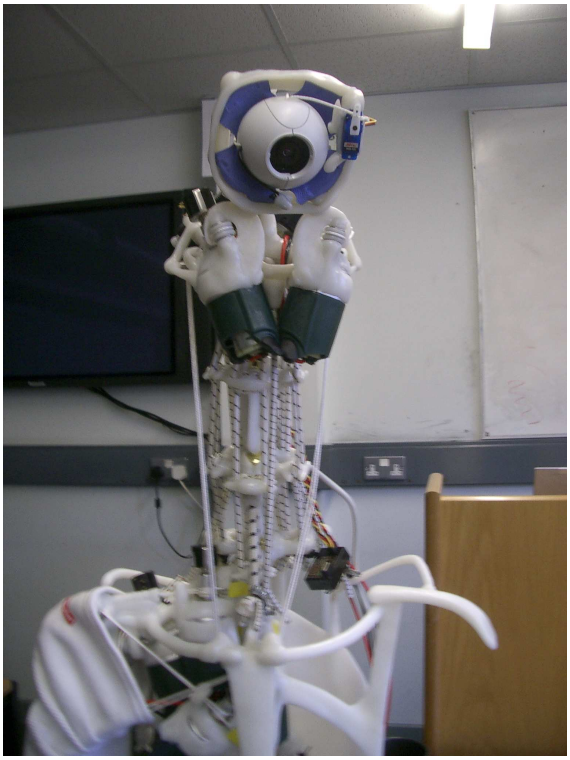


Figure 2: The prototype head and neck.

Figure 3 shows the first prototype of the torso and arm. The structure of the spine, which was purely passive at this stage, shows six vertebrae with rather exaggerated lateral extensions. Each vertebral joint was formed around a chromed sphere cast into one vertebra, and free to rotate within a matching cup in the other vertebra. The shoulder joint was quite a faithful rendering of the real thing – in fact, it could be dislocated in exactly the same way as a human shoulder – but did not include the shoulder blade. (The function of the shoulder blade is quite complex – see [ ] for what we believe is the only example of a robot with a correctly functioning but rather abstract shoulder blade).

These initial subsystem prototypes were extremely useful in developing the appropriate modelling technology. However, the first indication of the nature of an anthropomimetic robot came from the first system prototype, CRONOS, which combined a torso, arm, and head. As can be seen from Figure 4, the sheer profusion of powered and unpowered tendons gives a strong qualitative impression of a biological system. This impression soon gives way to the realisation that the robot represents something qualitatively distinct from a conventional robot, even before it is powered up. You take his hand and shake it: it moves easily, *and so does his whole skeleton*. This multi-degree-of-freedom structure, supported by the tensions between dozens of elastic elements, responds as a whole, transmitting force and movement well beyond the point of contact. You take his arm and push it downward: the elbow flexes, the complex shoulder moves, and the spine bends and twists.

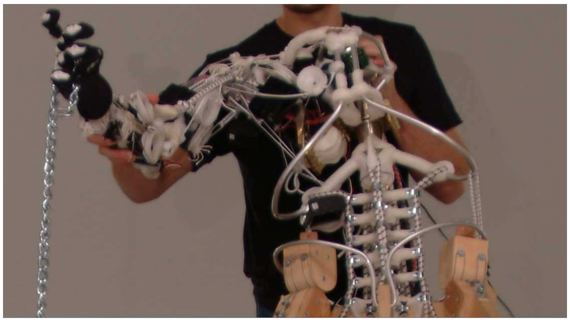


Figure 3: The prototype torso and arm

Figure 4: The first operational prototype, CRONOS.

When the robot is powered up, it moves to some equilibrium posture, but the character of the movement is again highly distinctive, because the disturbances due to the robot's own movement are propagated through the structure just like the externally imposed loads. Of course, if all that is wanted is a robot that fits into a human envelope, is able to operate in limited ways on a largely static and predictable world, and is tractable from the point of view of control, this flexibility is nothing but a nuisance. But if the target is a robot that as far as possible works in the same way as a human – an anthropomimetic robot – then we must face up to the problems that robots like CRONOS present.

These problems are not simply to do with the difficulty of controlling such a redundant and flexible structure. The intention is that, like a human, the robot will be predominantly visual, and so it has been equipped with a visual system that is also anthropomimetic (Figure 5). However, it differs from humans in having a single central eye; this simplifies visual processing enormously, and can be justified by the fact that around 20% of humans do not perform stereo fusion, yet their performance on visual tasks is within the normal range. The imaging unit (currently a 640 x 480 colour webcam with a 25 degree field of view, shortly to be replaced by a specialist high resolution camera with a 90 degree field of view) is mounted in a model eyeball, and is moved by functional analogues of the six extraocular muscles, able to control rotation as well as pan and tilt.

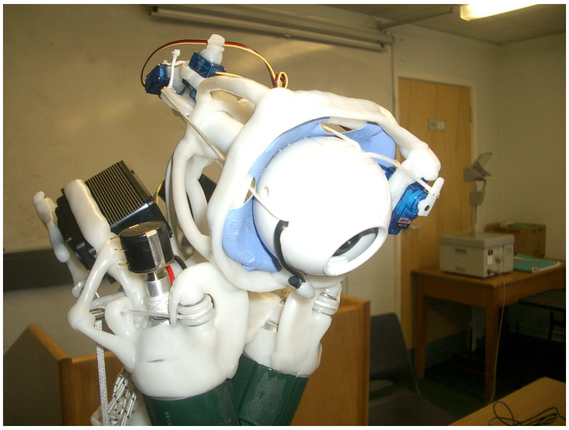


Figure 5: The prototype vision system

In order to reflect the nature of the human visual system, the first stage of image processing involves the application of a transform to mimic the reduction in density of photoreceptors between the fovea and the periphery. A foveal system places a heavy emphasis on the accurate and precise control of gaze direction, and the disadvantage of an anthropomimetic robot is immediately apparent: it provides a much less stable platform for the visual system than does a conventional robot. In the absence of any dedicated means of stabilisation, even the slightest change in external or inertial load is reflected through the whole skeleton, usually causing considerable movement and vibration at the extremity of the body, where of course the visual system is mounted. In a conventional robot, with a conventional unfoveated vision system, these problems would not arise; however, their presence in the anthropomimetic robot emphasises that they must have been solved by the brain, and so we will be pushed to solve them, perhaps even in the same way. (We have ordered an inertial measurement unit to act as a source of input analogous to that from the vestibular system.)

5 The second prototype

Building on the knowledge gained from the early prototypes, a second prototype has now been constructed and is undergoing further development and testing. It is still limited to a torso, head, and arms (the second arm has not yet been attached in the figure). The original arm design has been modified. The first shoulder design seemed unnecessarily complicated, and so several engineering simplifications were made. The new arm is superior in almost all respects, but its range of movement is now rather limited in the vertical direction, and it is in the course of being redesigned to include some features of the original. The hand has also been redesigned, mainly because the prototype hand occasionally broke under load. In the original hand, each finger was formed by compressing softened Polymorph with an edge at suitable intervals to define each joint and to form a hinge. These hinges could not always withstand the forces imposed on them, and so in the new hand, two lengths of kiteline are moulded into the finger before the joints are formed, ensuring that each hinge is reinforced with two strands of Dyneema. There have been no failures since. Once again, this illustrates the suitability of Polymorph for this type of construction.

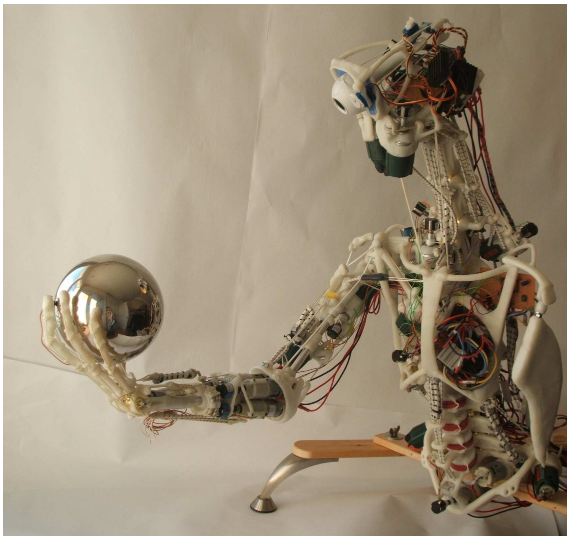


Figure 6: The head and torso with one arm fitted

Table 1 lists the powered degrees of freedom currently available, and relates them to the skeletal musculature where appropriate. Note that some degrees of freedom, such as those dealing with eye

No.	Section	Description	DOF	Actuator	Muscular equivalent
1	Eye	Eyeball orientation	Pan	Servo	Lateral/medial rectus
2	"	"	Tilt	Servo	Superior/inferior rectus
3	"	"	Rotation	Servo	Superior/inferior oblique (partial)
4	Head and neck	Head pitch and rotation	Head pitch and left rotation	Servo	Simplification of many muscles
5	"	"	Head pitch and right rotation	Servo	Simplification of many muscles
6	"	Neck can crane forwards and sideways with passive return to upright	Neck forwards, head rotate left	Motor	Sternocleidomastoideus
7	"	"	Neck forwards, head rotate right	Motor	Sternocleidomastoideus
8	"	"	Neck left	Motor	Simplification of many muscles
9	"	"	Neck right	Motor	"
10	Shoulder	Arm can raise and rotate	Arm raise sideways	Motor	Lateral Deltoid
11	"	"	Arm raise forwards	Motor	Anterior Deltoid
12	"	"	Arm adduction and internal rotation	Motor	Pectoralis Major
13	"	"	External rotation	Motor	Infraspinatus
14	"	"	Raise arm and external rotation	Motor	Teres Minor
15	"	"	Retract arm and internal rotation	Motor	Teres Major
16	"	"	Flex arm and raise forward	Motor	Biceps Brachii
17	Elbow	Controlled by (16), (17) and (18)	Extend arm	Motor	Triceps
18	"	"	Flex arm	Motor	Brachialis
19	Wrist	Pitch and yaw	Inwards and upwards	Motor	Simplification of many muscles
20	"	"	Outwards and upwards	Motor	"
21	"	Pitch	Downwards	Motor	"
22	"	Roll	Rotation only	Motor	"
23	Hand	Grip with passive release	Grip	Motor	"
24	Waist	Support from spine limits motion in all directions	Back and left	Motor	"
25	"	"	Back and right	Motor	Simplification of many muscles
26	"	"	Forwards	Motor	Mainly equivalent to Rectus Abdominis
27	"	"	Rotate left	Motor	Simplification of many muscles
28	"	"	Rotate right	Motor	Simplification of many muscles

Table 1: Powered degrees of freedom with one arm fitted.

movement, are implemented using conventional servos because the load is light and constant.

The behaviour of the second prototype has been studied in several contexts, and has proved very illuminating. Simply moving one degree of freedom, even jerkily, produces what looks like a fluid and coordinated whole-body movement which all observers to date have agreed is very natural and 'biological looking'. This is because the static and dynamic loads produced by the movement are transmitted through the skeleton and the elastic linkages, producing what we have called 'passive coordination'. Repeating the same movement under (open loop control) with a load (such as the round weight shown in Figure 6) produces an equally natural movement, but one in which the weight and inertial forces produce a rather different trajectory and finishing point. This emphasises that the command for such a movement, to be successful, must take account of the anticipated loadings; we believe this is unlikely to be successful if done purely reactively, and that feedforward compensation – anticipating and predictively cancelling the effects of the load – will be necessary for almost any movement, a somewhat daunting prospect when designing the controller.

6 Conclusions

Although we are still dealing with a prototype, we can already see that the anthropomimetic approach is distinct from the standard humanoid approach, and that it is much closer to the type of biological inspiration discussed in this symposium. The observations to date indicate that the combination of a realistic skeleton, series-elastic actuators, and a foveated vision system gives a unique insight into some of the problems the human brain has to solve in the areas of perception and action. We are no closer to solving any of these problems than we were at the start of the project, but at least we have some confidence that we are moving in the right direction.

Acknowledgements

This work was funded by the Engineering and Physical Sciences Research Council (GR/S47946/01)

References

[1] http://www.androidworld.com/

[2] http://robonaut.jsc.nasa.gov/

[3] http://world.honda.com/ASIMO/

[4] http://www.sony.net/SonyInfo/QRIO/

[5] Atkeson CG, Hale J, Pollick F, Riley M, Kotosaka S, Schaal S, Shibata T, Tevatia G, Vijayakumar S, Ude A, Kawato M: Using humanoid robots to study human behavior. *IEEE Intelligent Systems: Special Issue on Humanoid Robotics,* **15,** 46-56 (2000).

[6] Kawato M: Internal models for motor control and trajectory planning. *Current Opinion in Neurobiology,* **9,** 718-727 (1999)

[7] Wolpert D, Kawato M: Multiple paired forward and inverse models for motor control. *Neural Networks* **11,** 1317-1329 (1998)

[8] Collins, S.H., Ruina, A.L., Tedrake, R., Wisse, M. (2005) Efficient bipedal robots based on passive-dynamic Walkers, Science, 307: 1082-1085.

[9] Ramachandran, V.S. and S. Blakeslee Phantoms in the Brain: Probing the Mysteries of the Human Mind. New York: William Morrow, 1998.

[10] Being No One: The Self-Model Theory of Subjectivity. Thomas Metzinger, MIT Press, (2003)

[11] http://www.bartleby.com/107/

[12] Gray's Anatomy, 39th edition: The Anatomical Basis of Clinical Practice. Susan Standring, Churchill Livingstone (2004)

[13] http://www.bodyworlds.com

[14] http://www.mutr.co.uk/pdf_files/LIT0048.pdf

[15] United States Patent 5,650,704, Pratt et al., July 22, 1997

[16] http://yobotics.com/actuators/description/description.htm

[17] http://www.amorphicrobotworks.org/index.htm

A Metaphor for Swarm Trophallaxis Modelled on the Vampire Bat *Desmodus rotundus*

Mark Witkowski
Department of Computing
Imperial College London
180 Queen's Gate
London SW7 2AZ
United Kingdom
`m.witkowski@imperial.ac.uk`

Abstract

This paper reports on some observations arising from the use of a simulator in the context of energy sharing (trophallaxis) in swarms or communities of energy critical robots. It is presumed that such robots would gather naturally occurring fuel materials and convert them to energy as required. Using food-sharing behaviour reported for the common vampire bat species *Desmodus rotundus*, it is argued that energy sharing can bring substantial benefits to the group as a whole in situations where suitable materials are encountered unevenly by members of the population, in a manner insufficient to guarantee the continuing survival of any individual. Using the simulator the survivability of groups is investigated under various regimes, including sharing, the effects of energy dependents and non-sharers. The role of reciprocation is considered.

1 Introduction

This paper describes a simulation of mutual food sharing. Here the focus is on one particular type of food sharing strategy in which the transfer of food has been shown to take place between adult vampire bats *Desmodus rotundus* as an exemplar of energy management strategies that might be employed by energy critical robotic swarms in the future. The simulation parameters are primarily based on available data from the common vampire bat, but may be readily adapted to other scenarios to determine the stability of resource sharing under different conditions or where different sharing strategies are to be considered.

Wilkinson (1984, 1990) has determined, both by observation and by experiment that common vampire bats appear to form "self-help" food-sharing groups of approximately 20 animals, adults and infants, who regularly associate and roost together from larger colonies of several hundred bats. Specifically, the common vampire bat has been observed regurgitating blood meals, not only to their young, but also between adults; in the latter case on an apparently reciprocal basis (Wilkinson, 1984). He argues that this brings substantial benefits both to the individuals and group as a whole, and that claim is investigated here.

This is an example of *trophallaxis*, the transfer of food or energy between individuals, usually by regurgitation. Such behaviours have been widely observed in insect (several ant, termite, wasp and bee species), avian and mammalian (dog) species, either as a mechanism for specialisation within the group, or for the feeding of young. Other forms of apparently "altruistic", cooperative or reciprocal behaviour have been identified in nature. See, for instance, Clutton-Brock (2002) for a recent review.

The simulation is part based on previous work relating to the development of reciprocal "trust" in groups of trading software agents (Witkowski *et al*. 2000). These objective-Trust Based Agents (o-TBA) conduct a multi-player iterative trading game in which the current choice of trading partner from the group is determined on the basis of prior reciprocity, favouring those that have proved trustworthy in the past. As with the Iterated Prisoner's Dilemma (IPD), (Axelrod, 1984) each potential partner is forced by circumstances to select whether to support or defect on potential partners. Unlike the IPD, these agents are in a society of others, with whom they must choose to associate.

In general, tit-for-tat strategies between individuals, such as the IPD, have been extensively studied, both from both game theoretic and evolutionary viewpoints (e.g. Axelrod, 1984; Dawkins, 1989; Trivers, 1985). That these bats have developed an effectively "evolutionary stable strategy" (ESS) is hardly in doubt. This paper, therefore, seeks to identify the principal behavioural properties that comprise that strategy and investigate some of its consequences.

Direct energy transfer in this form also allows us to consider different mechanisms of energy sharing between individuals in a cooperating group or swarm of similar robotic devices. Potential applications for these energy-sharing methods include swarms of autonomous of robots sent to explore remote areas, for land based mapping or surveying, where conventional sources of electrical power are not otherwise available. Part of the behaviour cycle being given over to cooperative tasks, part to the individual search and acquisition of suitable energy substrates from the general environment. This paper only considers sharing behaviours. There is, of course, no suggestion whatsoever that any consideration should be given to designing robots that also share the bat's generally undesirable and parasitic sanguinivorous preferences.

Energy autonomy in robotics has become an increasingly important issue in recent years, mirroring interest in other forms of behavioural autonomy. In particular, developments in "alternative" methods for energy generation have been proposed (e.g. Wilkinson, 2000; Ieropoulos *et al.*, 2003, 2005), in which a robot's energy requirements are completely or partially met through energy extraction from normally available "foodstuffs" by microbial digestion. When developed, these technologies will allow autonomous robotic devices to perform tasks in a range of inaccessible locations, dependent primarily on the availability of suitable energy substrates.

The present generation of such energy cells are characterised by relatively slow energy production and small yields, and this has been mirrored in the development of strategies for strict power management in order to best utilise these characteristics (e.g. McFarland and Spier, 1997). This paper anticipates a time when bio-fuelled cells are efficient, but still dependent on a sufficient and regular supply of energy producing substrates. Under the assumptions that such robots will require regular intakes of the substrate, and that these materials may be either often, or periodically, be in short supply. Such energy cells may not be "re-startable" once depleted (Ieropoulos *et al.*, 2003) so maintaining a regular supply of energy substrate is a significant factor. Kubo and Melhuish (2004), for example, have described a simulated task in which multiple robots co-operating on a task may pass energy between them to extend the effective lifetime of the group.

Section two considers some pertinent evidence for food sharing and energy management in the vampire bat. This will be used to populate the simulation engine variables, described in section three. Section four presents results of the simulation under a variety of conditions. Section five considers the role of reciprocation in the model. Section six presents some discussion.

2 Vampire Bat Energy Model

The common vampire bat feeds only once a day, at the darkest part of the night. They do so by "sneaking up" on their victims (for preference, large farm animals, such as cattle, sheep and pigs), making a small cut with their modified incisor teeth and lapping (not sucking) up the blood that emerges. Their saliva contains a powerful anti-coagulant to prevent clotting. An adult will feed in this way for 20-30 minutes and ingest approximately 25-30ml of blood, a significant proportion of its bodyweight. Bats typically feed to satiation before returning to their daytime roosts.

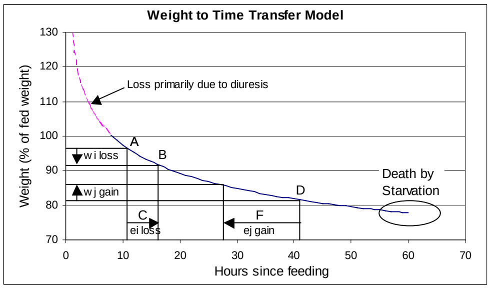


Figure 1: Wilkinson's Weight to Time Model

Wilkinson (1984) developed a predictive model for the nutritional state of an individual bat, summarised in figure 1. He noted that the weight of a bat was adequately modelled as a negative exponential (weight = $130.25 * \text{time}^{-0.126}$: for $8 \leq \text{time} \leq 60$) relative to its fed weight (100%), an allowance being made for the initial diuretic phase. He also observed that a bat would no longer be able to maintain its body temperature once its body weight fell to 75% of the initial weight, and would starve. This point occurs some 60 hours following a full feed. The bodyweight curve therefore acts as a predictor of the unfed time remaining. This also means that a fully fed bat has only two further foraging flights before it becomes seriously compromised.

Wilkinson further reported that the feeding success rate in adults (>2yrs) in the colony he studied averaged 93%, but that juveniles (<2yrs) only averaged a 67% success rate. Apparently juveniles must learn a deft touch with their prey.

Given that adult vampire bats have no effective mechanism for storing energy in their body tissues, Wilkinson (1990) later estimated that with no further mechanism to ameliorate this situation, the average annual adult mortality rate for adults should be 82%, whereas the measured rate is only 24% per annum.

2.1 Food sharing in vampire bats

Wilkinson conducted two seminal studies of vampire bat feeding and food transfer, one by observation of vampire bats in their native habitats, and one a "laboratory" style investigation.

In the first investigation Wilkinson (1984) and his team observed 110 cases of regurgitation during 400 hours of (daylight) observation (mean duration = 63.0 seconds, s.d. = 57.6s, longest observed = 390s). Of these, 77 were between mother and nursing offspring. The remaining interactions were analysed for known prior association and relatedness between individuals, indicating that prior association, rather than relatedness is the key indicator of sharing.

In the second study bats from two separate (50km apart) sites were captured and placed in a cage. The first group (from La Pacifica) comprised four adult females from one roost tree. The second (Santa Rosa) group comprised three adult females and one infant (related), and one adult male, known not to have associated with the females.

On successive nights one bat was removed from the cage before the remaining bats were allowed access to blood for two hours. The denied bat was then returned to the cage after a delay of between two and 12 hours, and all regurgitations in the following two hours recorded. All bats were deprived in this manner twice during the experiment. A total of 13 regurgitations were noted (mean = 25s, s.d. = 25s), 12 between bats from the same population (i.e. with prior association), but once between groups without prior association. Only the low-association male was not aided.

All recipients were within 24 hours of their estimated minimum viable weight (mean = 13.2h, s.d. = 4.7h), and all donors had in excess of 24 hours estimated time until starvation (mean = 39.9h, s.d. = 15.5h). Wilkinson also reports that of the donations made, "reciprocal" donations were made at a rate significantly more often than expected by chance (several potential donations remained at the conclusion of the experiment).

Wilkinson points out that using the figure 1 model, a donor bat transferring the quantity A-B (figure 1) loses C hours until starvation, whereas the recipient bat of like weight would gain F hours benefit if at point D and F > C. Wilkinson argues that if A > D, the recipient apparently gains more than the donor loses by the transfer.

Vampire bats are homoeothermic (McNab, 1973), maintaining their body temperature, and use more energy in ambient temperatures lower (or higher) than their body temperature. It is unclear therefore, whether this apparent effect is due to lowering of overall metabolic rate with time, changes in meal composition with digestion, or is only useful as an indicator. Wilkinson regards this apparent gain as essential factor in reciprocal feeding. This assertion is challenged in section 4.4.

Wilkinson's data indicates that each regurgitation supplies only a small amount of blood, sufficient for only a few hours of extra life (as a comparison, he estimated that a 400s regurgitation added the equivalent of 12 hours life to the recipient). From this data it may be inferred that an individual close to starvation will require multiple donations if it is to survive until the next feeding opportunity.

3 The Simulation

This section is given over to a discussion of a simulation of the mechanisms of trophallaxis. In the absence of appropriate robot data, the available data from the behaviour of the common vampire bat will be used to inform the parameters of the simulation. The data provided by Wilkinson's (1984, 1990) studies leaves several questions only partially answered, and the simulation provides some insights into the mechanism that has been adopted, and makes some predictions that might be resolved or verified with further study of the bat in its natural habitat.

3.1 Group structure

The simulation assumes a single tight-knit group of individuals, split between sharers (R_{share} = percent sharers), non-sharers ($R_{non\text{-}share}$) and dependents ($R_{dependent}$ = % ratio dependents to non-dependents). At present the simulation emulates the energy dynamic amongst a fixed group (default *groupsize* = 20, 16 sharers and four dependents). Groupsize, the proportion (%) of sharers (i.e. female adults) to non-sharers (male/female sharing is highly asymmetric) and the proportion (%) of non-dependent to dependents (i.e. infants) can be set as required. These defaults reflect the supposed composition of a vampire bat group. Future versions of the simulator will allow for more detailed investigation of the group dynamic occasioned by births, deaths, arrivals and departures.

3.2 Basic simulation cycle

Each cycle of the simulation represents one hour of actual elapsed time, energy consumption and gains (through feeding or transfer) are calculated at each cycle. Ages are expressed in days. The simulation may be single stepped (hour), allowed to run for a defined period (days) and may be paused and restarted. It can be set to stop on each death, and stops when all members of the group are lost.

3.3 Energy model

The energy state of a simulated individual will be indicated in "hours remaining until death" (e_i, for an individual i). Clearly, once an individual dies it plays no further part in the simulation, but it remains visible for inspection. For all the simulations presented here, energy consumption (e_{cycle}) will be set at one unit/hour. The simulation assumes a linear energy consumption model. As the bats are homoeothermic their energy requirements over basal need are largely determined by the ambient temperature and the activities (including flight) they perform. Wilkinson's (1984) weight model will therefore be taken as predictive, rather than proscriptive.

Individuals are assumed to attempt to feed once in every 24-hour cycle. Feeding success will be simulated as a "probability of success" (*feedrate*). Based on Wilkinson's data, a feedrate of 93% (7% fail) will be used as a baseline value for non-dependents. By controlling this success rate in simulation, effects of energy deprivation on the individual or group can be investigated.

It will be assumed that an individual who feeds does so to satiation (e_{max}), and will have 60 units (i.e.

hours) of energy available for use or to share. Unless it feeds or is assisted an individual dies after 60 hours. This simulation only takes account of death through energy starvation (when $e_i = 0$); a more detailed simulation would overlay an additional morbidity function including other causes, such as disease, predation and mischance.

3.4 Energy transfer model

Four additional parameters are used to control the energy transfer strategies emulated. t_{need} defines the energy level at which an individual in need will seek a food donation. t_{have} defines the energy level above which an individual is prepared to make a donation. $t_{transfer}$ defines the number of units of energy that will be transferred from donor to recipient during a donation. Lastly, $t_{efficiency}$ defines the energy loss in the transfer mechanism, as a percentage transferred.

Values for these figures have been established empirically (not further reported here), using Wilkinson's (1984) figures as a starting point. All the simulations presented here use t_{have} = 28hours, t_{need} = 24h and $t_{transfer}$ = 3h. The system is stable when $t_{have} > t_{need} + t_{transfer}$.

A value of less than 24h for t_{need} leaves an increasing number of individuals at risk of being unable to forage and so must die if transferable resources dry up. The effect is detrimental to the group as a whole. Higher values of t_{have} leave the group unnecessarily exposed, as energy that might be transferred to save a fellow is retained, to no particular advantage to the potential donor, as it is still guaranteed a further foraging opportunity.

3.5 Reciprocation mechanism

At each cycle, a "haves" list of individuals ($e_i > t_{have}$) and "needs" list ($e_i < t_{need}$) is formed. Needy individuals are matched to potential donors at each cycle, the process repeating until all potential recipients are fed to t_{need} in increments of ($t_{transfer}$*($t_{efficiency}$/100)) or until no individual has energy reserves greater then t_{have}. The reciprocation mechanism changes the order in which this distribution occurs. Dependents are always fed preferentially to non-dependents.

The model offers four alternative approaches to sharing between individuals, referred to as (i) *Random*, (ii) *TitForTat*, (iii) *Delta* and (iv) *Associative*. Random mode (control, no explicit reciprocation) simply transfers energy from the needs to have lists randomly. TitForTat records the instantaneous imbalance between donations and receipts between each pair. Delta mode employs a variant of the standard delta rule (prevalent in reinforcement learning methods), a_{bal} gives a record of the balance of donations between any two individuals i and j:

$a_{bal\text{-}ij} = a_{bal\text{-}ij} + \alpha * (1 - a_{bal\text{-}ij})$ if $\text{receipt}_{i\text{-}j}$, and
$a_{bal\text{-}ij} = a_{bal\text{-}ij} - (\alpha * a_{bal\text{-}ij})$ if $\text{donation}_{i\text{-}j}$
$a_{bal\text{-}ij} = 0.5$ initially, $(0 \leq \alpha \leq 1)$

A value of 0.5 indicates parity between i and j, >0.5 indicates a debt to the partner, <0.5 indicates a credit, defaulters asymptotically approach zero. This formulation was also employed in the earlier objective trust based agents (o-TBA) model (Witkowski *et al.*, 2000). Delta mode is used as the primary method in the results that follow, bringing a good balance between responsiveness and stability to the simulation.

Associative mode records the complete history of the transactions between the pairing:

$a_{bal\text{-}ij} = \log(\Sigma_{ij}\text{receipts} / \Sigma_{ij}\text{donations})$

Zero indicates parity, <0 indicates a credit. It is sensitive about zero to small imbalances, but becomes increasingly stable as the long-term sharing association between each pair develops. This has consequence for an individual's reaction to defectors – section 5.

Once dependents have been maintained (iterative transfer from an arbitrary have to the most needy) the reciprocation mechanism is run as an iterative "beauty contest" between non-dependents, in which an arbitrary member of the have list selects the member of the need list to which it has the greatest debt to transfer to. In a community of like members, this leads to a highly stable configuration. Under conditions of plenty, individuals may build credit. Defectors, however, build debt. Once $a_{bal\text{-}ij}$ falls to a threshold a_{thresh} transfer ceases. Lower values of a_{thresh} indicate a higher tolerance to defection (actual values are strategy dependent). Under this rule a defector may rejoin the group by starting to share again. The investigations in section 4 used an unattainable value for a_{thresh} to remove any effects of cumulative "bad-luck" in feeding being interpreted as defection.

The degree of association between i and j will be given by:

$a_{assoc\text{-}ij} = \Sigma_{ij}\text{receipts} + \Sigma_{ij}\text{donations}$

Note that this is different from Wilkinson's notion of association: the sampled ratio of times i and j were observed roosting together.

3.6 Interface and display

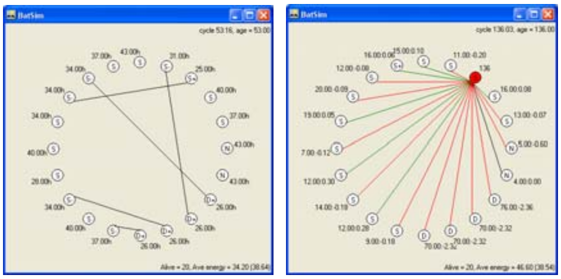


Figure 2: Simulated Sharing

Figure 2 (left) shows the simulation display. Each individual is represented by a circle; the lines denote a transfer between individuals on the current cycle. Deceased individuals are marked with an "X". The number beside each circle indicates the current e_i value for the individual. The current cycle (ddd:hh) is indicated in the top right, with the average age of the group. As individuals die their contribution to average age becomes static. When the whole group dies or the simulation ends this value may be used to indicate the relative longevity of the group as a whole, compared to other simulation runs.

Figure 2 (right) presents an alternative view, occasioned by holding the cursor over an individual, showing the a_{bal-ij} and $a_{assoc-ij}$ relationship between two individuals i and j. Green lines are used to indicate credit, red debit.

The parameters described can be adjusted on a "control panel" for the simulator (not shown). Complex or lengthy experimental procedures have been automated with pre-coded scripts and the results recorded automatically. The simulator is implemented in C++ and runs under the Windows operating system.

While simulations in this form provide a useful tool to explore the varying effects of parameters and strategies, they should only be taken as complementary to, rather than a substitute for, formal analytical techniques.

The results section, next, describes a number of complete simulation runs, each designed to test some aspect of the trophallaxis mechanism inspired by the vampire bat, and which may prove to be of value in establishing and evaluating viable energy sharing behaviours for communities or swarms of energy critical robots.

4 Results

The simulator is used here to evaluate a number of factors relevant to the sharing of energy resources by energy critical individuals formed into a group. The first experiment confirms the vulnerability of the group to starvation without any form of energy transfer. The second experiment confirms the effect of a simple energy sharing strategy among a group of adults. The third investigation considers the effects of dependents (infants, for instance) on the energy balance of the group under different food availability conditions. The fourth investigates Wilkinson's assertion that benefit/loss of sharing must be asymmetric by evaluating the effects of loss during transfer (spillage, etc.) The fifth considers the effects of non-sharers, those that take donations, but do not contribute. Finally, the value of reciprocal behaviour in managing non-contributors is considered.

Note that all these simulations are "one-shot". The group starts with a fixed number of members, and the average time they survive is recorded using different conditions and settings. Each simulation is repeated 20 times with a different started seed to the random number generator to obtain an average result (there is considerable variability between runs under identical conditions due to the randomisations used).

Unless otherwise noted each simulation lasts the equivalent of 10 years (3650 days), reflecting the upper limit of natural lifespan for a vampire bat in the wild. Any group that consistently lasts without loss to starvation for the full period is deemed energy stable. Any group that shows partial losses is considered quasi-stable, but vulnerable to transient shortages, any that are lost completely are considered energy unstable. The average age of the population (previously described) at the conclusion of the experiment is generally used as a measure for comparison.

4.1 Life expectancy without trophallaxis

This investigation confirms the extreme vulnerability to starvation of the common vampire bat under observed conditions without any compensating strategy. Figure 3 demonstrates the effect of survivability for groups of 20 individuals (average of 20 repetitions), under the conditions where each individual in the population has a 93%, (7% fail, as noted by Wilkinson for adults), 90%, 80% and 70% (average for juveniles) chance they will obtain a full meal on any given attempt. For this simulation sharing is disabled, other parameters not used.

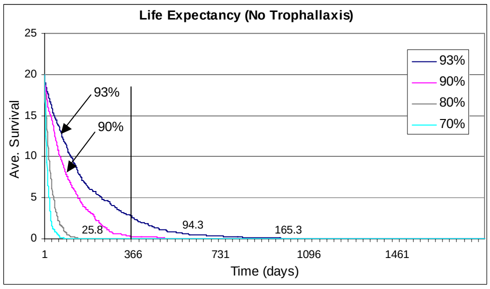


Figure 3: Life Expectancy (No Trophallaxis)

These results confirm Wilkinson's assessment that at 7% feed fail rate, a population would fall to 80% of its original size within one year, clearly insufficient to sustain a population with a birth rate of one pup /female/year. The prognosis for dependents (<70% feedrate) is dire, none surviving after 108 days, the majority lost within one month (this, of course, assumes no parental care, but their situation is little better). The numbers at the ends of the curves indicate the average age of the group at the end of the run (165.3 days for 93%). The mortality rate here simply reflects the probability of two consecutive feeding failures.

4.2 Effect of trophallaxis

This investigation repeats that of the last, with the sharing mechanism enabled. In this case feedrate is set to 93%, 70%, 65%, 60% and 50% to determine the effects of survivability (intermediate values omitted for clarity). R_{share}=100%, sharing enabled, e_{max}, e_{cycle}, e_{need}, e_{have} and $e_{efficiency}$ all set to defaults.

Figure 4 shows that all groups with a feedrate of 70% or above survived completely. At 65% some of the groups fail, at 60% the groups are clearly unsustainable, and at 50% failure is rapid. The beneficial effect of trophallaxis is very pronounced in this case; inevitable decline is replaced by robust sufficiency over a wide part of the energy input range. No more energy is input into the group as a whole, it is just utilised more effectively to the mutual benefit of all members of the group. Proof by exposition, if it is required, that this straightforward behavioural modification can completely compensate for the inability to store food energy, an otherwise fatal digestive mal-adaptation in the vampire bat.

It may be observed that complete groups tend to fail, rather than a simple diminution across groups. This is manifest in the step nature of the averaged curves presented, it is more apparent when individual groups are observed. This appears to be due to a run of poor feeding nights killing several members of the group, followed by further effects due to group size. Groups close to their energy limit being more vulnerable than others, as feeding success in the model is determined at random on an individual basis, not for the group as a whole. These results represent an ideal energy situation, several factors including losses during transfer and care of infants are considered later.

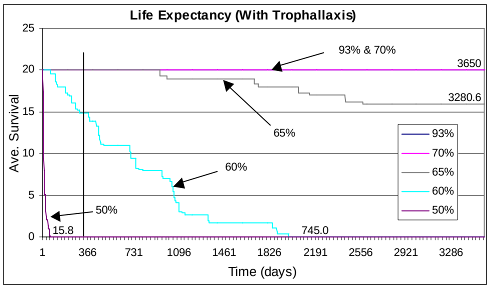


Figure 4: Life Expectancy (With Trophallaxis)

4.3 Effect of dependents

This experiment uses the simulator to investigate the effects of dependents on the energy survivability of the group. It is assumed that sharers must feed dependents if they are to survive, and that they will be fed preferentially before energy is distributed among those sharers, who may need donations also.

The experiment varies two parameters, the feedrate (93%, 90%, 85%, 80%, 75% and 70%), and the number of dependents in the group ($R_{dependent}$) is varied from 4 to 10. The remainder of the group are sharers. The other simulation parameters were left at their defaults for this run.

Figure 5 shows the combined effect. With each sharer contributing at their estimated normal feedrate (93%), a group of 20 can sustain at least seven dependents, and is only very slightly vulnerable with eight. Some of the group will be inexperienced, not capable of the full contribution, but at a joint feedrate performance of 80%, the group can still support four dependent individuals.

Reference to bat gestation and nursing times suggest that a group of this composition would be supporting 4-5 nursing pups and a number of juveniles at any one time. As before, each step in the feedrate reduces the energy available to the group, and each infant places an extra (if necessary) burden on the group.

These results indicate that the vampire bat should incorporate a mechanism to ensure that births do not become highly synchronised, as the extra energy requirement could cause the group to become critical. Some bat species do synchronise births to seasons when food is more plentiful, and the vampire birth rate may show seasonal variation in some parts of its Central and South American range.

In nature, it is observed that the majority of vampire bat regurgitations are between an adult and infant, usually between mother and her pup.

Robots, as yet, do not support their young, but these results may be seen as a metaphor for a robot community or swarm where foragers must support some other form of specialised robot, but still within the energy critical scenario. Alternatively, one might imagine a situation where lost robots are replaced by naïve ones from stock, who must be supported while they learn strategies for survival within the prevailing environment.

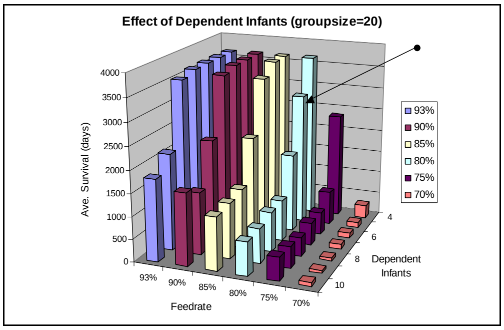


Figure 5: Effect of Dependent Infants

4.4 Energy transfer efficiency

This investigation considers the effect of transfer efficiency on the group. Wilkinson (1984) makes much of the observation that the recipient appears to gain more than the donor loses (figure 1). This may be so, but is it an essential property of the sharing process? As previously noted, this simulation does not model any such advantage (although it could), choosing energy exchange parity.

This experiment explicitly models the effects of energy loss during transfer, due to spillage or other losses and its effect on survivability. A feedrate is chosen at the boundary of what the group will support (70%), and five runs (repeated 20 times) made with the $e_{efficiency}$ parameter successively set to 100% (full transfer), 90%, 80%, 70% and 60%. For comparison, all the remaining parameters are set to the same as those of figure 4 at the 70% feedrate (i.e. the 100% line is equivalent in both figures).

From figure 6, and by direct comparison with figure 4, it can be seen that the 80% transfer efficiency is largely equivalent to survivability to the 65% feedrate (from 70%) and 60% efficiency to the 60% feedrate.

From this it may be concluded that no special status need be attached to transfer efficiency, either as a requirement for mutual trophallaxis, or as anything other than a drain on the energy on the system. Using this observation, the group would be able to support fewer offspring concurrently, or be more vulnerable to famine – but the basic tenet of energy sharing is unaffected. This is in keeping with the observations of Kubo and Melhuish (2004) in a simulated robotics context.

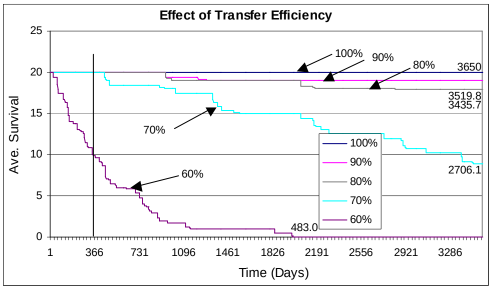


Figure 6: Effect of Transfer Efficiency

4.5 Effect of defectors

In all the investigations so far, all individuals have behaved "honestly", what happens if some proportion of the group take the assistance when they need it, but do not reciprocate? This is, of course, the conundrum set in the "evolution of altruism" literature (Axelrod, 1984; Dawkins, 1989). Figure 7 shows the effect of increasing numbers of non-sharers (from none to 11) in a group of 20 with four dependents, community feedrate = 93, 90, 85, 80 and 75%, other parameters default. As a point of comparison, note that the "0" defector line of figure 7 is equivalent to the four dependent line of figure 5.

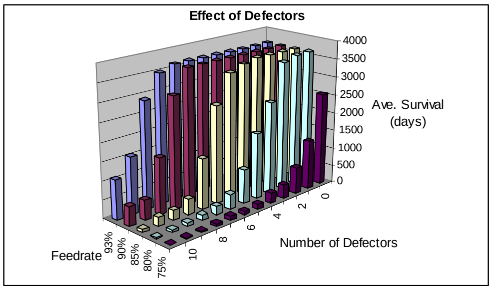


Figure 7: Effect of Defectors

The impact of a defector in the group is not as dramatic as an infant, as they will feed themselves at the normal rate, but acts as a drain on the community's resources. At a feedrate of 93%, the community will support six defectors without immediate ill effect, at lower feedrates, predictably, the number of defectors that can be supported falls.

It might be imagined that the expression of any gene that leads to defection in this sense lends little short-term advantage to the defector, in that (according to the strategy presented here) each contributor's energy resource never falls below the reserve required for at least one further forage.

The net effect is to make the whole group increasingly vulnerable to transient shortages. When the group fails the defector is (nearly) as vulnerable as any of the others, and it might be expected any "defection" gene to be selected against on this basis alone. The next section considers additional safeguards provided by reciprocal feeding behaviours.

5 Why Reciprocation?

The main postulated purpose of reciprocation in trophallaxis is to guard the population against any incursion by individuals with a predisposition to "defect", to take the benefits of sharing, but not to return them. Both the associative and delta methods previously described are highly intolerant to non-reciprocators, provided the threshold step is in place. Without this, as with the random sharing strategy, the defaulter may continue to receive the benefits without the costs, enjoying, on balance, a higher mean energy level than its peers, who are otherwise struggling to maintain the energy balance of the larger group. Set against this is the presumed need to protect individuals, who while normally reciprocating are temporarily unable to feed.

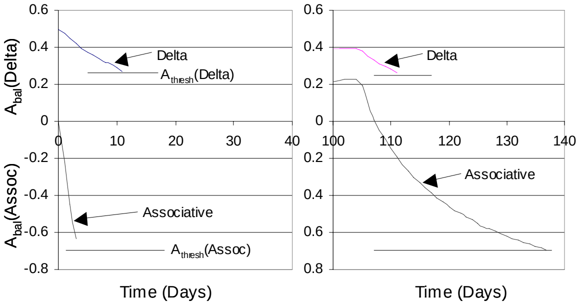


Figure 8: Effect of Reciprocation Strategies

Figure 8 shows the effect when individuals in the group were programmed not to reciprocate, and also not to feed, under both the Delta and Associative modes. The fall in aggregate support from the remainder of the group is charted against time, when support reaches a_{thresh} (= 0.25 for Delta, α = 0.1; and –0.7 for Assoc.) no further feeding support will take place. It may be seen that both strategies are relatively aggressive at locking the defector out of the group at the beginning of the sequence (left); death follows at 10.8 days, s.d. = 0.46, n = 10 for Delta and at 5.63 days, s.d. = 0.23 for Associative.

The Associative strategy is far more tolerant than Delta for an individual that has been a functioning part of the group for a period of time (right, after cooperation for 100 days, death follows at 33.7 days, s.d. = 6.84, n = 10). Any individual that regains its ability to feed may rejoin the group without apparent long-term effects. It should be noted that individuals that continue to feed at their normal rate survive for a period of time after loss of group support (following figure 3).

This is more significant. As males reach maturity they cease to reciprocate, and usually revert to a shorter, largely non-sharing, existence. Females continue to share and become self-sufficient as a group. Males need only do their duty; females must rear young. Any female that does not share reverts to the unsupported category, and will hardly be able to rear one pup, and so be highly disadvantaged in the gene pool. The consequences of partial or intermittent defection have yet to be investigated.

6 Discussion

That vampire bats share blood meals by regurgitation among their immediate group is well established (Wilkinson, 1984) and its beneficial effects confirmed here. The evidence for actual reciprocation is based on a small number of observations in a contrived experimental situation and, perhaps more significantly, the presumption that it must be so from evolutionary theory. In general, it has not been possible to replicate

Wilkinson's experimental results assume a single regurgitation in exchange for one received previously. He only noted regurgitations for two hours directly after reintroduction of the hungry animal to the cage. The simulation would strongly indicate that several more donations would have been required to sustain the animal until the next feeding opportunity. Further work is needed to resolve this.

Vampire bats are also highly social, and reciprocation may also have significance far beyond its immediate feeding implications in terms of social bonding. It might be noted that male and female adults adopt very different social behaviours. While females are highly gregarious, active males generally seek territories to defend and generally are not observed to cooperate, though there is some evidence that they may do so in male only "bachelor roosts".

While there is no incontrovertible evidence that common vampire bats use reciprocation as a mechanism to defeat non-contributors, it is clear that any such reciprocation mechanism must be suppressed for infants and juveniles, and introduced in a controlled manner as adulthood is approached if the individual is not to be rejected from the group.

Reciprocation implies recognition between individuals and effective memory. This is taken as a given in the robot scenario, as robots may communicate directly or be fitted with a suitable detectors. Several mechanisms for inter-bat recognition have been proposed, including auditory signals and olfactory cues; memory abilities are not known. Wilkinson reports that regurgitations are often preceded by bouts of grooming, possibly for identification and probably as a cue to initiate transfer.

7 Conclusions

It is clear that bat social behaviour is complex, and it has only been possible to focus on a relatively small number of aspects of this here, mostly relating to energy management for the purposes of formulating an abstracted simulation. Results have confirmed the central role of trophallaxis in this species, modified behaviour compensating for an inability to store food energy, and its potential value in robotic swarms. The role of transfer efficiency and dependents and non-sharers has also been considered, leading to a discussion of the role of reciprocation in managing the behaviour of the group as a whole.

While more immediately applicable to artificial devices, the requirements for inter-individual recognition and memory of past events to achieve these idealised algorithms seems to put an unrealistic behavioural burden on the bat. Further work is required to better understand the mechanism of, and the role trophallaxis plays in the overall social organisation of the animal.

References

Axelrod, R. *The Evolution of Cooperation*, New York: Basic Books, 1984.

Clutton-Brock, T. Breeding Together: Kin Selection and Mutualism in Cooperative Vertebrates, *Science*, **296**:69-72, 2002.

Dawkins, R. *The Selfish Gene*, Oxford University Press (extended second edition, 1989, first published 1976), 1989.

Ieropoulos, I., Melhuish, C., Greenman, J. Artificial Metabolism: Towards a True Energetic Autonomy in Artificial Life, in *7th Euro. Conf. on Artificial Life*, Springer-Verlag LNAI 2801, 792-799, 2003.

Ieropoulos, I., Melhuish, C., Greenman, J. and Horsfield, I. Artificial symbiosis: Towards a robot-microbe partnership, in: *Proc. "Towards Autonomous Robotic Systems" (TAROS-2005)*, Imperial College London, 89-94, 2005.

Kubo, M. and Melhuish, C. Robot Trophallaxis: Managing Energy Autonomy in Multiple Robots", in: *Proc. "Towards Autonomous Robotic Systems" (TAROS-2004)*, University of Essex, 77-84, 2004.

McFarland, D. and Spier, E. Basic cycles, utility and opportunism in self-sufficient robots, *J. Robotics and Autonomous Systems*, **20**:179-190, 1997.

McNab, B.K. Energetics and the Distribution of Vampires, *J. of Mammalogy*, **54**(1):131-144.

Trivers, R. *Social Evolution*, Benjamin-Cummings Publishing Co., 1985.

Wilkinson, G.S. Reciprocal food sharing in the vampire bat, *Nature*, **308**:181-184 March 1984.

Wilkinson, G.S. Food Sharing in Vampire Bats, *Scientific American*, 64-70, February 1990.

Wilkinson, S. Gastrobots – Benefits and Challenges of Microbial Fuel Cells in Food Powered Robot Applications *Autonomous Robots*, **9**:99-111, 2000.

Witkowski, M., Artikis, A. and Pitt, J. Experiments in Building Trust in a Society of Objective-trust Based Agents, in: Falcone, R., Singh, M. and Tan, Y-H. (eds.) *Trust in Cyber-Society: Integrating Human and Machine Perspective*, Springer LNAI 2246, 111-132, 2000

Modulation of exploratory behavior for adaptation to the context

Arnaud J. Blanchard⋆
A.J.Blanchard@herts.ac.uk

Lola Cañamero⋆
L.Canamero@herts.ac.uk

⋆Adaptive System Research Group
School of Computer Science
University of Hertfordshire
College Lane, Hatfield, Herts AL10 9AB

Abstract

For autonomous agents (children, animals or robots), exploratory learning is essential as it allows them to take advantage of their past experiences in order to improve their reactions in any situation similar to a situation already experimented. We have already exposed in Blanchard and Cañamero (2005) how a robot can learn which situations it should memorize and try to reach, but we expose here architectures allowing the robot to take initiatives and explore new situations by itself. However, exploring is a risky behavior and we propose to moderate this behavior using novelty and context based on observations of animals' behaviors. After having implemented and tested these architectures, we present a very interesting emergent behavior which is low-level imitation modulated by context.

1 Introduction

For autonomous agents (children, animals or robots), exploratory learning is essential as it allows them to take advantage of their past experiences in order to improve their reactions in any situation similar to a situation already experimented.

In previous work (see Blanchard and Cañamero (2005)), we developed a biologically inspired architecture to make a robot learn which perceptions it should try to reach in order to maximize its comfort (i.e. minimize the distance of internal variables to ideal values). These perceptions (called *desired perceptions*) are memorized either because of pleasantness or familiarity associated with them. They correspond to the zones of comfort in the sense of Likhachev and Arkin (2000). The balance between the importance of the familiarity and the pleasantness has been managed using different time scales selected in function of the level of comfort of the robot. The resulting behavior is interesting in robotics as it allows an autonomous robot to learn which perception it should try to reach, but it is also interesting in biology because it models the imprinting phenomenon, a behavior the ethologist Konrad Lorenz first noticed in the 1930's: He showed that animals, like geese, automatically follow the first thing they see (usually the mother). This behavior seems very important in animals' life, and Bateson and Martin (2000) show the phenomenon is not as simple as it seems to be; animals can adapt themselves and modify the thing they follow depending on their experiences (pleasantness or familiarity). Our previous architecture modeled this behavior as well.

However, if we do not interact with the robot, the robot does not have any opportunity to experience any new situation and will try to reach the most familiar and pleasant perception it had, which is the one it always had and therefore it will not move. This is a problem as it prevents the robot from learning anything in a static environment and it does not model behaviors observed in nature, as animals are always looking for novelty (Panksepp (1999)). Moreover, throughout evolution the young of many species still devote a great deal of time and energy to play despite the risks (e.g. injury, meeting predators, energy expenditure). Therefore play must have important biological functions in influencing the rate of survival and ultimately, success in later reproduction (Power (2000)). We see play as a way of exploring but there are some other possibilities to encounter new situations such as moving randomly or imitating other agents; in any case the difficulty is to learn without taking too many risks. We are therefore interested to build architectures biologically inspired allowing a robot to take initiatives to explore in the right context and look for novelty. In the Oxford dictionary, *novelty* is defined as "a new or unfamiliar thing or experience" and our working definition is a non predicted sensation where sensation corresponds to the input of the sensors.

We see the fact that exploratory behaviors are more likely to occur in a familiar context (Dunn (1977); Likhachev and Arkin (2000)) and that we automatically imitate more often the persons with whom we have stronger affective bonds (Hatfield et al. (1994)) as a way of balancing learning and risk of exploration depending on the environment. Therefore, in this paper we propose models for a new approach of development where exploration and imitation depend on affect. In the Oxford dictionary, *affect* is defined as "emotion or desire, especially as influencing behavior or action" where emotion is an "instinctive or intuitive feeling as distinguished from reasoning or knowledge". Our working definition is that affect is an immediate or instinctive evaluation of a situation (positiveness or negativeness) without direct or logic explanation. We can evaluate affect using for example the proximity of the agent (child, animal or robot) to an object of attachment (Likhachev and Arkin (2000)) or to a desired perception which can have been learned through the experience (Blanchard and Cañamero (2005)). The purpose of this paper is to propose a mechanism to select a kind of behavior (exploration, exploitation or imitation) rather than a specific action, as it is more often the case in action selection architectures, in order to increase autonomy in robots and to better understand the development of children or animals.

In section 2, we present an architecture producing an exploratory behavior with moderation of the novelty. In section 3, we complete the architecture in order to take the affect into account and modulate the explorative behavior. Finally in section 4, we show how this architecture produces low-level imitation depending on the affect. All the architectures respect our bottom-up approach: we try to make them as simple as possible and we add progressively new elements to make them satisfy or explain new features. To represent the architectures, we use a standard representation in engineering (SADT also called IDEF Knowledge Based Systems (2004)) where each process is represented by a box. The inputs of each process come from the left, the parameters come from the top, and the outputs exit from the right of each box. Each box can be composed by many sub-boxes, i.e. a process composed by many sub-processes. We have implemented the architectures on a real robot and we will present the data of one representative experiment for each architecture.

2 Explorative behavior

2.1 Principle

In Blanchard and Cañamero (2005), our robot was able to learn which perceptions are associated with comfort, and was trying to reach these desired perceptions. The problem was that if the experimenter did not actively put the robot in new situations, the robot would never experiment new perceptions. It would stay motionless having the best perception it knew, which is actually the only one it had experienced. However, studies (e.g. Panksepp (1999), Power (2000)) show that *novelty* is a primary need in animals motivating the animal to explore. The term *novelty* can have different interpretations but we define and use novelty as the mismatch between the actual sensation and the predicted sensation. Fig 1 describes how novelty is computed depending on a static parameter defining the *sensitivity* of the matching detector.

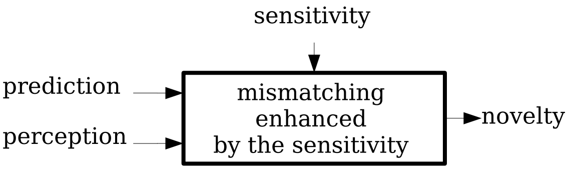


Figure 1: Representation of novelty.

Seeking novelty is not only a phenomenon observed in animals, but it is also a feature essential in autonomous robotics. It allows robots to be self-motivated to learn (Steels (2004); Kaplan and Oudeyer (2004)) and experience different situations even in a static world. Nevertheless, exploration and seeking for novelty should not become the main behavior. First, it is hazardous, for a robot or an animal, to be in a situation totally new and totally unpredictable. Secondly, in order to learn, animals and robots need a progressive increase of novelty (Kaplan and Oudeyer (2005); Oudeyer and Kaplan (2004)). Therefore, we need a process to inhibit exploration when there is too much novelty.

A simple way of implementing an explorative behavior is to set a primary need or deficit of novelty raising a motivation to explore (which can be satisfied by generating random actions, wandering around, etc.) increasing with time. When something unexpected happens, the motivation to explore is reset as the situation is not "boring" anymore (Fig 2). The actions raised by the exploration's motivation are used to create a homeostatic control of the novelty.

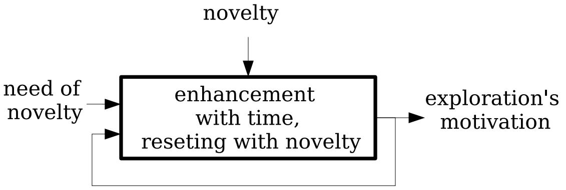


Figure 2: Representation of the exploration's motivation.

2.2 Implementation

2.2.1 Architecture

In order to check wether our simple theoretical models are applicable in reality and provide the desired behavior, we have implemented them in a real robot. The robot is a Koala K-Team (2002), a six-wheeled robot with long range distance sensors at its front. In this implementation, the considered sensation of the robot is its distance to a box at its front; its possible action is to set its velocity. As the relationship between the velocity and the distance is not direct (the velocity is proportional to the derivative of the distance), we use the sensation of velocity which is the difference between the actual sensation of distance (S_d) and the temporally smoothed sensation of distance ($\overline{S_d}$). This simplifies our problem and it has been shown that this pretreatment happens in biology; for example in the retina, some neurons are activated uniquely by the sensation of motion in the visual field. To temporally smooth values, we use equations like in (1) where the value to smooth is S_d and the value of the parameter ϵ is comprised between 0 and 1; smaller it is, more important is the smoothing. In our cases we use a value of 0.1 and we represent each smoothed value by over-lining it.

$$\overline{S_d} = \overline{S_d} + \epsilon.(S_d - \overline{S_d}) \qquad (1)$$

To show the principle of our architecture, we need some kind of predictor (P) even if it is not accurate. Therefore we use a very simple one which predicts a null velocity at any time, this corresponds to what happen most of the time (nothing moving). We compute the novelty (n) by calculating the error (er) between the prediction of velocity (P_v) and the actual sensation of velocity (S_v). The novelty is a value between 0 (no novelty at all) and 1 (maximum of novelty) the speed of convergence to 1 depends on a parameter (s) representing the sensitivity of the robot to unpredictable sensations. When the novelty is too low, it will be raised by a movement forward of the robot activated by the motivation to move forward for exploration ($M_{f(e)}$). We present the implemented architecture Fig 3 (next page) where non novelty ($\neg n$) equals one minus novelty (n) and e represents the exponential function. The output on the right of each box is defined by the result of the equation inside the box where inputs are the incoming arrows. It is not represented in the figure, but before going to the motors, the motivation to move forward is temporally smoothed in order to avoid sharp movements.

2.2.2 Experiments

The aim is to make our robot explore by itself the different possible distances to a box. Therefore, we put the robot at about 80 cm (maximal distance detection) of a box, and we start it without doing anything else (see Fig 4).

Figure 4: Setup of the experiment with the target box on the left and the Koala robot on the right.

We present in Fig 5 the successive position of the robot toward the box during three representative experiments among ten similar for four different values of the sensitivity parameter (250, 500, 1000 and 2000). The range of values (S_d) given by the sensors are varying from 50 when the box is at about 80 cm from the robot to 1000 when the robot touches the box.

It is interesting to notice that the robot has a behavior similar to those observed in animals' approach behavior. During the approach behavior, the robot moves then stops, moves again and so on as an animal would do. The robot's behaviors would be even more similar to animals' behaviors if the predictor were able to learn and do better and better predictions; in this case, the robot would inhibit less and less its explorative behavior, as it would have less and less novelty (better prediction). It is the same with an animal which becomes more and more confident with habit (better prediction).

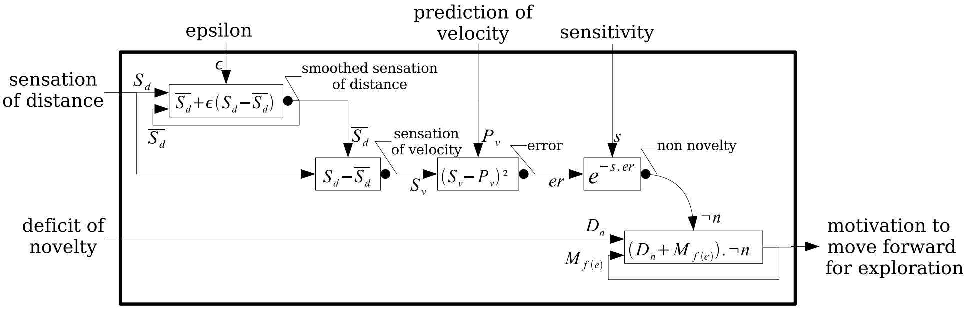


Figure 3: Representation of the simple explorative behavior.

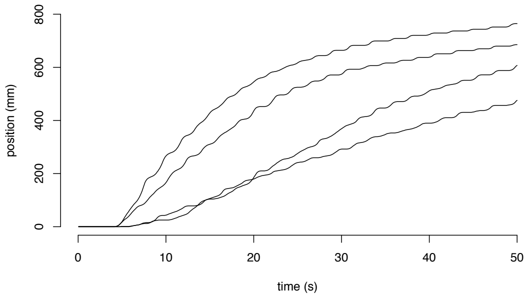


Figure 5: Progression of the robot toward the box for four values of sensitivity, (250, 500, 1000 and 2000) for curves from top to bottom respectively.

3 Perseverance and retraction

3.1 Principle

Exploration is an advantage only if the robot is able to exploit the discovered situations. It should continue explorative actions leading to positive results (positive affect), and on the contrary avoid or even cancel exploratory actions with negative results (negative affect). Therefore, we use an association between perception and action allowing the robot to enhance a new perception when the affect is positive or on the contrary to reduce it when the affect is negative. In our case, this association is hard coded: for example, an unexpected perception of the box coming closer activates the command of moving forward when affect is positive (to increase the new perception) and the command of moving backward when affect is negative (to reduce this new perception). However this association could have been learned during a "babbling" phase as it is done by Andry et al. (2003) and Demiris and Dearden (2005).

The exact calculation of the value of affect is out of the scope of this paper, we are interested here only on its effect on behavior. As we said it can be correlated with the proximity of the robot to a desired perception as defined in Blanchard and Cañamero (2005) or the proximity to an object of attachment in the sense of Likhachev and Arkin (2000). The direct consequence will be that the robot explores more easily when it is close to a desired perception or an object of attachment which corresponds to a familiar and positive situation. On the contrary, it will hesitate more or even go back when it is in an unfamiliar and negative situation. We can see the principle of the architecture Fig 6 (next page).

3.2 Implementation

3.2.1 Architecture

To implement this architecture on the robot, we add a mechanism to the previous architecture which is able to amplify or reduce (motivation to continue M_c) a new perception through action (motivation to move forward M_f). This does not interfere with the exploratory behavior as the exploratory behavior is inhibited when there is novelty. Whereas, the behavior amplifying or reducing a new perception is triggered only when there is novelty and therefore, merging the two signals consists of summing them together. In our case, the sensori-motor association is implemented by the fact that the value of the sensation of velocity is sent to the motors' command (M_f)through a positive or negative amplification (M_c) depending on the affect (af). The implemented architecture is represented Fig 7.

3.2.2 Experiments

We use exactly the same setup that in the section 2, but with this new architecture and for different values of affect. For all the experiments, we use an average

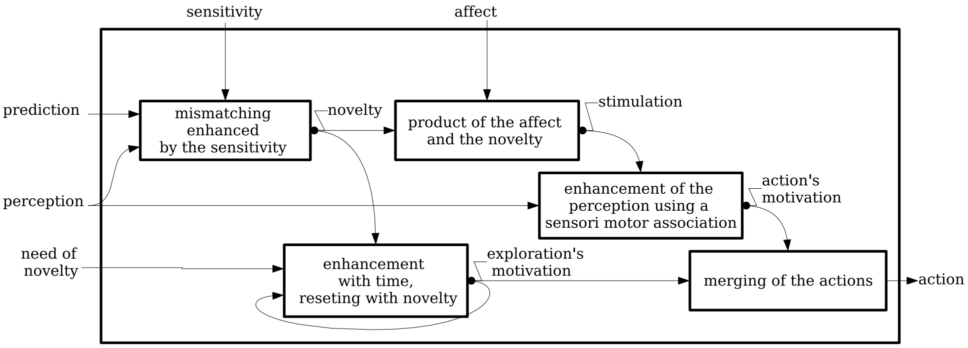


Figure 6: Representation of the exploration behavior with perseverance and retraction.

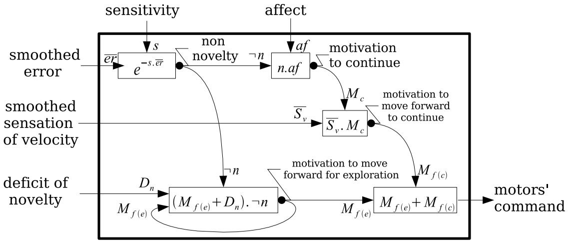


Figure 7: Architecture of the robot providing perseverance and retraction.

value (500) of the sensitivity (s) defining the "character" of the robot.

3.2.3 Results

The ideal values of affect are completely dependent on the apparatus but in order to keep the system stable the absolute values have to be strictly inferior to the quotient of the motors' command (motivation to move forward M_f) by the sensation of velocity (S_v) associated. It means that if the robot has usually a sensation of velocity x when it sends a command y to the motors, the absolute value of affect must be strictly inferior to $\frac{y}{x}$. If we do not respect this, the movement of the robot will not converge, the robot will either oscillate (when the affect is negative) or moving faster and faster (when the affect is positive). In our case, this maximum value is 0.0004 and we present in Fig 8 (next page) the successive positions of the robot for three values of affect (0, 0.0002 and -0.0003).

When the affect is null we have exactly the same behavior that with the exploration only architecture; actually the parts that we have added are totally neutralized. However, when the affect is positive we have much smoother movements, the robot seems to go more directly to the unknown situation—be close to the box. This is due to the fact that positive affect reinforces the motivation of the robot to keep the first initiated action. On the contrary with a negative affect, the robot does not just stop when a perception is new but it acts in order to avoid this new perception. It moves forward, and then as soon as something happens, it does not only stop but moves backward, and after a while moves a bit more forward and so on. We see during the last fifteen seconds in Fig 9 the comparison between the positions of the robot when the affect is negative (solid line) and when the affect is null (dashed line). This behavior is very similar to the one of an animal exploring a new space in a very unfamiliar environment.

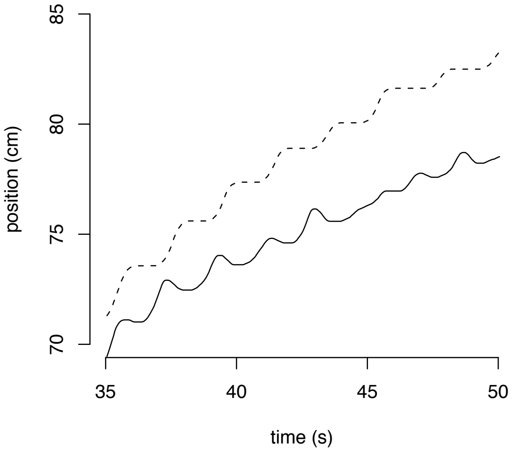
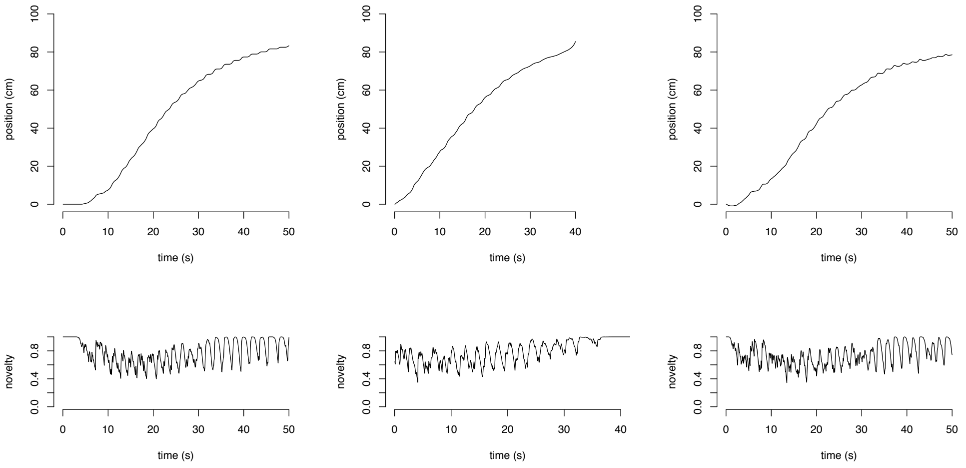


Figure 9: Comparison of the approach of the robot with negative affect in solid line (the robot is moving back sometimes) and with null affect in dashed line.



null affect positive affect negative affect

Figure 8: Approach of the robot toward a box (at 80 cm) for different values of affect (from left to right respectively, 0, 0.0002 and -0.0003). We see on the top the successive positions of the robot, and on the bottom the successive values of novelty.

4 Low-level imitation

In our previous experiments, we have shown how the exploration process can be modulated by the affect. The resulting behavior is really similar to the one observed in animals during exploration in different context (familiar or not). Moreover this simple architecture is not only interesting to generate appropriate explorative behaviors but it produces as well low-level imitation behavior depending on the type of affect. The exploratory behavior and the imitative behaviors are not interfering and the robot autonomously switches from one behavior to another one, mainly depending on the fact that there is something interacting or not.

When affect is positive and the experimenter moves toward the robot, for example, he will generate novelty for the robot and, as the affect is positive, the robot will try to increase the new perception and therefore moves forward as well; if the experimenter moves backward, the robot moves backward as well. Therefore we produce low-level imitation depending on our notion of affect, and not based on the principle of correcting an error like Andry et al. (2003) or Demiris and Dearden (2005). On the contrary, in our case imitation results as a side effect the principle to increase an error (the error of prediction) which can accelerate the learning as it contrasts the new perceptions. However when affect is negative, the opposite happens, the robot avoids any new situation and avoids the experimenter if he tries to approach the robot. When the affect is null the robot does not interact with the experimenter and moves only if it misses novelty.

5 Conclusions and perspectives

We have presented here the basis of simple architectures producing explorative and imitative behaviors useful for learning. There are four main direct interests in this work:

1. It is based and provides solutions on commonly accepted needs in autonomous agents, notably exploration and research for novelty.
2. With simple biologically plausible functions, it reproduces behaviors observed in nature, which can give some clues about the operation of the brain.
3. It allows us to build architectures managing an appropriate level of novelty for constant learning.
4. Increasing the novelty when affect is positive can accelerate learning as it contrasts the new perceptions.

Some adaptations or additions could make this architecture more interesting. First, instead of using a simple static predictor as we did, a predictor able to learn and to increase the prediction with time would make the robot more and more confident, and make it explore progressively more and more. The exploration which was simply moving ahead could be more sophisticated and proposes actions randomly, or even better the actions which maximize the learning progression Kaplan and Oudeyer (2005).

In the future, we will develop imitation as an emergent property depending on affect in order to allow learning through imitation and for human-machine interaction. Actually, the possibility of the robot to initiate actions or imitations depending on its familiarity with its partner can be useful for the development of relationship between a user and a companion robot. It will also give clues in the comprehension of the turn taking behavior. We will also study how an appropriate stimulation of the robot could be used as a reward in itself and modify affective bonds like it seems to be the case in men and animals. It is interesting that a good interaction can improve the relationship in order to have even more interaction. Finally, more work should be done in order to go from low-level imitation through more complex imitation such as imitation of sequences.

Acknowledgments

We would like to thank Dr. Carol Britton for her feedback on a draft of this paper. Arnaud Blanchard is funded by a research scholarship of the University of Hertfordshire. This research is partly supported by the EU Network of Excellence HUMAINE (FP6-IST-2002-507422).

References

P. Andry, P. Gaussier, and J. Nadel. From sensorimotor development to low-level imitation. In *2nd Intl. Wksp. on Epigenetic Robotics*, 2003.

P. Barron Bateson and P. Martin. Sensitive periods. In *Design for a Life : How Behavior and Personality Develop*, chapter 8. Simon and Schuster, 2000.

A. Blanchard and L. Cañamero. From imprinting to adaptation: Building a history of affective interaction. *Proc. of the 5th Intl. Wksp. on Epigenetic Robotics*, pages 23–30, 2005.

Y. Demiris and A. Dearden. From motor babbling to hierarchical learning by imitation: a robot developmental pathway. *Proc. of the 5th Intl. Wksp. on Epigenetic Robotics*, pages 31–37, 2005.

J. Dunn. Distress and comfort. In J. Bruner, M. Cole, and B. Lloyd, editors, *The Developing Child*, chapter Crying, Comfort and Attachment, pages 67–75. Fontana/Open Books, 1977.

E. Hatfield, J. Cacioppo, and R. Rapson. *Emotional contagion*. Cambridge university press, 1994.

K-Team. http://k-team.com/robots/koala, 2002.

F. Kaplan and P-Y. Oudeyer. Maximizing learning progress: an internal reward system for development. In F. Iida, R. Pfeifer, L. Steels, and Y. Kuniyoshi, editors, *Embodied Artificial Intelligence*, LNCS 3139, pages 259–270. Springer-Verlag, London, UK, 2004.

F. Kaplan and P-Y. Oudeyer. The progress-drive hypothesis: an interpretation of early imitation. In K. Dautenhahn and C. Nehaniv, editors, *Models and Mechanisms of Imitation and Social Learning: Behavioural, Social and Communication Dimensions*. Cambridge University Press, 2005. to appear.

Inc. Knowledge Based Systems. Idef - integrate definition methods, 2004. URL `http://www.idef.com/`.

M. Likhachev and R. Arkin. Robotic comfort zones. In *SPIE Sensor Fusion and Decentralized Control in Robotic Systems III*, pages 27–41, 2000.

P-Y. Oudeyer and F. Kaplan. Intelligent adaptive curiosity: a source of self-development. In Luc Berthouze, Hideki Kozima, Christopher G. Prince, Giulio Sandini, Georgi Stojanov, G. Metta, and C. Balkenius, editors, *Proceedings of the 4th Intl. Wks. on Epigenetic Robotics*, volume 117, pages 127–130. Lund University Cognitive Studies, 2004.

J. Panksepp. *Affective Neuroscience*, chapter 8-SEEKING Systems and Anticipatory States of the Nervous System, pages 144–162. Oxford University Press, 1999.

T. Power. *Play and Exploration in Children and Animals*. Lawrence Erlbaum Associates, Publishers, 2000.

L. Steels. The autotelic principle. In I. Fumiya, R. Pfeifer, L. Steels, and K. Kunyoshi, editors, *Embodied Artificial Intelligence*, volume 3139 of *Lecture Notes in AI*, pages 231–242. Springer Verlag, Berlin, 2004.

Towards Robotic Self-repair by means of Neuronal Remodelling

B. Torben-Nielsen*, K. Tuyls*, E.O.Postma*
*University Maastricht
IKAT, P.O. Box 616, 6200MD Maastricht,
the Netherlands
{b.torben-nielsen,k.tuyls,postma}@cs.unimaas.nl

Abstract

Adaptivity in mobile robotic systems is recognized as a difficult issue. In the invertebrate world several species are known to have remarkable adaptive behaviour triggered by environmental changes. The adaptivity is the result of remodelling their brain according to environmental needs. In this conceptual paper we propose a methodology to obtain true adaptivity in robotic systems based on neuronal remodelling in insects. Contrasting other AI related research in the field of robotics, we will use complex neuronal structures instead of standard artificial neurons. We propose three research tracks and our results from the first track are presented and discussed.

1 Introduction

Adaptivity is acknowledged to be a core problem in mobile robotics by the AI research community (Matarić, 1998). Adaptivity in this domain is the ability of a robot to adapt itself to changes in the environment. These changes are either caused by factors in the environment itself, by external factors or by a sudden malfunctioning of the robot which causes it to have a noisy perception. In this paper we solely focus on the adaptation of the robot to environmental changes induced by malfunctioning which is generally referred to as *self-repair*. An example clarifies what is meant. Suppose that a sensor breaks down or fails. Sensor failure results in perception of corrupted information and from the point of view of the robot the environment is now changed. Evidentially, not only sensing failures affect the robot's behaviour but also other kinds of failure such as a partially blocked locomotion system. Since all mobile robots possess a certain level of autonomy self-repair is an important and challenging research area.

In the past decade several solutions have been proposed for the problem of self-repair. Roughly speaking, the proposed solutions are either hardware-repair (*hard-repair*) or software repair (*soft-repair*). Soft-repair is an instance of robotic self-repair in which the control system (e.g., neural network) has the task to repair the behaviour of the robot. An example of hard-repair (also called self-reconfiguration) is the HYDRA project (Østergaard et al., 2005). In this paper we only focus on soft-repair. An exhaustive listing of proposed solutions for soft-repair is beyond the scope of this paper, however, we wish to present some ideas from the community. A popular methodology is evolutionary computation (EC) (Koza, 1992) in combination with artificial neural networks (ANN) (Nolfi and Parisi, 1997; Nolfi, 2002). In this approach parameters of a neural robot controller are tuned in a way similar to Darwinian evolution. The *artificial evolution* of robot controller is performed in real-time on real robots (Mondada et al., 1994; Nolfi and Parisi, 1997) or in simulation as in Floreano and Urzelai (2001); Torben-Nielsen et al. (2005a). Plastic Neural Network (PNN) are often proposed as solution; they no fixed weights in the neural network but rather update the weights on-line during task execution (Floreano and Urzelai, 2001). PNNs proved to be highly adaptive within evolved strategies (Torben-Nielsen et al., 2005a).

It is worth noting that most of the aforementioned techniques of soft-repair are inspired in some way by biology. Recently, a new research field, *biorobotics*, has arisen in between engineering (e.g., robotics) and biology (Webb and Consi, 2001). The goal of biorobotics is twofold, it aims at being beneficial for both engineering and biology. For engineering, biology has proven to be an almost inexhaustible source of inspiration. For biology, synthesised models of biological systems can be used to test, falsify, or generate hypotheses (Webb, 2002). However, in most biorobotic projects the aim is either engineering or biology. In the domain of soft-repair several biorobotic projects were conducted. For example GasNets by Husbands et al. (1998) who used a recently discovered model of gas emission as extra inter-neuronal communication leading to faster evolution of desired controllers (Smith et al., 2002). Another example directly aimed at damage recovery in mobile robots is the application of a neuronal activity-dependent model on a robot platform (Elliott and Shadbolt, 2001).

The results show that these solutions work quite well in a highly specific environment while performing a predefined task. Although some remarkable techniques are proposed to tackle the problem of self-repair, self-repair in highly dynamic and noisy environments has not been achieved yet. With respect to the earlier listed results, we can post a general requirement that needs to be satisfied in order to achieve reasonable capabilities of soft-repair: *on-line generation of new behaviours.* This requirement can be divided into two parts. The first part *on-line* refers to the fact that a robot must be able to repair itself during job execution. It is not feasible to return the robot to the lab, repair or re-train the robot and put it back into the environment. The second part *generation of new behaviours* refers to the fact that adaptation within a single behaviour is not sufficient; a completely different behaviour can be required. As an example, think of a robot that has to move towards a light source with damaged light sensors. In this case, using a temperature sensor might result in successful performance which is achieved by a completely different behaviour or strategy. However, note that the term *new* does not imply that the new behaviour is developed from scratch but rather that it is new and different with respect to the failing behaviour.

In this paper we propose a new methodology to tackle the problem of self-repair. We propose a method based on the mechanism exhibited by specific invertebrates that remodel their central nervous system to meet new environmental requirements. Our proposed methodology consists of three research tracks: (i) constructing a neuronal model that allows information processing similar to certain computations in real neurons, (ii) automated remodelling of the generated neuronal morphologies, and, (iii) embedding the results from track (i) and (ii) on a robot platform. Our proposed method is original in the future employment of a new artificial neuronal model. The model will be based on realistic neuronal morphologies and plausible information processing in the neurites.

The next section explains the biological inspiration of our work as well as the biological background required to understand posted assumptions and decisions. Section 3 presents our new approach to tackle the problem of self-repair. Three research tracks are formulated and results from the first track are presented. We present a conclusion in Section 4.

2 Biological inspirations

In the invertebrate world, several species are known to change their central nervous system (CNS) upon changing environmental needs. The remainder of this section elaborates on the phenomenon of *neuronal remodelling* (e.g., Duch and Levine (2000)). Two specific cases are presented as they are relevant for the conceptual application of neuronal remodelling in the robotics domain.

First, the tobacco hornworm (*Manduca sexta*). The Manduca belongs to the class of metabolous animals that change shape during their lifetime. Moreover, holometabolous animals (like the Manduca) change the shape of their body completely in different life phases (Consoulas et al., 2000; Tossit and Stocker, 2000). The animals go from a larva phase where they are caterpillars, to a pupal phase in which they do not move and prepare for the metamorphosis to finally end as a moth. It is not hard to see that the different life phases require different behaviours, e.g., a slow crawling movement as caterpillar and flight behaviour as a moth (Duch and Levine, 2000; Libersat and Duch, 2002). Several studies investigated the neural mechanism that allowed the Manduca to change its behaviour so drastically.

It was found that both the anatomy and physiology of the involved neurons changed (Consoulas et al., 2000). An interplay between anatomical changes (i.e., changes in morphology) and physiological changes was observed. Nevertheless, “there is little known about how the physiological changes accompany structural remodeling” (Duch and Levine, 2000). Yet, as pointed out in Consoulas et al. (2000) “dendritic remodeling might also be important for modifications of the intrinsic properties of motorneurons”, implying that morphological changes affect the passive information processing due to changing cable properties of a dendrite (e.g., Mainen and Sejnowski (1996); Libersat and Duch (2002)) (for a review of the *cable theory*, see for example Mel (1994)). For this reason, we concentrate on the morphological changes and assume that electro physiological change accordingly[1].

---

[1] In the remainder of the manuscript, remodelling of the CNS is used as synonym for anatomical remodelling.

It was found that the CNS is remodelled by (i) cell death, (ii) cell growth, and (iii) reshaping of persistent neurons (Libersat and Duch, 2002). The latter is of importance for our study: how does the morphology of neurons affect the individual and ensemble behaviour? In case of the Manduca, it was found that the dendritic morphology of the persistent motor neurons undergoes complete transformation to go from slow firing neurons (as larva) to oscillatory neurons for flight behaviour.

Second, crickets (i.e., the *gryllus bimaculatus*) show strong behavioural adaptation during phonotactic behaviour. Cricket phonotaxis is the behaviour a cricket exhibits when tracking a specific sound coming from other crickets (e.g., Popov and Shuvalov (1977)). Two tasks are required for successful phonotaxis: (i) recognition, and (ii) localisation of the sound. Different studies from both biology (Simmons, 1988; Michelsen, 1998) and biorobotics (Webb and Scutt, 2000; Torben-Nielsen et al., 2005b) show that the cricket has a highly tuned auditory apparatus and dedicated auditory neurons (Schmitz et al., 1982; Thorson et al., 1982) to perform the two tasks successfully[2]. The highly tuned auditory apparatus has two main eardrums and transfers directional information (required for localisation) to the brain (Michelsen, 1998). The exceptional adaptivity manifests in the fact that even with one ear drum amputated or occluded the cricket still performs phonotactic behaviour with a reasonable performance (Schmitz et al., 1983; Huber et al., 1984; Schildberger et al., 1988). Experimental studies showed that after loss of (relevant) sensory input from one of the ear drums, specific auditory neurons change their morphology (Huber and Thorson, 1985; Schildberger et al., 1986; Schmitz et al., 1988). By this alteration in neuronal morphology, the cricket is enabled to perform phonotaxis reasonably well[3]. For our study, the cricket serves as an example of a living organism that remodels its CNS after injury, thus repairing itself.

Both examples show the power of changing neuronal morphologies with respect to behavioural change and damage recovery. It can be observed that the triggers for remodelling the CNS are different in both examples: the *Manduca* brain undergoes modelling as a result of a hormonal *trigger*, whilst the *Gryllus* brain remodels after a sensory trigger. For application in the robot domain, only the second trigger is of interest. However, regardless the trigger, the mechanism underlying changing neuronal morphology to achieve new behaviour is crucial. We aim at establishing a mapping between morphology and function.

3 Our approach

This section presents how the biological inspirations contribute to our approach to tackle the problem of self-repair. Our approach consists of three main research tracks. This paper presents the first track. For completeness, we start this section with an overview of the three tracks.

3.1 Three research tracks

We propose a methodology to tackle the problem of self-repair in mobile robotics which is inspired by the ability of insects to remodel their CNS to cope with changing environmental requirements. The long-term goal of our study is to port main principles underlying remodelling of the CNS in invertebrates to a robotic system with the ability to repair itself. The rationale here is that the neurons in the neuronal robot controller will alter their morphology to meet new environmental demands. With a brain as flexible as (some functions of) the invertebrate brain, a robot should overcome injuries to a certain extend. Generally, three research tracks need to be pursued to reach our goal: (i) constructing a neuronal model, (ii) automated remodelling, and (iii) embedding the results of the two previous tracks in a robot system.

First, a more realistic neuronal model is required. As elaborated in Section 2, the morphology plays a key role in information processing capabilities of neurons. However, current neuronal models as employed in Artificial Intelligence (AI) are considered shapeless computational units (e.g., McCulloch and Pitts (1943); Yao (1999)). Therefore, we need to construct artificial neurons that have analogous morphological features as neurons found in nature. In this track we investigate how to develop a suitable neuronal model.

Second, the artificial neuron as generated by the model from track one needs to be changeable in terms of its morphology. The process of remodelling has to be executed on-line and automatically. In this track we want to investigate how principles of neuronal plasticity (e.g., homeostasis or Hebb (Turrigiano and Nelson, 2000)) can regulate the remodelling process.

---

[2] For sake of completeness we have to say that the two tasks, recognition and localisation cannot be seen apart from each other. For a detailed explanation see Webb and Scutt (2000); Torben-Nielsen et al. (2005b).

[3] Changing morphology gives an explanation for long-term recovery of ear-injuries; instantaneous recovery is also observed but cannot be explained by changing neuronal morphology.

Finally, the mechanism underpinning neuronal information processing and neuronal plasticity investigated in the two other research tracks needs to be embedded in a robotic system. This track is the most tentative track. Nevertheless, we believe that the sensory trigger initiating remodelling in the cricket brain might give insights into the embedding of abstract methodologies into a body (e.g., robot) and how the interactions between control structure and body function.

3.2 Track one: neuronal model

First, we summarize two biological findings to present the rationale of the first track. Then, we present a technique we developed to generate artificial neurons with complex morphological properties and finally, we present results from the first research track.

Rationale

The rationale is based on two biological observations. First, neuronal morphology plays a crucial role in information processing. Second, the change in neuronal morphology plays a key role in the generation of new behaviours in some invertebrates. Indeed, a plausible morphology in combination with a passive processing model will allow us to mimic specific computations performed in biological neurons. I.e., delay lines as results of dendritic filtering (London and Häusser, 2005).

However, as pointed out before, neurons in AI are considered shapeless computational units. According to the abstract concept of artificial neurons, the computational power of a neuron results solely from the integrative capabilities of the cell body (Segev and London, 2000). In the more biologically accurate view, a neuron consists of a cell body, a dendritic field and an axon. consequently, the information processing capabilities also result from integrative properties of dendrites and axons. The integrative power of dendrites is explicitly shown in the compartmental models (Segev and London, 2000), and the cable theory concerning signal propagation (London and Häusser, 2005). We can conclude that *standard* AI neurons are not sufficient for work in our methodology. In order to be able to remodel neuronal morphologies and exploit the morphology-function mapping, we need biologically more realistic neurons. More realistic neuron models can be obtained from *virtual neurons*, digitised biological neurons that are used in neuroscience for subsequent modelling. In turn, virtual neurons are obtained by tracing, reconstruction or generation from scratch (Ascoli et al., 2001). We have developed a new technique for generating virtual neurons from scratch. We point out a major difference with virtual neurons as used in neuroscience. Our goal is to generate neurons possessing specific biological properties (i.e., morphology) without attention for biological accuracy.

Morphology generation

We use a mathematical formalism of rule rewriting, L-Systems named after its inventor Aristid Lindenmayer (Prusinkiewicz and Lindenmayer, 1990). L-Systems are used to algorithmically describe branched structures like plants or in our case, neurons. The idea is powerful yet simple. A set of axioms and a set of production rules is defined over a certain alphabet. Then, cyclic rewriting of the axioms takes place: each rule symbol in an axiom or rule is substituted by the contents of the according production rule. In this way, complex strings can be generated. Below an example of an L-System.

axiom:	F[X]
rules:	F → YF
	X → BX
1^{st} cycle	YF[BX]
2^{nd} cycle	YYF[BBX]

In this example, the L-System consists of the alphabet $\{B, F, X, Y\}$, one axiom and two rules. During the initialization phase, the L-System stores a string containing but the axiom. After the first cycle the string contains the substitution of the axiom symbols by the content of the evoked rules. In each cycle all symbols corresponding to a rule are now substituted by the content of each rule.

An L-System is nothing more than a way of generating large strings from an alphabet, i.e., a syntax. Without semantics L-Systems have no meaning. The semantics are defined by a geometric interpretation which translates the rewritten string to a graphical structure. We use the Rotation-Elevation interpretation to convert a string to a 3D structure. This scheme is analogous to polar coordinates and uses two parameters to define positions in three dimensions. A rotation angle defines the rotation on a plane, the elevation angle denotes the rotation on a second plane orthogonally intersecting the other plane (illustrated in Figure 1).

Results

We implemented the neuron generation technique. Figure 2 illustrates a 2D virtual neuron as generated

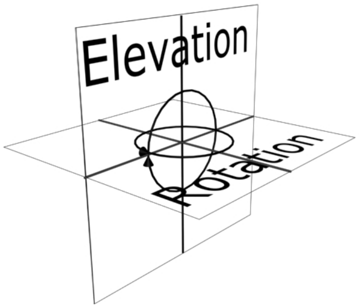


Figure 1: Rotation-Elevation interpretation of an L-System. A point in 3D space is defined by the rotation and the elevation on the plane orthogonally intersecting with the rotation plane. When generating 2D structures only the rotation angle is required.

by our system. The illustrated virtual neurons are all generated by the same L-System: usage of random parameters mimics the intrinsic uniqueness of real neurons. The different neuronal elements like axons and dendrites are clearly observable in the illustration.

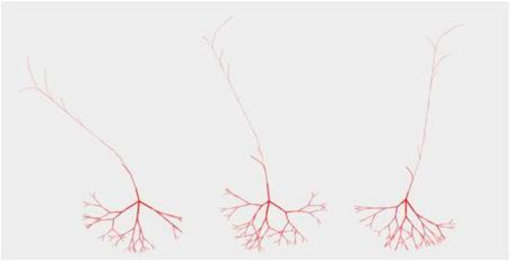


Figure 2: Two dimensional neuronal structure as generated by our system. Explanation in the text.

Furthermore, our system is able to generate neuronal structures in three dimensions as illustrated in Figure 3. The shape is stored in a Cartesian format as adopted by other neuronal morphology modellers (Cannon et al., 1998). The Cartesian storage allows the shape to be used in further experiments. We have to note that the neuronal structures illustrated in both figures are not biologically accurate structures (i.e., their parameters do not match with *fundamental parameters* of neuronal shape as defined by Hillman (1979)). For this work the biological accuracy is not crucial as long as we create complex neuron-like structures that in combination with passive cable properties allow complex non-linear computations.

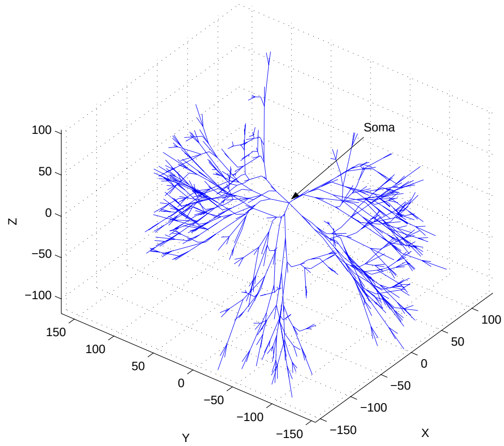


Figure 3: Three dimensional neuronal structure as generated by our system. Explanation in the text.

4 Conclusion

In this paper we have proposed a new three-fold methodology to tackle the problem of self-repair in mobile robotics, of which the first step has been completed. We introduced a methodology based on principles of neuronal remodelling in invertebrates. The approach is original in the fact that we are the first to propose to exploit morphological properties of neurons to achieve nature-like complex non-linear computations. For this reason we introduce non-standard artificial neurons. We introduced a technique using L-Systems to generate artificial neuronal structures. We showed empirically that we are able to generate both two and three dimensional neuronal structures. Currently, we only have morphometric descriptions of neurons. Functionality is not yet built in but dedicated neurophysiological simulators (e.g., Genesis (Bower and Beeman, 1998)) can simulate information processing for virtual neurons described in the SWC format that we use.

Current topic of research is the generation of biologically realistic and accurate virtual neurons. In general, virtual neurons are considered biologically accurate when there is no significant difference in the *fundamental parameters* of a specific biological neuron and the virtual counterpart. The fundamental parameters are a set of parameters describing neuronal shape defined by Hillman (1979); these parameters are still used as a reference for proving biological accuracy (Ascoli et al., 2001).

For future research, and keeping the next proposed research track in mind, it is useful to construct a virtual neuron generation method that interacts with the *substrate* in which the neuron is grown. We already built this functionality in in our system, but this function is currently limited to two dimensional neuronal structures (unpublished results). The interactions are based on principles of chemical attracting and repulsion (Feng et al., 2005).

Acknowledgements

The research reported here is part of the Interactive Collaborative Information Systems (ICIS) project, supported by the Dutch Ministry of Economic Affairs, grant nr: BSIK03024.

References

Giorgio A. Ascoli, Jeffrey L. Krichmar, Ruggero Scorcioni, Slawomir J. Nasuto, and Stephen L. Senft. Computer generation and quantitative morphometric analysis of virtual neurons. *Anat. Embryol.*, 204:283–301, 2001.

J.M. Bower and D. Beeman. *The book of GENESIS: exploring realistic neural models with the general neural simulation system*. TELOS, Santa Clara CA, 1998., 1998.

R.C. Cannon, D.A. Turner, G.K. Pyapali, and H.V. Wheal. An on-line archive of reconstructed hippocampal neurons. *J Neurosci Methods.*, 84(1-2): 49–54, 1998.

Christos Consoulas, Carsten Duch, Ronald J. Bayline, and Richard B. Levine. Behavioral transformations during metamorphosis: remodeling of neural and motor systems. *Brain research bulletin*, 53 (5):571–583, 2000.

Carsten Duch and R.B. Levine. Remodeling of membrane properties and dendritic architecture accompanies the postembryonic conversion of a slow into a fast motorneuron. *J. NeuroScience*, 20(18):6950–6961, 2000.

T. Elliott and N.R. Shadbolt. Growth and repair: instantiating a biologically inspired model of neuronal development on the khepera robot. *Robotics and autonomous systems*, 36:149–169, 2001.

Ning Feng, Gangmin Ning, and Xiaoxiang Zheng. A framework for simulating axon guidance. *Neurocomputing*, 68:70–84, 2005.

Dario Floreano and Joseba Urzelai. Evolution of plastic control networks. *Autonomous robots*, 11:311–317, 2001.

D.E. Hillman. *Neuronal shape parameters and substructures as a basis of neuronal form*, volume The neurosciences, Fourth Study Program. The MIT Press, Cambridge, MA, 1979.

Franz Huber and John Thorson. Cricket auditory communication. *Scientific American*, 253(6):46–54, 1985.

Franz Huber, H.-U. Kleindienst, Theo Weber, and John Thorson. Auditory behavior of the cricket iii. tracking of male calling song by surgically and developmentally one-eared females, and the curious role of the anterior tympanum. *J. Comp. Physiol.A.*, 155:725–738, 1984.

Phil Husbands, Tom Smith, Nick Jakobi, and Michael O'Shea. Better living through chemistry: evolving gasnets for robot contrl. *Connection Science*, 10(3-4):185–210, 1998.

John Koza. *Genetic programming: On the programming of computers by means of Natural Selection*. MIT Press, Cambridge, 1992.

F. Libersat and C. Duch. Morphometric analysis of dendritic remodeling in an identified motorneuron during postembryonic development. *J Comp Neurol*, 450:153–166, 2002.

Michael London and Michael Häusser. Dendritic computation. *Annu. Rev. Neurosci.*, 25:5003–532, 2005.

Zachary F. Mainen and Terrence J. Sejnowski. Influence of dendritic structure on firing pattern in model neocortical neurons. *Nature*, 382:363–366, 1996.

Maja J. Matarić. Behavior-based robotics as a tool for synthesis of artificial behavior and analysis of natural behavior. *Trends in cognitive sciences*, 2(3): 82–87, 1998.

W. McCulloch and W. Pitts. A logical calculus of the ideas immanent in nervous activity. *Bulletin of Mathematical Biophysics*, 7:115–133, 1943.

Barthlett W. Mel. Information processing in dendritic trees. *Neural Computation*, 6:1031–1085, 1994.

Alex Michelsen. The tuned cricket. *News Physiol. Sci.*, 13, 1998.

Francesco Mondada, Edoardo Franzi, and Paolo Ienne. Mobile robot miniaturisation: A tool for investigation in control algorithms. In *Experimental Robotics III*, pages 501–513, Kyoto, 1994. Springer-Verlag.

S. Nolfi. Evolving robots able to self-localize in the environment: The importance of viewing cognition as the result of processes occurring at different time scales. *Connection Science*, 14:3:231–244, 2002.

S. Nolfi and D. Parisi. Learning to adapt to changing environments in evolving neural networks. *Adaptive Behavior*, 5:75–98, 1997.

Esben H Østergaard, David J. Christensen, Peter eggenberger, tim Taylor, Peter Ottery, and Henrik H. Lund. Hydra: From cellular biology to shape-changing artefacts. In *International Conference on Artificial Neural Networks (ICANN2005), Warshaw, Poland*, 2005.

A.V. Popov and V.F. Shuvalov. Phonotactic behavior of crickets. *J. Comp. Physiol. A*, 119:111–126, 1977.

Przemyslaw Prusinkiewicz and Aristid Lindenmayer. *The algorithmic beauty of plants*. Springer-Verlag, 1990.

K. Schildberger, D.W. Wohlers, H.-U. Kleindienst, and F. Huber. Morphological and physiological changes in central auditory neurons following unilateral foreleg amputation in larval crickets. *J. Comp. Physiol. A*, 158:291–300, 1986.

Klaus Schildberger, J.J. Milde, and M Horner. The function of auditory neurons in cricket phonotaxis. ii modulation of auditory responses during locomotion. *J. Comp. Physiol. A*, 163:633–640, 1988.

Barbara Schmitz, Hans Scharstein, and Gernot Wendler. Phonotaxis in *Gryllus campestris L* (orthoptera, gryllidae) i. mechanism of acoustic orientation in intact female crickets. *J. Comp. Phys.*, 148:431–444, 1982.

Barbara Schmitz, Hans Scharstein, and Gernot Wendler. Phonotaxis in *Gryllus campestris L* (orthoptera, *Gryllidae*) ii. acoustic orientation of female crickets after occlusion of single sounds entrances. *J. Comp. Physiol. A.*, 152:257–264, 1983.

Barbara Schmitz, Hans-Ulrich Kleindienst, Klaus Schildberger, and Franz Huber. Acoustic orientaton in adult, female crickets (*Gryllus bimaculatus de Geer*). *J. Comp. Physiol. A*, 162:715–728, 1988.

Idan Segev and Michael London. Untangling dendrites with quantitative models. *Science*, 290:744–749, 2000.

L. Simmons. The calling song of the field cricket *Gryllus bimaculatus (de Geer)*: constraints on transmission and its role in intermale competition and female choice. *Anim. Beh.*, 36:380–394, 1988.

Tom Smith, Phil Husbands, Andy Philippides, and Michael O'Shea. Neuronal plasticity and temporal adaptivity: Gasnets robot control networks. *Adaptive Beahvior*, 10(3-4):161–184, 2002.

J. Thorson, T. Weber, and F Huber. Auditory behaviour of the cricket ii. simplicity of calling-song recognition in gryllus and anomalous phonotaxis at abnormal carrier frequencies. *J. Comp. Physiol. A*, 146:361–378, 1982.

Benjamin Torben-Nielsen, Guido de Croon, and Eric Postma. Timing is important: delaying task execution in plastic neural networks. In *Belgium-Dutch Artifical Intelligence Conference (BNAIC2005), Brussels, Belgium*, 2005a.

Benjamin Torben-Nielsen, Barbara Webb, and Richard Reeve. New ears for a robot cricket. In *International Conference on Artificial Neural Networks (ICANN2005), Warshaw, Poland*, 2005b.

Madeleine Tossit and Reinhard F. Stocker. Metomorphosis in drosophila and other insects: the fate of neurons throughout the stages. *Progress in Neurobiology*, 62:89–111, 2000.

Gina G. Turrigiano and Sacha B. Nelson. Hebb and homeostasis in neuronal plasticity. *Curr opin neurobiol*, 10:358–364, 2000.

Barbara Webb. Robots in invertebrate neuroscience. *Nature*, 417:137–158, 2002.

Barbara Webb and Thomas R. Consi, editors. *Biorobotics*. AAAI Press / MIT Press, 2001.

Barbara Webb and Tom Scutt. A simple latency-dependent spiking neuron model of cricket phonotaxis. *Biol. Cybern.*, 82:247–269, 2000.

Xin Yao. Evolving artificial neural networks. *Proceedings of the IEEE*, 87(9):1423–1447, 1999.

Nature-Inspired Systems for Parallel, Asynchronous and Decentralised Environments

4th April 2006

Organisers

Enda Ridge, University of York
Edward Curry, National University of Ireland
Daniel Kudenko, University of York
Dimitar Kazakov, University of York

Programme Committee

Contents

Parallelism vs Communication Overhead Trade-off in a JADE Multi-Agent Implementation of Cellular Automata

Hooman Amini[⋆]
[⋆]Department of Computer Science
University of York
ha502@york.ac.uk

Dimitar Kazakov[†]
[†]Department of Computer Science
University of York
Kazakov@cs.york.ac.uk

Enda Ridge[‡]
[‡]Department of Computer Science
University of York
ERidge@cs.york.ac.uk

Abstract

This paper investigates the implementation of a two-dimensional cellular automaton as a multi-agent system using the JADE agent middleware. The agents are distributed on separate computers in a local network, and their performance is measured to evaluate the net benefits of parallelising the algorithm, once the additional communication cost is taken into account. The results demonstrate that (1) the overall performance is a non-monotonic function of the number of agents employed; (2) a multi-agent platform may perform comparably to a single process implementation even for problems of relatively modest size; finally, (3), the trends observed in those cases indicate that the multi-agent platform may provide the best solution, but additional experiments would be needed for a definitive answer.

1 Introduction

This paper describes and studies the benefits of a multi-agent implementation of two-dimensional cellular automata using the JADE agent platform.

Multi-agent systems (MAS) can be used to decompose an algorithm and parallelise its execution. This implies a number of possible benefits. A multi-agent approach is usually characterised by the nature of information available to agents. Each agent only has access to local information, therefore, in the general case, it has no knowledge of the state of the system as a whole. If the desired system behaviour can be represented at agent level on the basis of this local information alone, then the use of agents is justified. The use of an agent-based platform is therefore linked to the possibility of decomposing the task at hand.

The suitability of an algorithm to be split into separate processes depends on two factors: (1) the amount of communication, and (2) the degree of indepence (or asynchronicity) between individual processes. The less communication is needed, the lower the computational overhead on top of the one required by each process. In addition, the less often one process needs to wait for another, the better.

Cellular automata (CA) are a modelling approach in which a system is represented as a lattice of finite state automata (FSA), where the next state of each FSA is only dependent on its own state and that of its neighbours. Therefore, as local information is sufficient to compute the next state of the system, CA are potentially suitable for a MAS implementation.

The basic idea is to partition the CA lattice and implement each partition as a separate agent. The only communication between these agents is limited to neighbours informing each other about the state of their immediately adjacent border areas (see Figure 1). The amount of information exchanged relative to the partition area depends on the shape of the partitions (e.g., for a square and rectangle with the same surface, the former will have a smaller circumference).

Despite the apparent asynchronicity of the communication involved, which is limited to messages between pairs of processes (neighbours) not using a global clock, a *de facto* global synchronisation among all processes emerges. This is due to the transitive nature of the information needed by each process, since a partition cannot be updated without information from all neighbours.

Here the benefits of employing a certain number of agents are a result of a trade-off. For a given CA lattice, allocating a part of it to a new agent increases the parallelism of the system, as this agent can run

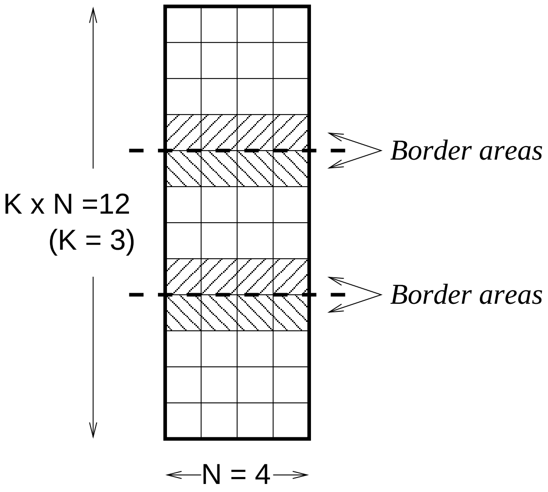


Figure 1: Partitioning of a CA.

on a separate computer/CPU, but creates additional communication overhead as new borders are created.

The aims of this work are to study the pros and cons of a multi-agent implementation of CA, and see if the additional benefits of parallelism can outweigh the additional communication overheads and result in a faster simulation. The factors studied here are (1) the number of agents/partitions employed, and (2) the size and shape of partitions.

2 Background

Cellular automata are dynamic systems consisting of a lattice of cells (in any number of dimensions) each of which has a number of associated discrete states. The state of these cells is updated at discrete time steps, the resultant state dependent upon local state rules Wolfram (1994). The specific set of rules is not important in our experiments. We have chosen a setup based on John Conway's well-known Game of Life, which uses a 2-D orthogonal lattice, and a cell's state rules only refer to the cell's 8 immediate neighbours.

The concept of an *agent* is a general modelling abstraction representing a system through its *behaviour*, mapping inputs from the environment onto system outputs. In this context, a single agent is virtually synonymous with the concept of a system Bertalanfi (1968). Using a multi-agent model of a system introduces the possibility of agent to agent interactions, and highlights issues, such as communication and coordination.

¿From the narrower software engineering perspective, agents are a “*design metaphor for . . . applications structured around autonomous, communicative elements*” involving software tools supporting the approach. At the same time, agents are “*a source of techniques and algorithms for dealing with interactions in dynamic and open environments*” Luck et al. (2002).

While there are no general purpose agent orientated languages, certain standards have emerged in the area. FIPA is one such standard for agent management and communication Dale (2005). Here we use JADE, a Java agent development framework that is compliant with the FIPA specifications Bellifemine et al. (2001). Of particular interest here are JADE's asynchronous Peer-to-Peer messaging and its scheduling of multiple agent behaviours.

3 Design

The experimental set-up consists of three types of agent. These agents and their functionality are as follows:

System Pilot Agent: is responsible for overall simulation coordination and lattice partitioning. Coordination includes, signalling the start of generations and distributing new partitions to Partition Simulator Agents.

Partition Simulator Agent: is in charge of each partition's simulation. It communicates with other Partition Simulator agents to negotiate and update border values. Partition Simulators subscribe to the System Pilot agent to receive simulation trigger commands. They report their results to the System Pilot at the end of each generation.

Simulation Inspector Agent: analyses the output and generates diagrams.

An experiment consists of the following five stages: (1) lattice initialisation, (2) partition creation, (3) lattice distribution, (4) partition simulation, and, (5) simulation termination. In stage 1, a lattice of a given size is created and its initial state set. In stage 2, a number of Partition Simulator Agents are created, each running on a separate, identical PCs in a local network. In stage 3, the each of the partition agents is initialised with the corresponding part of the lattice. Stage 4 is the central part of the simulation, in which each lattice partition is simulated on a separate agent. In this phase, all simulation agent interactions are pairwise (peer to peer), and there is no synchronisation (or global clock) provided by the Pilot Agent.

It is only at the end of the whole run that the partitions are assembled again to reconstruct the state of the complete lattice.

In all experiments, the lattice is of width N (where $N = 50$ or 100) and height $K \times N, K \in [1, 30]$. The lattice is divided into identical partitions as shown in Figure 1. The figure illustrates the case when K is the same as the number of partitions (the borders of which are marked with a dashed line), but this is generally not the case. This study describes experiments with 1, 3 and 5 partitions for K taking values in the above range. The dependent variable studied is the overall simulation time. All experiments were averaged over 3 runs for each set of parameters, and the CA was made to go through 10 consecutive states in all cases.

4 Results and Evaluation

The first set of experiments study the simulation time as a function of K for 2 different values of N (Figure 2). Despite the fact that the lattice surface (number of cells) is linear w.r.t. K, the simulation time trend is non-linear, and increases more rapidly, which is clear in the case of $N = 100$.

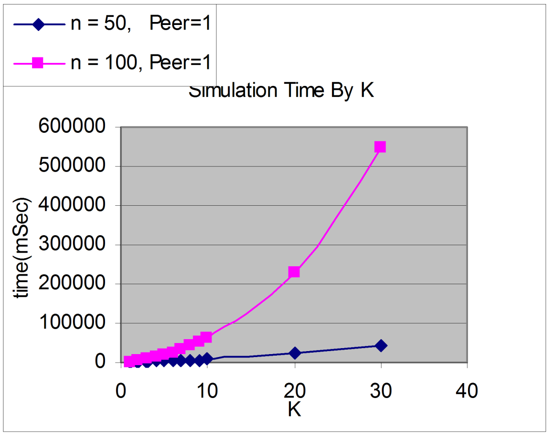


Figure 2: Simulation time as a function of lattice size.

Having established this trend, the next set of experiments studied the main research question addressed here, namely the net effect of the number of agents (partitions) on the overall simulation time (see Figure 3). The observed trends are revealing: simulating the whole CA as a single process, on the same PC, is fastest. The slowest runs are in the case of 3 agents: it appears that the benefits of decomposing and parallelising the task are outweighed by the additional overhead that communication between agents adds. However, this trend is clearly reversed when the number of agents is increased to 5. One is tempted to predict that a MAS CA with an even larger number of agents would outperform the single agent, but no more data is available at the moment.

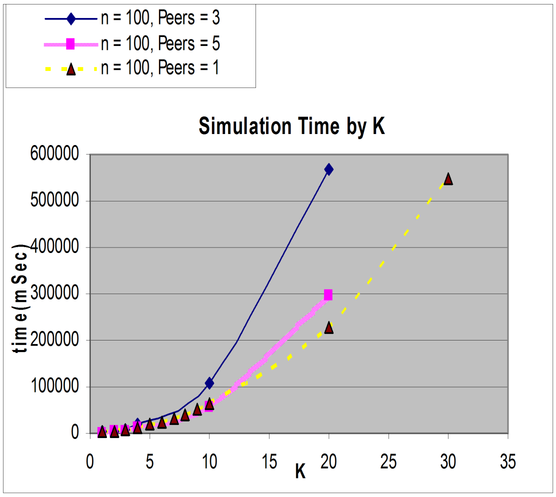


Figure 3: Agent simulation time by K for 1, 3 and 5 agents (peers).

The simulation times in general appear quite high, which was confirmed when we compared them to a simple object-orientated, non-agent, single thread Java implementation of the CA (data not reported here). Another interesting observation was that the time to create and initialise partitions was considerable (Figure 4), and in fact, comparable or longer than the simulation (which only run the CA for 10 cycles).

While considerable, this overhead is constant, as it does not depend on the number of CA cycles, and therefore, it does not influence the time complexity order of growth.

To have a rough comparison between agent-based and non-agent-based implementations of our cellular automaton, we ran the same simulation on a non-JADE non-MAS (single machine, single thread) platform (Figure 5). The non-JADE framework exploits the same model structure as the other one. The representation of the lattice cells is copied and the execution routine mirrored. Obviously, the issue of segmentation, and related issues of communication and synchronisation, are not present in the non-JADE algorithm. Whilst performance is significantly higher in the non-MAS scenario for CA size up to the hardware limitation of a single machine, extendibility and reusability issues are still more satisfactory on the MAS platform.

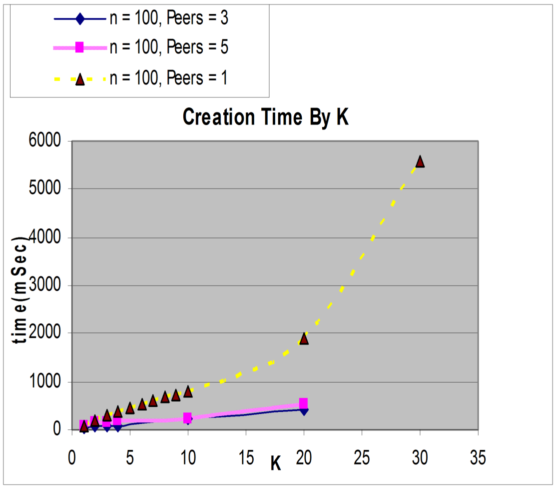


Figure 4: Creation time by K for different number of peers.

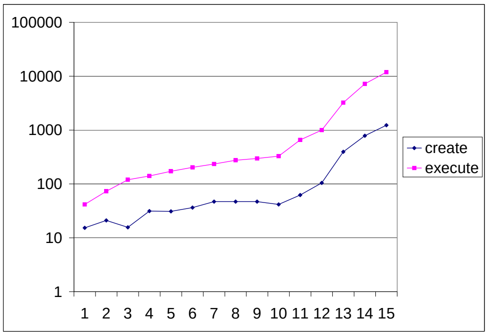


Figure 5: Creation and simulation times (log scale, in *ms*) for different values of K (X axis).

5 Conclusions

The conclusion to be drawn from the results is that the simulation time is a non-monotonic function of the number of agents employed. The observed trend suggests that a MAS CA gives a performance comparable to that of a single process despite the added communication overhead, even for relatively small lattices that do not impose memory problems. Furthermore, the trend observed clearly suggest that a larger number of agents may be able to outperform a single process/agent. It is also clear that using a MAS implementation and/or standard middleware (JADE) may promise scalability, but also comes at a price, as suggested by the creation time data and comparison with a non-agent implementation.

References

Fabio Bellifemine, Agostino Poggi, and Giovanni Rimassa. JADE: a FIPA2000 compliant agent development environment. In *Proceedings of the Fifth International Conference on Autonomous Agents*, pages 216–217. ACM Press, New York, NY, USA, 2001.

L. von Bertalanfi. *General System Theory: Foundations, Development, Applications*. George Brazier Inc., NY, 1968.

Jonathan Dale. The Foundation for Intelligent Physical Agents, 2005. URL `www.fipa.org/`.

M. Luck, P. McBurney, O. Shehory, and S. Willmott. Agent technology roadmap, 2002.

S. Wolfram. *Cellular Automata and Complexity*. Perseus Books Group, 1994.

An Integrated Approach Towards Researching and Designing Real-Time Brain-Like Computing Systems

Antonello Ceravola and Christian Goerick⋆
⋆Honda Research Institute Europe GmbH
Carl-Legien-Strasse 30
63073 Offenbach / Main
Germany
`Antonello.Ceravola@honda-ri.de`

Abstract

In this paper we discuss the design issues of brain-like real-time computing and describe an approach towards constructively creating such systems in an iterative fashion. The approach is based on the subdivision of systems into instance, functional architecture and computing architecture. We also shortly introduce the tools we have implemented for creating such systems. Finally we comment on some of our experiments, showing the feasibility of our approach and the level of scalability we could reach with it. The experiments comprise a biologically inspired active vision system as well as a biologically inspired system for computational auditory scene analysis.

1 Introduction

The aim of this work is to design a system for researching and implementing large scale biologically motivated systems in a real world environment on the functional level. The underlying assumptions are that major progress in brain-like computing can only be achieved if the systems aspect is a central point of research and that the real power of brain-like computing will only unfold if real world tasks are approached. The notion of a system comprises sensing, processing and the generation of behavior in the real world by means of actuators, which closes the loop to the sensors via the environment. We demand also a sufficient generality of the system with respect to the solution of several different tasks without a separate system-design for each task. Therefore, the spectrum of research topics ranges from biological models and processes via abstract system-theoretical aspects, simulation and real-time processing environments towards engineering of hard- and software computing elements. It is our strong belief that all those aspects have to be addressed concurrently in order to design a consistent overall system.

Since the knowledge about biological neural systems is limited, an iterative analysis by synthesis approach is pursued: Start the incremental design from the general to the specific, observe and analyze the behavior of the system and condensate the gained knowledge in an increasingly dedicated architecture. This includes the utilization of different means for constructing and examining the system. Once a critical mass of concepts, infrastructure and implementation is established, previously isolated concepts about some functional mechanisms in the brain can be tested in a comprehensive environment. In order to do so, all the tools and means must allow the researchers to focus more on the conceptual level of their work and less on the implementation level. A large influence on the design of the system comes from the area of real-time computing, since the system must interact with the outside world by means of sensors and actuators. Additionally, to enable the implementation of brain-like computing concepts, the system must provide interfaces to standard computer science algorithms and tools. We have decided to support this work with the standard software engineering process: design, develop and test. This concretely means that for every cycle we may redesign our systems, consolidating and making a more coherent design and implementation. The needs of such a process comes from the inherent complexity of cutting systems into pieces (modules).

2 Architectur Concepts

The design of the system is separated into three major parts: The computing architecture, the functional architecture and the instance of an application. The

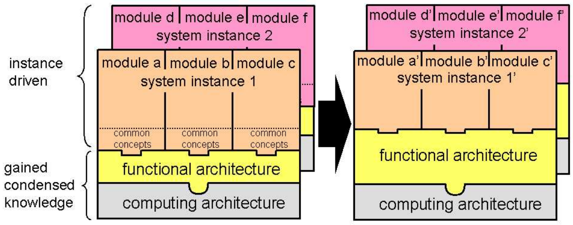


Figure 1: Subdivision into computing architecture, functional architecture and instance. The evolution of the architectures is mainly driven by the evaluation of different instances and modules. In the figure it is shown how some common concepts present in different modules are being transfered from the instance level to mainly the functional level. Consequently, the functional architecture grows from the step depicted on the left side to the step on the right side.

computing architecture comprises all software and hardware technology in order to provide a framework and an infrastructure for computing. Here the major design constraints and means are determined by technology and real-time computing, and, at this stage, by only few but fundamental principles from biology. A functional architecture represents the constraints of a hypothesis or model of the network of functional areas in the brain that makes different modules or components interact. Since the design of the functional architecture is not a ”simple” design but a major research field, it can only be approached in an iterative fashion. Within a functional architecture, several different instances of applications can be implemented (see also figure 1). The design decision to separate the system into computing and functional architecture was made in order to lead to a stable computing environment within which several functional architecture revisions can be evaluated. Any advances on the functional side may call for changes on the computing side, especially since the separation can never be perfect. But the idea is that changes on the computing side are on a slower time scale than on the functional side if the computing side is sufficiently general. The same applies to the relation of the functional architecture and instances. We target instance (applications) with the following characteristics: modular, data-driven control-loops, asynchronous communication, distributed and parallel (multi-thread and multi-machine). The basic assumption for each module is asynchronous computation.

3 Computing architecture

The computing architecture is the framework and the means for implementing functional architectures. In order to allow optimal flexibility for the different design dimensions on the functional side, the appropriate means of the computing side have to be provided. The computing architecture, that we refere to in this paper, is currently based on standard computers and operating systems. A certain degree of portability should be supported by a computing architecture in order to leave the possibility to test systems on platforms that provide support for special hardware or software libraries. We currenlty support: Linux, SunOS, Windows and VxWorks.

The computing architecture should provide the following services:

- Modularization support;
- System integration support;
- Simplifying the definition of parallel applications;
- Automatic data synchronization among modules/threads/processes;
- Ensure real-time performance (deterministic execution);
- Multi-platform support.

We have implemented such services by mainly creating our own component-based real-time middleware and runtime environment called RTBOS (Real-Time Brain Operating System).

4 Functional architecture

It is our opinion that *ex post* integration of independently designed modules will never lead to a well integrated system as a whole. Therefore, we propose inverse integration, i.e. a series of comprising functional architecture hypotheses, which guide the research and development of functional modules, has to be devised. There are several partial solutions to specific problems available in the scientific community, but the integration is either rather low or less coherent than necessary. According to our opinion it is crucial to aim for an integrated, incremental and convergent development effort to reach a critical mass necessary for tackling fundamental problems of computational intelligence.

The functional architecture should provide the following services:

- Image, sound and matrix manipulation libraries;
- Computing and data module definitions;
- On-line and off-line data inspection;
- Design phase support through scripting and graphical environment;

A general assumption on the functional side is that the computational resources are sufficiently large in order not to limit the research and the development of concepts. It is a responsibility of the computing architecture to provide such kind of resources. Nevertheless the research for functional algorithms should be guided by a set of principles which ensure that the fundamental design philosophy is met and the functionalities contribute to the goals. This comprises a proper scalability and computational complexity of the algorithms. We have implemented such services by creating libraries for image, sound and matrix manipulation, defining component models for data and computing modules, creating a graphical design tool as well as an integrated monitoring system.

5 Instance

If the described constraints in sections 3 and 4 are met, a wide range of applications can be created with the proposed infrastructure. There is still a full spectrum of conventions and decisions that have to be considered / estabilished in order to implement any instance application. But in this phase, the computing architecture and the functional architecture provide already a large support. At this stage, new solutions, new approaches and new methodologies should be implemented/investigated in order to achieve the desired functionality.

In the instance layer the following issues should be taken into account:

- Definition of a process for system design, creation, test and development (more in the biological meaning);
- Interface standardization;
- Architecture definitions at different levels;
- Module's repository organization;
- Learning (in all respects).

We have implemented such services by the definition of processes. Organizing our research and development phases. We have also created standard interfaces for our modules and data (see section 6).

6 Our experience and experiments

It is already more than four years that we are working on our integrated development system and we are using it for three years already. Most of the work has focused on ensuring modularity, scalability and generality. All parts of our integrated system can be used separately and can be modified with the minimum interaction with the other parts. One important characteristic is that we have created the concept of computing components (CM) and data components (DM) in order to encapsulate our algorithms and our data. Most of our specifications and tools are quite stable now (about 3 years old). With them we have reached a certain speed in developing and testing our systems, providing effective support. A detailed description of our system will be published in a further paper. Parallelization in our systems is firstly realized through OS-threads, secondly through machines. We do not use OS-processes within a machine since threads are lighter and less expensive for the CPUs of a machine. In order to give a glimpse on the implemented instance applications we shortly describe some of them.

One of our systems under development is a biologically motivated active vision system that autonomously chooses gaze direction and identifies known objects (attention can also be attracted by humans). The system can learn new objects from their visual features and associate them to a name through speech interaction with the users (partial description can be found in Goerick et al. (2005)).

This system is composed by two main subsystems: the gaze control loop and the online object recognition and learning. The gaze control subsystem performs the following tasks: image acquisition; attention control and motion control. The object recognition and learning sub-system perform object segmentation, feature extraction, object recognition, online learning and user interaction (with speech recognition and production).

The setup of the gaze control sub-system (composed by three loops) which runs in two machines, consist of: 98 threads , 128 instantiated CM components, 89 instantiated DM components. The image loop cycle takes 40msec, the attention loop cycle takes 250msec and the motion loop cycle takes 90msec.

For the online object recognition and learning subsystem (composed by two loops), which runs in one machine, the setup consist of: 25 threads, 74 instantiated CM components and 68 instantiated DM components. The cycle time for the image loop takes 40msec and the cycle time for the object recognition / learning cycle varies between 80msec and 320msec.

Another system we are currently working on, is a biologically motivated sound localization system. It is also composed by two sub-systems: the hardware interface and the sound localization computation. This system controls one of our ASIMO heads equipped with a stereo microphone set. This system is composed by the following layers: a sound acquisition, 3 streams for sound position computation (Interaural Temporal Difference, Interaural Intensity Difference, Interaural Envelop Difference) and an integration stage and head control.

The setup of the hardware interface sub-system is the following: 4 threads, 30 instantiated CM components, 35 instantiated DM components, and the stream cycle time is 50msec.

The setup of the sound localization computation sub-system is the following: 6 threads, 192 instantiated CM components, 243 instantiated DM components, and the stream cycle time is 50msec.

7 Related works

There is a spectrum of related work in the academic and commercial community where the focus is sometimes put on the modularity, communication or the real-time capabilities. Rarely all these aspects are integrated together in one single development system. We can clearly identify products that propose integrated solutions like ControlShell/Constellation RTI (2001) which targets clock based real-time systems. This development environment is less suitable for multithreaded systems where audio and video streams are the main part of the system instance communication. The work by Giorgio Metta and Natale (2005) proposes the definition of a system that makes usage of a library handling image and matrix manipulation. We also believe that this is an important feature in order to simplify the creation of intelligent systems. On the other side it only supports the definition of input/output communication through a library, leaving the rest of the work to the researchers. The work described in Lüders (2004) and Scheutz and Andronache (2004) proposes a modular approach through the usage of a ”component model” (packing patterns) in order to reduce complexity and improve integration. We strongly believe that this is a key feature. A more integrated system is proposed in the work done by Nesnas (2005), Schlegel (2004) and Vincenzo Ambriola and Ennas (2004). In these papers there are many different approaches being followed, mainly custom made to specific problems or to specific architectures. Our approach is to keep a general architecture in order to leave the largest freedom to the researchers.

8 Summary

In this paper we have summarized some aspects of what we consider important in the design of brain-like real-time computing. We have described the issues in constructing such systems, reporting on our experience and experiments.

Acknowledgments

Thanks to all our colleagues for the contributions.

References

Paul Fitzpatrick Giorgio Metta and Lorenzo Natale. YARP: Yet Another Robot Platform. *International Journal on Advanced Robotics Systems*, 2005.

Christian Goerick, Heiko Wersing, Inna Mikhailova, and Mark Dunn. Peripersonal space and object recognition for humanoids. In *Proceedings of the IEEE/RSJ International Conference on Humanoid Robots (Humanois 2005), Tsukuba, Japan*, 2005.

Frank Lüders. Adopting a software component model in real-time systems development. In *Proceedings of the 28th Annual NASA/IEEE Software Engineering Workshop*. IEEE Computer Society Press, February 2004.

Issa A.D. Nesnas. CLARAty: Towards Standardized Abstractions for Robotic Systems. *presented to workshop Principle and Practice of Software Development in Robotics, ICRA2005, Barcellona - Spain*, 2005.

RTI. *ControlShell User's Manual, Version 7.0.* Real-Time Innovation Inc., California, USA, 2001.

Matthias Scheutz and Virgil Andronache. Architectural Mechanisms for Dynamic Changes of Behavior Selections Strategies in Behavior-Based Systems. 2004.

Christian Schlegel. A Component Approach for Robotics Software: Communication Patterns in the OROCOS Contex. *presented to workshop Principle and Practice of Software Development in Robotics, ICRA2005, Barcellona - Spain*, 2004.

Diego Colombo Vincenzo Ambriola, Antonio Cisternino and Giorgio Ennas. Increasing decoupling in a framework for programming robots. *presented to workshop Principle and Practice of Software Development in Robotics, ICRA2005, Barcellona - Spain*, 2004.

Mammogram Mass Enhancement By Means Of Emergent Computation Using Simple, Reactive Multi-Agent Systems

Jeff Jones
University of Chester
Parkgate Road
Chester
United Kingdom
CH1 4BJ
jeff.jones@chester.ac.uk

Mohammed Saeed
University of Chester
Parkgate Road
Chester
United Kingdom
CH1 4BJ
m.saeed@chester.ac.uk

Abstract

We introduce methods of enhancing the contrast of digitised mammograms by means of emergent computation; specifically, the interactions between a population of simple, reactive agents and their habitat landscape comprising of a digitised mammogram. The behaviour of the agents stimulated by their environment generates emergent patterns that exhibit contrast enhancement. The application of selection pressures to the emerging patterns via parallel environmental dynamics results in further enhancement of the mammogram image, effectively highlighting the contrast between suspicious areas and the background. The system introduces a novel use of reactive multi-agent systems to generate emergent patterns and utilises parallel dynamics to further amplify the patterns produces. The system represents a visualisation aid for the reading and interpretation of mammogram images.

1. Introduction

Breast cancers are still one of the major causes of early fatalities in the world. The diseases, primarily affecting women, account for over 150,000 deaths annually according to the Worldwide Health Organisation. Successful treatment and likelihood of survival of breast cancers is strongly correlated with early detection and treatment. X-ray mammography is currently the most popular method of screening the target population due to its relative low cost, simplicity and wide availability. Mammogram screening has been shown to be effective in reducing the mortality of breast cancers (Anderson and Sigfusson, 1987). Analysis of mammographic images is based upon comparing images of both breasts, commonly in the mediolateral or craniocaudal planes (or both), and looking for disparities between both images in terms of asymmetry and distortion of breast architecture, development of masses and the formation of microcalcification clusters.

The analysis of mammogram images is hampered by the relatively poor contrast within breast tissue. Tumours may not significantly differ from surrounding tissue in terms of their attenuation of X-rays and so may differ only slightly in appearance from the surrounding tissue. Other features within the breast may also occlude any developing tumours such as ducts, blood vessels and fibrous connective tissue.

Analysis is made more difficult when the breast tissue is glandular in nature, presenting an image where is it very difficult to discriminate between details. Breast tissue that is mainly glandular is associated with a greater risk of tumour development (Gravelle et al, 1986) and is usually seen in younger women who are unlikely to be at an age to be considered to participate in mass screening programmes.

2. Previous Work

Many different approaches are possible to enhance mammogram images. Global approaches to contrast enhancement can remap the grey levels by histogram equalisation or stretch the signal range of the image. Both standard histogram equalisation (HE) and adaptive variations were reported to aid lesion visualisation in mammogram images (Gupta and Undrill, 1995) and (Stefanoyiannis et al, 2003). Spatial filtering approaches (Dhawan et al, 1986), fuzzy methods (Cheng, 2002) and wavelet based systems (Chang and Laine, 1999) have also been developed although, as noted in (Bovis and Singh, 2000), objective metrics of assessing contrast enhancement are often difficult to achieve.

Ramos has shown that simple reactive multi-agent systems, originally inspired by models of ant behaviour, can effectively perceive global image features (Ramos and Almeida, 2000). Liu has also used simple mobile agents to perform image feature extraction, noting the efficiency in the search for relevant image features (Liu et al, 1997). Zhuang has effectively used ant-colony systems for image processing functions such as edge detection (Zhuang and Mastorakis, 2005). The use of simple agent-based systems shows great promise as it encompasses the local sensory methods seen in the spatial filtering approaches whilst retaining the ability to perceive (collectively) global image features. The probabilistic behaviour of agent systems also lends them towards the processing of images with uncertain, incomplete or noisy features.

3. Multi-Agent Framework

PixieDust is a programming framework developed by ourselves that is used to perform experiments in image processing by simple, reactive multi-agent systems. An input image is interpreted as a three dimensional topographic landscape upon which a population of simple directional agents are located. The height of the landscape corresponds to pixel grey level intensity, zero (black) represents the lowest level and 255 (white) represents the highest level on the landscape. The agents can sense their local environment, typically a 3x3 window centred about their current location, and the agents can move about the landscape. The agents sense and act in a local manner and the population is de-centralised in terms of control. The framework is influenced by the stigmergy paradigm of indirect communication devised by Grasse to explain the complexity of termite building behaviour (see a summary in Bonabeau and Theraulaz, 1999), that has since been used as a model for communication in many different fields including some examples of human communication and robotics (Holland and Melhuish, 1999).

Agents receive stimuli from their local landscape configuration, influencing the agents' behaviour. The agents are able to modify the landscape in three different ways: The deposition of a trail as the agents move, the deposition of specific marks when an agent receives a stimulus that exceeds a particular threshold and the direct modification of landscape height by agents.

The interaction of the population with the landscape, over time, produces a complex emergent global pattern produced by the agents' simple local behaviours. The sensitivity of the agents to the landscape may be adjusted so that, for example, the agent is only sensitive to variations above a certain value. A maximum sensitivity may also be specified, so that the agents may effectively ignore stimuli that are too strong.

The landscape may also be subjected to parallel environmental effects such as erosion and diffusion. These effects can be applied to the trail patterns, specific marks or even the landscape data itself. The parallel environment effects can be used to amplify the emergent patterns formed by the agent behaviours by providing a selection pressure to the emergent patterns.

Agents inherit their basic behaviours from a prototype superclass and different agents may be created to serve different image processing functions. The algorithms controlling the agent behaviour have been made as simple as possible in order to ensure that any patterns formed are as a result of emergent behaviour and not as an artefact of the program complexity. Previous results using the PixieDust framework for the tasks of edge detection and impulse noise removal using different agent types have been presented in (Jones, Saeed and Lewis, 2003) and (Jones and Saeed, 2005) respectively.

4. Method and Results

The aim is to enhance the contrast between and the surrounding background tissue. As earlier stated, this is a difficult task as 'background' is difficult to define in terms of the mammogram. The background tissue may be of similar, or even higher, density than the target foreground. The strategy chosen for this emergent, multi-agent approach is as follows:

1. Select an agent with simple defined behaviours to generate the initial interactions with the landscape environment.
2. The interaction between the agent population and their environment results in emergent pattern formation.
3. These emergent patterns should discriminate (increase the contrast) between background tissue and contrast lesions.
4. Apply parallel environmental selection pressures to further amplify the contrast enhancement.

4.1 Selection of Initial Agent Behaviour.

A simple agent was developed that is sensitive to, and attracted towards, pixel intensity. Each agent samples its local 3x3 neighbourhood in the current direction it is facing (one of eight compass points). The agent considers the difference between the pixels at the front-left (FL) and the front-right (FR) of its neighbourhood. If the agent receives a sufficiently large stimulus from either direction, the agent then turns to face this direction and deposits a specific mark to indicate the location of the stimulus. The simple algorithm governing the agent behaviour is shown by the following pseudocode:

```
If FL > (FR+minT) AND FL < (FR+maxT )
        Turn 45 degrees to face FL
        LEAVE MARK at current location
ELSE IF FR > (FL+minT) AND FR < (FL+maxT)
        Turn 45 degrees to face FR
        LEAVE MARK at current location
ELSE
        Continue to face in the current direction.
```

The symbols *minT* and *maxT* are threshold values that specify the minimum difference in intensity and the maximum difference in intensity. The use of a minimum threshold helps to avoid the agent being diverted by minor fluctuations in the landscape, whilst the use of a maximum threshold allows the agent to ignore sudden very large fluctuations in the landscape, such as those presented by bands of connective tissue. These parameters may be adjusted to suit different tasks.

It is also possible to specify a probability value ***pCD*** that specifies the probability of an agent randomly changing direction, and ***pTP***, which specifies the probability of an agent randomly jumping ('teleporting') to a new, random location on the image landscape. Both of these parameters help to ensure even coverage of the image landscape with a relatively small number of agents and also serve to stop the agents becoming stuck in cycles of movement created by the configuration of the landscape.

4.2 Emergent Pattern Formation

The agents are initially placed on the landscape at random positions and each assigned a random direction. The behaviour of the agent, specified in the pseudocode, influences each agent's choice of direction and the agent moves a single step in the current direction before repeating the sensory stage again. Every time the agent moves forwards in the current direction, a small amount of trail is deposited by the agent. This is different from the specific mark described in the above pseudocode, which is only left when the agent changes direction in response to a significant stimulus. Unlike conventional image processing where the algorithm processes and modifies the actual image data, the PixieDust framework uses the input data to affect the behaviour of the agent population. Three resulting patterns of behaviour can therefore be seen, as shown in figure 1:

1. A pattern of agent distribution (1c) that is different from their initial random placement (1b) on the image landscape.
2. An emergent pattern of trail deposition (1d), created whenever the agent moves forwards.
3. An emergent pattern of mark deposition (1e), created in response to significant stimuli at certain image features.

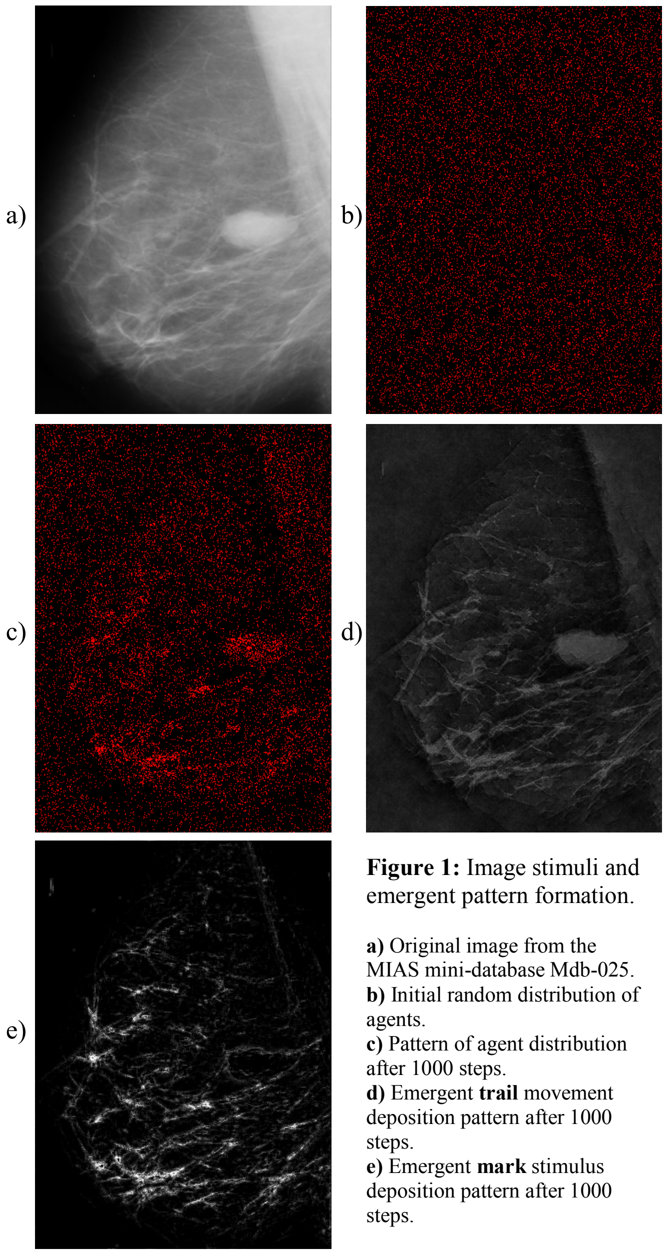


Figure 1: Image stimuli and emergent pattern formation.

a) Original image from the MIAS mini-database Mdb-025.
b) Initial random distribution of agents.
c) Pattern of agent distribution after 1000 steps.
d) Emergent **trail** movement deposition pattern after 1000 steps.
e) Emergent **mark** stimulus deposition pattern after 1000 steps.

The patterns are an emergent phenomenon produced by the interaction of the entire population with the image landscape. The agents adhere to the tenets of emergent behaviour, namely: locality, simplicity, de-centralised control and any emergent behaviour occurs at the population level. Individual agents can only perceive what is '*in front of their nose*', so to speak. At a population level, however, the entire pattern and features of the landscape are perceived. Chialvo and Millonas (1995) liken the emergent behaviour of agents in a swarm to the development of perceptual cognitive maps in the human brain.

4.3 Feature Discrimination and Contrast Enhancement in the Emergent Patterns.

The results shown in figure 1d and 1e show simple image enhancement taking place. Since the agents have a tendency to move (preference for) towards lighter areas in the image, the agents will aggregate in lighter areas of an image (1d). As there are more agents present within lighter areas, more trail will be deposited in those areas. Over time, the emergent pattern of trails will become correspondingly brighter in those areas, achieving a simple contrast enhancement effect. The emergent pattern of specific shown in 1e illustrates some basic feature detection, namely edge detection, within the image landscape. It can be noted that the lesion region in the image shows high mark levels at the border of the lesion, but low levels within the lesion. This is due to the comparatively even texture within the lesion region itself providing little stimuli for the agents.

4.4 Application of Selection Pressure.

The general tendency of a single agent's behaviour to move towards lighter areas of an image results in an emergent trail pattern that enhances the contrast between these lighter areas and the background tissues. As noted in the introduction, however, there are other areas of the mammogram, notably fibrous connective tissue, that are non-lesion areas (background). These areas are also light in appearance (i.e. high in pixel intensity). These areas are also enhanced during the emergent pattern formation, as shown in figure 1d. It is necessary to provide a mechanism that will enhance only the desired areas of the image, the lesion area itself. This moves the task from simple *image processing* towards image segmentation. Poli suggests that any image processing function can be thought of as a filter (Poli, 1995) since if we enhance one part of an image (foreground) from its background, we are effectively filtering out the background. If the removal of the background is absolute then segmentation has ultimately occurred.

The emergent pattern formation provides the enhancement of the lighter areas in the image. This can be considered as bringing the foreground (lesion and other light areas) *forwards*. A mechanism is required to move the background (non-lesion areas) *backwards*, to further enhance the contrast. The desired aim is also to place the bright structures that are not lesions in the background of the image.

In nature the process is called a *selection pressure*; an external pressure from the environment that ensures only certain features will persist over time. The maxim "Survival of the fittest" is a well known example of a selection pressure in nature where external environmental pressures such as food availability, reproduction opportunities and predation threats serve to ensure that only the *fittest* individuals persist over time.

The emergent trail and mark patterns can be considered as a distributed shared memory of agent behaviours. In particular, the trail patterns show a memory of frequency of visits of all areas of the landscape. It is possible to apply an external selection pressure in the form of a parallel erosion of trail deposits in the environment. By eroding the trail pattern, we effectively shorten the period of time for which the trail pattern has a memory of agent movement.

There will be a certain rate of trail erosion that is sufficient to 'dampen down' the trail level in the background areas of the image whilst (because of a higher concentration of agents in these locations) enabling the lesion areas to maintain their accretion of trail.

A number of different strategies for erosion of the trail patterns are possible. The simplest strategy is standard erosion, constant across all areas of the landscape. More complex possibilities are possible that mimic physical systems. For example, the evaporation of water on a physical landscape occurs more slowly in areas that are shaded from direct sunlight. Some of these more complex erosion strategies may be used for image enhancement. It is possible to erode the trails in direct, or inverse, proportion to the average pixel intensity in the 3x3 neighbourhood of each cell in the landscape.

Furthermore, it is also possible to adapt the *deposition* behaviour of the agents. At the simplest level the deposition is a standard amount configured at the start of each experiment. It is also possible to adjust the deposition to make it, for example, proportional to the average pixel intensity, or proportional to the standard deviation from the mean intensity in the agent's local neighbourhood window. Different deposition and erosion strategies may be applied to enhance areas in images that have differing features.

The results in terms of image contrast enhancement of applying different selection pressures to the system and of also modifying the deposition models can be seen in Figure 2 based upon the original mammogram image Mdb025 shown in figure 1a.

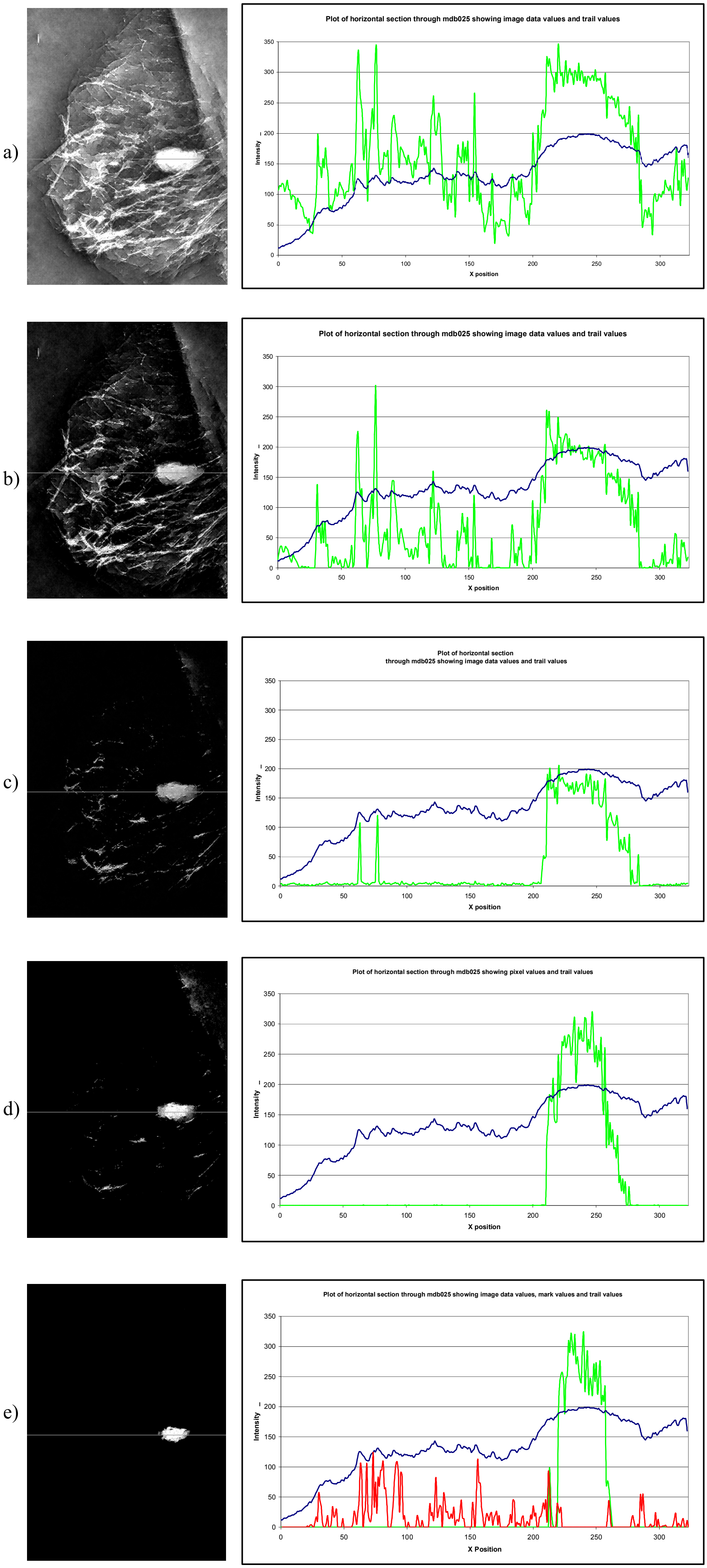


Figure 2: Selection pressure example images and cross section graphs.

a-e: The effects of adding a selection pressure in the form of trail erosion. At each time step trail is eroded in each cell. Original data is coloured blue, trail levels shown as green.

a) Trail pattern without any erosion applied. Some basic enhancement is already seen. The graph shows a cross section of data across the image at the point where y = 243 (indicated by faint horizontal line across lesion area). Blue line on graph shows pixel intensity level. Green line indicates trail height across the section.

b) Erosion is applied at a constant rate throughout image. Image (shown as trail pattern) is clearer but other bright areas are still highlighted as can be seen in the trail level.

c) Erosion applied in inverse proportion to the average pixel intensity in a 3x3 window around each cell. More background areas have been suppressed. In the cross-section, only the lesion area and strong curvilinear structures are visible in the trail level.

d) Erosion applied in inverse proportion to average pixel intensity as in ‘c’. Trail deposition for each agent is now proportional to the average pixel intensity of the 3x3 window of the current pixel at which the agent is currently located. Most background areas are now suppressed.

e) As in ‘d’, deposition of trails is in proportion to the average pixel intensity at the agent’s current neighbourhood. Erosion is applied as in ‘d’ but extra erosion is later applied that is in direct proportion to the local neighbourhood mark level (seen in red), as in figure **1e**. This removes the curvilinear lines corresponding to connective tissue of similar pixel intensity. Trail levels in the lesion area remain high because the mark levels in the same area are low, due to lack of stimulus by the relatively uniform lesion area.

4.5 Enhancing Microcalcifications

The enhancement of large masses is effective because they tend to contain an area within the centre of the mass that is relatively uniform in texture and intensity. This bright area will attract the light sensitive agents and the agents will deposit the trail at that location. The agents tend to stay within the mass area due to their ‘preference’ for the lighter parts of the image.

The eradication of curvilinear structures from the trail pattern is successful because those structures will tend to have a high mark level (due to their sudden changes in gradient) associated with them. By degrading the trail in direct proportion to the mark level, the thin bright edges will be degraded whereas the interior of the mass will remain relatively unchanged. The structure of microcalcifications, however, is almost the opposite to that of a larger mass. Microcalcifications consist of very small (as little as 0.1mm) particles of high density and are associated with sudden gradient changes within the image. Although the same light sensitive agents may be used to aggregate towards the microcalcifications, a different approach to the erosion and deposition of trail is necessary due to their different visual features.

Empirical results suggest that the most suitable deposition strategy to use in order to enhance microcalcifications is the laying down of trail in direct proportion to the standard deviation from the mean pixel intensity of each pixel neighbourhood. The erosion strategy should be simple erosion, i.e. the same amount eroded from every cell in the landscape.

Figures 3 and 4 illustrate some further results in mammographic mass enhancement and microcalcification enhancement using images taken from the MIAS mini mammographic database (Suckling et al, 1994). Some results using HE and linear contrast stretching methods are included for comparison.

5 Conclusions

We have presented a multi agent system that does not process images in the same way as conventional image processing operators, but instead uses the image to provide stimuli to a population of mobile agents for emergent pattern formation to take place. The emergent patterns formed are then amplified by the application of an external selection pressure. Unlike the global HE and linear contrast stretch, the system operates in a local fashion. This eliminates the ‘washing out’ of colours seen in HE in some images (e.g. Figure 3, k) and removes the need to blend together the separate windows used in adaptive HE techniques. Parameterisation of the system is constant for most mammogram images, however for images with very dense tissues, sensitivity of the agents needs to be increased to discern details in the very light areas. We are currently developing an automatic sensitivity system for each agent that determines sensitivity depending on the local neighbourhood contrast and intensity. The system is also emergent in the literal sense – image result patterns take time to emerge (in a manner similar to photographic development processes), approximately 30 seconds with selection pressure amplification. The computation time reduces fairly linearly with more powerful PC platforms. The computation may be stopped by either running the system for a fixed number of steps or halting the system when the output patterns reach a desired predefined contrast.

References

I. Anderson and B.F. Sigfusson. (1987) Screening for breast cancer in Malmo: A randomized trial, Recent results, *Cancer Research*, **105**, 62-66.

K.J. Bovis, and S. Singh. (2000) Enhancement Technique Evaluation Using Quantitative Measures on Digital Mammograms, Proc. 5th International Workshop on Digital Mammography, Toronto, Canada, M.J. Yaffe (ed.), Medical Physics Publishing, pp. 547-553, (11-14 June, 2000).

C.M. Chang and A. Laine. (1999) Coherence of multiscale features for enhancement of digital mammograms., *IEEE Transactions on Information Technology in Biomedicine,* **3:1**, 49-56.

H.D. Cheng and H. Xu. (2002) A novel fuzzy logic approach to mammogram contrast enhancement., *Information Sciences,* **148:** 167-184.

D.R. Chialvo and M.M. Millonas. (1995) How Swarms Build Cognitive Maps , *In: The NATO ASI series. Series F, Computer and system sciences*, 144, 439.

A.P. Dhawan, G. Buellton and R. Gordon. (1986) Enhancement of mammographic features by optimal adaptive neighbourhood image processing., *IEEE Transactions on Medical Imaging*, **M1-5:1**, 8-15.

I. Gravelle, J.C. Bulstrode, R.D. Bulbrook, D.Y. Wang, D. Allen, J.L. Hayward. (1986) *A prospective study of mammographic parenchymal patterns and risk of breast cancer*. Br. J Radiology, **58:**487-491

R. Gupta and P.E. Undrill. (1995) The use of texture analysis to delineate suspicious masses in mammography. *Physics, medicine and biology,* **(40):**835-855.

O. Holland and C. Melhuish. (1999) *Stigmergy, self-organization, and sorting in collective robotics.* Artificial Life, **5**(2): p. 173-202.

J. Jones, M. Saeed and S. Lewis. (2003) *Emergent Computation and Image Processing - Feature Extraction Patterns Generated by the Interactions Between Simple Agents and Their Image Environment*. in Proc of *MESM 2003*. 2003. Sharjah, UAE.

J. Jones and M. Saeed. (2005) Impulse Noise Detection and Removal From Images – An Emergent Multi-Agent Based Approach, *Proc of IEEE* 3^{rd} *Int Conf on Systems Signals and Devices SSD-05*, March 21-24, Sousse, Tunisia.

J.C. Liu, Y.Y. Tang and Y.C. Cao. (1997) *An evolutionary autonomous agents approach to image feature extraction.* IEEE Trans. on Evolutionary Computation, **1**(2): p. 141-158.

R. Poli., (1996), Genetic Programming for Image Analysis, in J.R. Koza., D.E. Goldberg, D.B. Fogel and R.L. Riolo. (Eds.), *Genetic Programming 96, Proc. of the 1 st Annual Conference*, Stanford Univ., MIT Press, pp. 363-368.

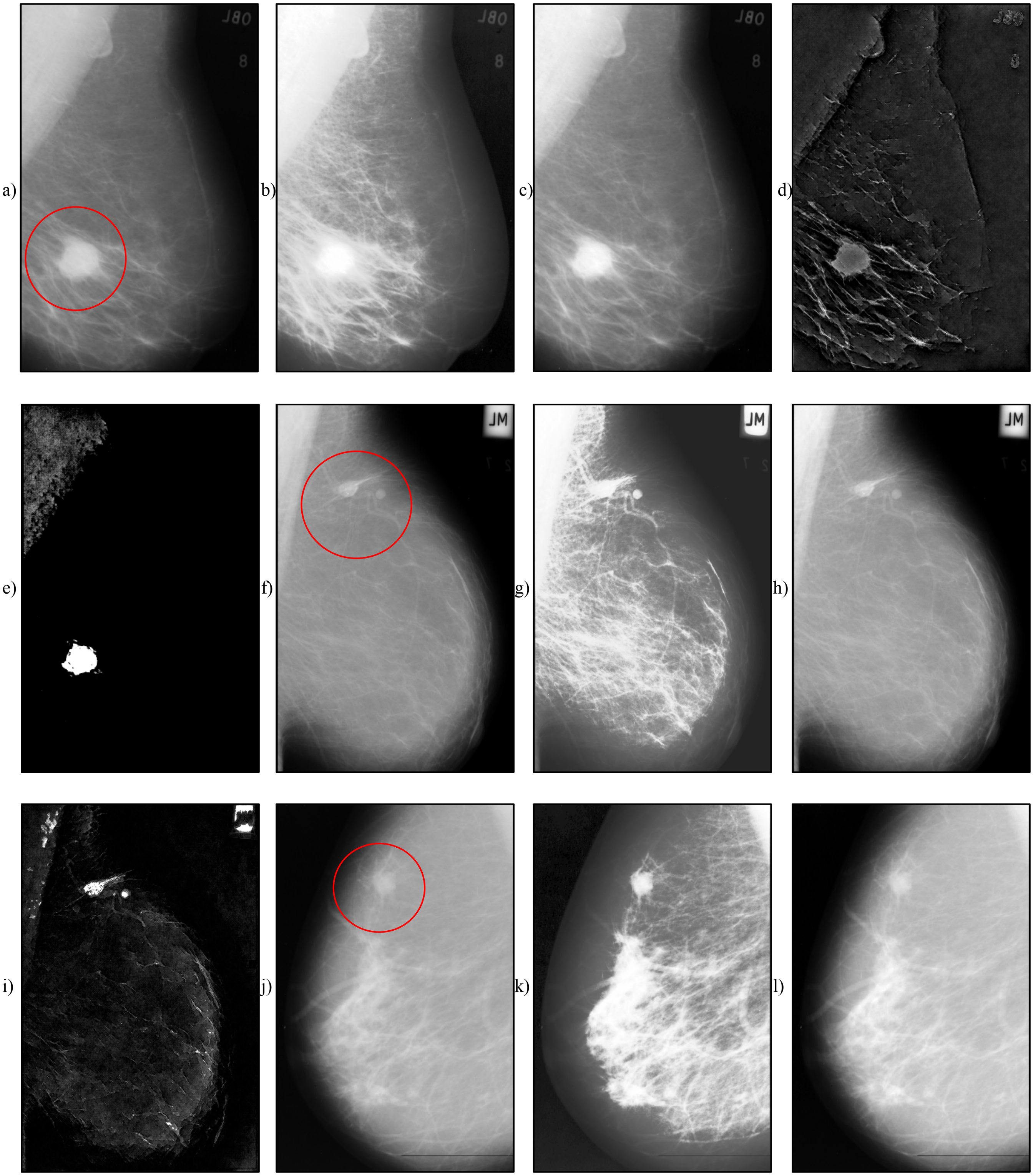
V. Ramos and F. Almeida. (2000) *Artificial ant colonies in digital image habitats - A mass behaviour effect study on pattern recognition.* Proc. of Ants 2000 2nd Int. Workshop on Ant Algorithms, 2000: p. 113-116.

A.P. Stefanoyiannis, L. Costaridou, S. Skiadopoulos and G. Panayiotakis. (2003) A digital equalisation technique improving visualisation of dense mammary gland and breast periphery in mammography, *European Journal of Radiology*, **45:** 139-149.

J. Suckling, J. Parker, D. Dance, S. Astley, I. Hutt, C. Boggis. I. Ricketts, E. Stamatakis, N. Cerneaz, S. Kok, P. Taylor, D. Betal and J. Savage. (1994) The mammographic images analysis society digital mammogram database. *Exerpta Medica. International Congress Series*, **1069:** 375-378.

G. Theraulaz and E. Bonabeau. (1999) *A brief history of stigmergy.* Artificial Life, **5**(2): p. 97-116.

X. Zhuang and N.E. Mastorakis. (2005) Image Processing with the Artificial Swarm Intelligence, *WSEAS Transactions on Computers*, **4:4**, 333-341.



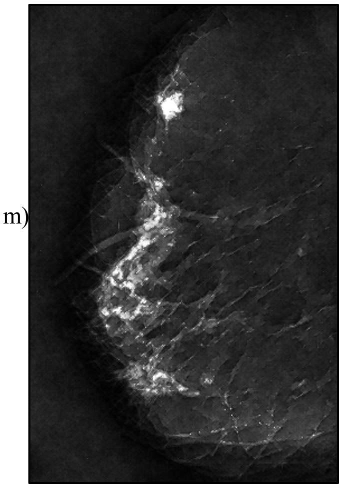


Figure 3: – Further Results: Tumour area highlighted with red circle

Platform: native x86 code from static compilation of Java bytecode. Intel Pentium mobile processor 1.8 GhZ, 1GB system memory.

From left to right:

a) Original MIAS image mdb028.
b) Histogram equalisation.
c) Histogram contrast stretch.
d) Emergent trail pattern with simple erosion.
e) Emergent trail pattern, erosion proportional to mark level and inversely proportional to pixel intensity.
f) MIAS image mdb132.
g) Histogram equalisation.
h) Histogram contrast stretch.
i) Emergent trail pattern, erosion inversely proportional to pixel intensity.
j) MIAS image mdb023.
k) Histogram equalisation.
l) Histogram contrast stretch.
m) Emergent trail pattern, erosion inversely proportional to pixel intensity and proportional to mark intensity.

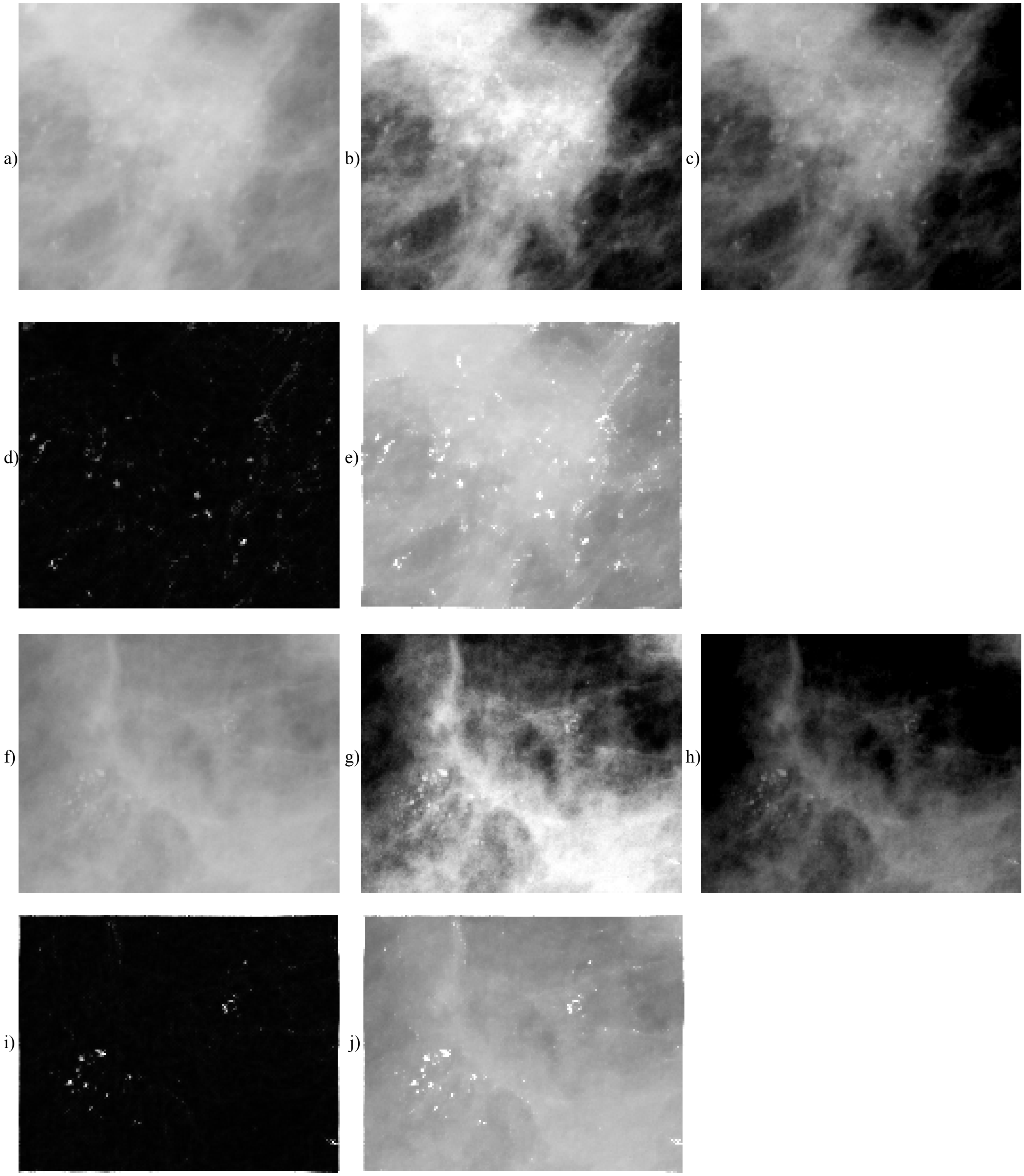


Figure 4: - Microcalcification enhancement results:

From left to right

a) Original region of MIAS mdb209 image.
b) Histogram equalisation.
c) Histogram contrast stretch.
d) Emergent trail pattern, deposition proportional to pixel standard deviation, simple erosion.
e) Original region of mdb209 image with emergent trail pattern overlaid.
f) Original region of MIAS mdb233 image.
g) Histogram equalisation.
h) Histogram contrast stretch.
i) Emergent trail pattern, deposition proportional to pixel standard deviation, simple erosion.
j) Original region of mdb233 image with emergent trail pattern overlaid.

Formalized methodology to treat Nature-Inspired Systems mathematically based on particle-dynamics physics laws

Carlos J. Martinez[†] [*]
[*] Dept. Electronic Engineering
Polytechnic University, Madrid 28046, Spain
jcmartinez@die.upm.es

Luis Estebanez[†]
[†] Hispatel Ing. SRL
Umbrete, 9 -Murcia 30009, Spain
luis@hispatel.es

Abstract

We present a methodology based in physics laws and particles in order to represent complex systems formally with an abstraction of particles, physics-laws, and collisions, treating information or signals as interacting "particles" and processes as locations or space-temporal "zones" of collisions. We introduce the particles and zones through a mathematical formalization and compare it with a well-accepted meta-model able to model many traditional computational systems. We show some cases where the "Particles" could model nature-inspired systems, with examples methods to model pheromones or swarms as particles moving in space-time, and review some advantages found.

1 Introduction

The constantly evolving computing and network paradigms are followed by new ways to represent, simulate and implement the complex systems.

We think decentralized algorithms can benefit from visual representations and models able to capture spatial coordinates for visualization. Our methodology formalizes the spatial coordinates integrating them in the mathematical representation. The complexity of modern designs could thus reuse existing visual tools while having a formal representation of nature-inspired concepts – for instance the "explored area" in mazes.

We introduce a "particle dynamics" abstraction to design and implement complex systems based on physics -interacting particles-. A mathematical relation to a well-known meta-model serves as a method to convert the particle, zones and rules concepts into classical computing models and their tools

The model presented here relies on the collisions or reactions among particles, and we aim at applying it for nature-inspired systems. It does not require a great depth of particle-physics knowledge, and the use of common physics like billiards balls is suitable to ease up design efforts.

Among the wide variety of methodologies for modelling systems, we find that basing our representation in nature ended up in being beneficial. As in Extreme Programming (XP) (Wells and Fuerts 2000) paradigms, where teams develop a common vision or "metaphor" of how a program works, we found that designing with a "particles" vision helped our models. The XP publications tell us how choosing simple system metaphors for developing systems give uniformity to the code, simplified algorithms and reduced processing time. We found the same result using a particle vision. The particle metaphor often pointed out parts that could be simplified by copying an analogous particle or behaviour of physics interactions.

As mentioned by XP authors before, we also found that metaphors - comparing real systems with nature - let us grasping the global picture quickly. In particle models we chose some evocative description of a system before modelling it, such as "this ant colony works like a gas mixture, where the ants are X ions and the pheromones electrons." An advantage found in some experiences was the possibility to bring nature side-effects back to our design and convert them in new features or easy improvements to the system under design. For instance, an important side-effect found was "a way to model stochastic processes", that will be explained later.

Our first experiences since late 2002 used particles to build a classical system: a thousand software IP-phones calling a telephony-switch under test, and several advantages were found and they led us to follow up building a mathematical formalization.

Now, we aim that the application of this methodology to the systems inspired in nature will provide new advantages. We provide possible examples in this paper after an intuitive definition of the model. We established some basic "particle" formal foundations that apply to classical computation and also to natured-inspired cases.

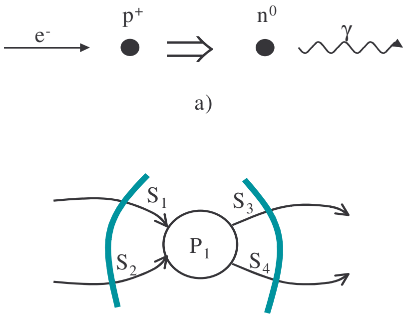
2 Intuitive description of the idea

Given the complexity of technology systems, we looked at the most elemental and smallest systems in nature looking for a modelling abstraction. That led us to choose a particle-physics metaphor as a reference to model complex behaviours.

We based computing models on very elemental parts of nature – the physics particles –. First we start with an intuitive analogy to continue the next section with a mathematical formalism to support this physic <-> computing analogy.

2.1 An elemental computing example

Let us see a particle inter-action example as in figure 1, and relate it to a traditional system. Consider the collision of an electron and a proton to produce a neutron and a photon in figure 1a.



a)

b)

Figure 1:

The interaction between two particles in this way can serve for modelling a process with two input signals or messages, and the particles resulting of the collision correspond to the output signals (S3 S4). We can see an analogy in how the signals S1 S2 in figure1b can be seen as “colliding” - like electrons and protons - in a process P1 “zone” that could just be seen as a collision point.

Besides the graphical similarities we present in next section a formal way to treat process and signals as particles and zones respectively. It allows a different view of networks too, where messages are the intelligent actors and routers colliding zones.

For representing computing systems each observed particle represents a signal - formally shown in next section-. When modelling networks each particle can model a message.

2.1.1 Interpretation for nature-based examples

In colonies, the proton can model an insect in a “looking state”, while the electron models the information coming from a pheromone’s evaporation. We see the collision as the pheromone (perception - the state change as in a state-machine). The resulting neutron is the insect in state “have directions”.

This analogy also serves as an example of side-effects as follows: In nature and physics the direction of the resulting neutron depends on the momentum, direction, and attraction of the electron. Thus, we could model our colony of “particles” taking advantage of this effect and letting the insect move directly by the physics collision/ attraction with the electrons. This could serve to implement the attraction towards a pheromone field.

When modelling systems we can choose the type of particles and its behaviour, and there is no need to use nuclear physics. Having found advantages in the particle physics paradigms we experimented further using bigger conceptual elements such as molecules in chemical reactions. The use of macroscopic objects of daily use seems a particularly easy way to represent systems within this new particles-view. For instance we could model ants with heavy atoms with extra electrons –i.e. ions charged electrically- moving in an electric field with a physical gradient equivalent to the food-pheromone field. We model food as protons that integrate into the atoms’ nucleus. Some beneficial side effects are found here: when the ant (atom) eats (get protons) its charge will equilibrate and the ant stop moving right away, since it does not feel the field. The designer did not need to program an extra behaviour or “stop moving” procedure – it was found as a natural effect.

An example in adaptable dynamic agents - able to reconfigure and react to the environment-: We can see S2 (proton) as the agent implementation and S1(electron) as a message. Here P1 is just a collision in the particle-view, where in other models might seem P1 is a static agent itself, and S3 (neutron) can represent a new-agent with an “adapted” behaviour.

When we have a collision zone consisting of sub-zones we can also map zones corresponding to several exploration areas of a maze. The modeller or designer can choose the modelling of zones and subzones that better fit.

2.1.2 Parallel, Asynchronous, Decentralized

Once the designer has specified a list of particles and basic interaction rules among them, a particle engine interpreter can run the system and treat multiple particles as totally parallel entities. A “particle compiler” could translate a particle model to a code able to work as a nature-inspired algorithm independently. Such tools are not the main goal of this work, and are left for future developments.

For decentralized systems, collisions allow the introduction of space coordinates; some of its advantages are having a model oriented to distributed systems and topology. We use the name *zone* to represent a process in computing but it represents a network or a node in communications or distributed paradigms. The particle view is not limited to multi agent system or active networks but is usable for many complex systems matching the mathematical formalism in next section.

For asynchronous communications we can have the examples of two molecules communicating by exchanging electrons or photons. A molecule sends an electron (message) towards a destination zone (containing a target particle). In a synchronous case no other collision will be expected besides a photon (return - produced remotely when the target collisions with the electron) coming back to the sender. In asynchronous cases, each collision can be seen as having a message/s out (the produced particle/s). We can have a generic model of Message-Oriented Middleware (MOM) with "semantic" particles that collide with "implementation-specific" ones in order to produce the "protocol message" particles.

Message Queues or centralized entities as "The collective" (Curry and Ridge, 2005) can use a composite of particles, such as a molecule, as information storage. The process of querying data is modelled as the collision of a photon (request) with the molecule (storage), the molecule lets a particle escape carrying the requested information.

2.1.3 Stochastic processes

Modellers familiar with quantum uncertainty can find useful to see that the result of two particles interaction has a "random" component or probabilistic outcome, since it can model stochastic processes.

Since we do not need physics knowledge, the analogy of billiard balls with just some "chance" to hit a certain place is a good metaphor for modelling random processes or create real Brownian paths.

We modelled IP phones using a set of heavy particles carrying phone implementations and light particles for messages. As massive particles "decay" into smaller ones within an average time <T>, it served us to generate a Poisson or non deterministic traffic for a mobile telecommunication network.

We propose to use similar concepts when modelling nature-inspired systems. In particular the mutations of a composite of particles can me modelled using a probability that a external particle (or just as time flows) change one of the particles and leaves the composite carrying a new information.

2.2 Advantages of the particle dynamics representation

This particle-representation could just be another way to model, simulate, or run systems, but we highlight here some benefits found: A First point of analysis of advantages or disadvantages is considering the ability to add spatial coordinates:

- Since the representation model associates zones, volumes, or space coordinates to processes or agents, it allows us to map the physical coordinates of a process. This can grasp the topology of distributed agents and decentralised algorithms, and let to model position-based pheromones. Designers could choose not to map coordinates to physical locations, or could for instance map z-axis to "protocol layer".

- The designer can identify zones producing particles as a special kind of source zone that is mapped to the interfaces with the external world. The input/ output of systems can be represented directly with sink / source *interface* zones.

- For instance, an entity that wishes to send an asynchronous message via JMS could just send one of its particles towards an interface containing a "JMS-coder" particle. The collision of the agent's particle (abstract message) with the "JMS-coder" produces a concrete and JMS-coded message that is send towards the target.

A second point of view is considering the model gives the importance of a system to the signals more than to the traditional processes. The application of this first postulate broke with the idea of intelligent processes exchanging simple signals, and left us with "intelligent" particles [like agents] being the real actors; they follow "nature rules" and collide in simple zones [processes]. Some advantages are:

- The representation methodology or model can ease and reduce the complexity of processes giving all the intelligence to the messages or agents. The routers/ processes are just simple areas where the incoming particles/ messages could collide

- If the particles are the main actors they have more power on the overall behaviour of collision zones, and therefore as there are no main processors, a system can potentially modify itself becoming reconfigurable and an adaptable system.

Other benefits and possibilities:

- We can use Particle-Dynamics applied in traditional computational models, such as synchronous data-flow or finite state-machines, if treat the particles as events and collision zones as the processes as described in the work of Lee & Sangiovanni (1998)

- It is applicable to many different levels of abstraction, from the definition of agents' network interfaces to a higher level definition of behaviours where the "zones" represent other abstract concepts such as virtual collaboration groups.

- For a large set of particles we can also apply physics formulas and concepts such as energy or entropy (Parunak and Brueckner 2001).

- The visual and common experience view of this Particle representation can be used with elements such as billiard or snooker balls, if they were closer to the designer's common experience.

The particle-dynamics approach applied to specifying and simulating systems can be managed by using a compiler or simulation design environment tool where we define a set of particles, collisions, and rules. Those are the basic elements to design and program with particles.

- A programming environment with well-known particles defined, and its rules already entered, can ease design tasks to designers with just basic physics laws as in macroscopic collisions.

- A designing environment could provide several flavours of system implementation (FIFO, LIFO, MOM, Asynchronous…) preconfigured, allowing the implementation to choose among them.

This compiler or simulation design environment tool is at a prototype level at present. This is a disadvantage that leaves the model and methodology in the theoretical arena only. Present particle specification tool, is plan to evolve to support the robustness of the theoretical formalism introduced

3 Formal foundations of the particle-dynamics methodology

A robust formalization of the Particle-Dynamics model allows mapping it with an accepted computing model below. The mapping gives a methodology to apply particles in a great number of system cases.

3.1 The model

We introduce formally here the "Particle" model in terms of observed particles, zones and rules. We use the sets T and V to store all possible tags and values respectively:

An "*Observed Particle*" p, or measurement is defined as a member of the T x V set, each p has a tag and a value, where the tag t allows modelling time, precedence relation-ships, but also other key properties like space coordinates in this model. We observe a set of coordinates (tags) and features (values v of V).Along a particle life history

A *Trajectory* or *Path* [signals] collects the space-time coordinates of a particle. The life trajectory or conversation s is defined as a subset of T x V, a set of particles, or equivalently a member of the power set $S = e^{TxV}$. A trajectory is the set of representations of a particle at different times, i.e. just the space-time information portion contained in a signal. The values in V can represent the operands and results of computation; we see V as multidimensional.

A behaviour is a tuple or vector s of N paths, is denoted as $b \in S^N = S \text{ x } S \text{ x}... \text{ x } S$. A process was defined in the reference frameworks as a subset of S^N for some N, i.e. a set of behaviour tuples s or b. We define a *zone z* to be the index of any process in the set B of processes P_1 P_2 ... P_M ... .

The *zones* are indexes, from 1 to M for instance, for the set of all possible behaviours. Among those we can have a unique zone representing the entire system, but it can also be split in a combination of smaller sub zones.

3.1.1 Validation for classical computing systems

A way to validate the Particle-Dynamics would be to demonstrate that it can be seen with analogous mathematics to the formal computing meta-model or reference framework of Lee & Sangiovanni (1998).

An *observed particle* is equivalent to what the reference framework defines as an "*event*". A trajectory corresponds to the $T(s) \in T$ - defined in the framework denoting the set of tags in a signal s. We can reuse the definition of *process* from the reference framework, calling them *zones*, therefore we can use zones for modelling any of the classical processes studied in the meta-model.

In more detail, let us consider the set of events in a generic way $E = T \text{ x } V = X_1 \text{x} X_2 \text{ x } ... X_D$, where X_i is a set of possible coordinates for i th dimension or x_i axis. This matches with the ordering and examples in the meta-model or framework, where some models -as ordered DE (Discrete Event)- are shown using a bi-dimensional w x w set for T and others suggest different types of values for V - seen as new dimensions in this particle model-. In our model V can carry information ranging from a database record value to energy or entropy level.

Lee & Sangiovanni (1998) demonstrated how their events – analogous to our particles – can model "states" in FSM (finite-state machines) and "places" in PN (Petri nets). We modelled Kahn process or dataflow networks with particles examples - left for future papers. To conclude, we introduced the model with some ideas for using it in nature-inspired uses.

References

Edward Lee and Alberto Sangiovanni. A Framework for Comparing Models of Computation, *IEEE Trans. on Computer Aided Design Of Integrated Circuits and Systems*, 17(12), 1998

D.Wells and W.Fuerst, Domain-oriented interface metaphors: designing Web interfaces for effective customer interaction, Proc of the 33rd Int'l Conference on System Sciences, 2000

Huaglory Tianfield, Formalized analysis of structural characteristics of large complex systems, *IEEE Trans. on Systems, Man and Cybernetics*, A 31(6):559 – 572, 2001

H.V.D. Parunak and S.A. Brueckner. Entropy and Self-Organization in Multi-Agent Sys-tems. *Procs of the 5th International Conference on Autonomous Agents. Montreal*, 2001.

Edward Curry and Enda Ridge, The Collective: A Common Information Service for Self-Managed Middleware, 4th workshop on Reflective and adaptive middleware systems 2005

Parallel Experience-based Ant Coloring

Gabriel NEGARĂ
Research Assistant, PhD Student
Faculty of Computer Science, Software Systems Department
"Al.I.Cuza" UNIVERSITY, IAŞI
ngabi@infoiasi.ro

Abstract

In this paper, we describe a parallel implementation of a new ant-like algorithm for the graph-coloring problem. *PEBAC* (Parallel Experience-based Ant Coloring) bases on an ant system meta-heuristic. It uses an ant system paradigm in which a number of agents (*ants*) perform specific colorings during a number of iterations. The colorings are performed using algorithms adapted to the ant system paradigm and are based on the experience from the previous iterations. *PEBAC* uses a new type of graph adjacency matrix and a new ant system approach for the graph-coloring problem. A set of transformations of the input graph provides the number of algorithm iterations. The agents perform specific graph colorings during iterations and update the system memory. To implement the ants work, we use a number of threads. The agents update the system memory and the graph structure changes appropriately. Algorithm ends when graph becomes a clique, the order of the clique representing the coloring number found by the algorithm. We tested the algorithm on graphs from the DIMACS Computational Challenge, obtaining good results.

1 Introduction

Cooperation between members of a colony of agents represents an attractive approach for solving NP-hard combinatorial problems. *Cooperative evolutionary heuristics* are iterative procedures that use a central memory. This memory collects information during the search process. An *iteration*, called also *generation*, is made of two complementary phases that modify the central memory. In the *cooperation phase*, a certain operation creates new offspring solutions, while in the *self-adaptation phase*, the new offspring solutions are modified individually. The output solutions of the self-adaptation phase are used for updating the content of the central memory. Termination of the search process may be triggered by reaching a predefined maximum number of iterations, by finding a satisfactory solution, or by satisfying any other stopping condition. The most successful evolutionary heuristics are hybrid algorithms in which a local search operator is used during the self-adaptation phase (Galinier, Hertz, 2003).

The graph-coloring problem is a difficult combinatorial optimization problem. While exact algorithms can solve instances with up to 100 vertices, heuristic methods are needed for larger instances. Recent graph coloring heuristics are either local search methods or hybrid algorithms that combine a local search with a population-based algorithm. It was observed that some graphs, especially large random graphs, can not be colored efficiently by using pure local search algorithms, and several approaches have therefore been proposed to deal with these difficult instances.

Therefore, there is much interest in implementing heuristic algorithms able to find feasible solutions in a reasonable number of algorithm executions - simulated annealing, ant systems, genetic algorithms etc (J.Shawe-Taylor, J.Zerovnik, 2001).

Ant systems represent a good alternative when trying to solve combinatorial optimization problems using cooperative, population based metaheuristics. *Dorigo and colleagues (Dorigo, Maniezzo, Colorni, 1991) proposed first the ant algorithms as a multi-agent approach to difficult combinatorial optimization problems* like the traveling salesman problem (TSP) and the quadratic assignment problem (QAP). Ant algorithms base on the observation of real ant colonies. Ants are social insects, that is, insects that live in colonies and whose behavior is directed more towards the survival of the colony as a whole than to that of a single individual component of the colony. An important and interesting behavior of ant colonies is their foraging behavior, and, in particular, how ants can find shortest paths between food sources and their nest. While walking from food sources to the nest and back, ants deposit on the ground a substance called pheromone, forming in this way a pheromone trail. Ants can smell pheromone and, when choosing their way, they tend to choose, randomly, paths marked by strong pheromone concentrations. The pheromone trail allows the ants to find their way back to the food source (or to the nest) and to find the shortest possible paths. That is, when more paths are available from the nest to a food source, a colony of ants may be able to exploit the pheromone trails left by the individual ants to discover the shortest path from the nest to the food source and back. Ants can perform this specific behavior of finding the shortest paths using a simple form of indirect communication mediated by pheromone laying, known as ***stigmergy*** (Grasse, 1959).

In *ACO* algorithms, a finite size colony of artificial ants with the above-described characteristics collectively searches for good quality solutions to the optimization problem under consideration. Each ant builds a solution, or a component of it, starting from an initial state selected according to some problem dependent criteria. While building its own solution, each ant collects information on the problem characteristics and on its own performance, and uses this information to modify the representation of the problem, as seen by the other ants. Ants can act concurrently and independently, showing a cooperative behavior. They do not use direct communication: the stigmergy paradigm governs the information exchange among the ants. High quality solutions are the emergent result of the global cooperation among all the agents of the colony concurrently building different solutions (Colorni, Dorigo, Maniezzo, 1992).

The goal of the presented algorithm is to find a coloring as good as possible for the input graph that is, trying to estimate the chromatic number of the graph. This problem is NP-hard. In the last years, a number of authors implemented ant algorithms for estimating the chromatic number, for example (D.Costa, A.Hertz, 1997). These implementations did not even outperform simple memory less coloring algorithms. PEBAC uses a new type of adjacency matrix, representing the system memory. Every agent (ant) performs specific colorings (based strictly on system memory), during a number of iterations. The algorithm is parallel, similar operations (ants' work) being divided among a number of threads. The goal of presented work is not to outperform the best coloring algorithms, but to present a new type of system memory, to present the strong cooperation between agents and to take advantage of parallel nature of the most part of the steps in the execution of *PEBAC*.

2 The graph coloring problem

2.1 Definition

The graph-coloring problem is the problem of finding, for a given graph G, an optimal coloring. If $G = (V,E)$ is a graph and k a positive integer, a k-coloring of the vertices of G is and assignment $c:V\rightarrow \{1, 2, .., k\}$ such that for each $i\in \{1, 2, .., k\}$, the set $c^{-1}(i) = \{v|v\in V, c(v) = i\}$ is a stable set in G, that is a set of non-neighbor vertices. This set is called the color class i of the coloring c. It is *NP*-complete to decide whether, for a given graph G and an integer k, exists a k-coloring of G (M.R.Garey, D.S.Johnson, 1979). The least possible number k of colors for which a graph G has a k-coloring is the chromatic number of G, denoted by $\chi (G)$. An optimal coloring means a coloring with $\chi (G)$ colors.

2.2 Approaches. Main idea of PEBAC

The decision problem of k-coloring is *NP*-hard for general graphs (M.R.Garey, D.S.Johnson, 1979), but also for special classes of graphs. As said before, the goal of *PEBAC* is to find colorings as good as possible, not to solve the k-coloring decision problem. In (J.Shawe-Taylor, J.Zerovnik, 2001), an ant algorithm for coloring a graph with a fixed number of colors is proposed. Based on approximate colorings obtained by single ants, edges of maximal evidences are added so that the new generation of ants does only attempt to construct colorings with additional restrictions. A colony of ants generates the solution. Each ant builds a feasible solution during the current iteration, i.e. it performs one run of a relatively fast algorithm that produces a near optimal solution. Initially, the ants do not have any knowledge of the problem they try to solve. At the end of the current iteration, the system memorizes the experience of every ant. The next iteration uses this information. The algorithm implemented for this approach uses a parameter similar to the temperature in the simulated annealing algorithm. If no k-coloring is found before a time limit, the algorithm stops. If the algorithm stops without finding a proper coloring, the best solution found in previous iterations represents an approximate solution provided by the algorithm.

The *PEBAC*'s approach for solving the graph coloring problem, instead, uses a pure experience based strategy, and a new type of adjacency matrix for implementing the global system's experience. It finds the coloring number using successive transformations of the initial graph, based on the values associated to the edges from the complement of the current graph (see 3.1., definitions).

These values are negative, floating numbers, representing the memory of the system, i.e. the ants' experience. If an agent identically colors two non-adjacent vertices in the graph, the value of the incident edge from the complement of the graph will decrease with a sub unitary value. This value is inverse proportional with the coloring number found by the agent in his current graph coloring. That is, a better coloring (less colors) means a more substantial decreasing of the values associated to every edge from the complement of the graph, if the incident vertices of the edge were identically colored.

The ant system based paradigm we use focuses on the concept of *S-memory* (system memory), describing the evolution of the system. S-memory is volatile and is updated based on the actions performed by the agents (*pheromone traces*). The pheromone traces memorize properties of solutions (number of colors, colors of vertices) and serve as decision criteria for any specific agent when performing an action. In a pure ant system algorithm, the pheromone trace memorizes (partial) solutions for the problem.

The global system's experience is managed using the system memory and is used to obtain new configurations. Any step performed by the agents during the execution depends on the system memory; *parallelism* and *experience* are the keywords of the algorithm.

PEBAC is an ant-like, parallel, approximate, probabilistic algorithm for coloring graphs, using features inspired from *ACO* metaheuristics. Specific coloring methods are used by a colony of agents (ants) performing, in parallel, graph colorings during a number of iterations. At the end of each iteration, the graph is modified by unifying a set of vertices pairs (each pair of vertices from the set is transformed into one vertex), based on system experience (as described in the next section), i.e. based on values from graph adjacency matrix.

A number of agents perform colorings during a number of iterations. Every agent (ant) shares the adjacency matrix *A*. At the end of every iteration, the matrix *A* is updated based on the colorings performed by the ants. In this manner, we implement the *pheromone traces* and the *vaporization*. After a number of iterations, the previous experience represents the most important factor when performing colorings specific to the current iteration. Any specific coloring method used by an agent in the current iteration depends on the current matrix values. To avoid *stagnation*, that is obtaining identical colorings for the most part of the agents at certain iteration, the specific colorings methods contain random based, stochastic built-in functionalities. More precisely, when an agent has many equally scored (based on matrix) local alternatives to continue, it uses a random strategy for selection the next move (in our case, the next vertex to color).

3 The PEBAC algorithm

3.1 Data structures. Parameters. Definitions

Define graph *G* by:

· $V = \{1, 2... n\}$ - vertices
· $E = \{(i, j) \mid i, j \in V, i$ adjacent with $j\}$ - edges

A new type of adjacency matrix holds the structure of the graph $G = (V, E)$:

$A = (a_{ij}), i, j \in \{1,2,...,n\}$ such that:
$a_{ij} = 1, (i, j) \in E$ and $a_{ij} \leq 0, (i, j) \notin E$;
initially, $a_{ij} = 0, \forall (i, j) \notin E$.

The matrix *A* is a new type of graph adjacency matrix. First, the matrix holds the graph structure. Second, the most important aspect regarding the matrix A is that it represents the modality to implement the *pheromone trace*. This is a concept specific to ant systems in the *PEBAC* algorithm. More details about how matrix *A* keeps track of previous experience of the agents during the execution of the algorithm - in the next subsection of the article.

Parameters of the algorithm:

· *NA*: number of agents (ants)
· *NT*: number of threads

Definitions:

· the *complement* of a graph *G* is a graph *H* on the same vertices such that two vertices of *H* are adjacent if and only if they are not adjacent in *G*
· a *clique* in an undirected graph G, is a set of vertices *V* such that for every two vertices in V, there exists an edge connecting the two.

3.2. Description

PEBAC is an algorithm derived from *EAC* (Croitoru, Negara, 2006), which implements successive contractions of the graph, using a set of edges from the complementary graph. The algorithm starts with the input graph as current graph to color. Agents perform colorings of current graph as in *EAC*, during a number of iterations. After performing the colorings from the current iteration, the graph is contracted using a set of edges from the complementary graph, selected using the values of the adjacency matrix (that is, based on global experience). The set of edges for contraction has the property that every two edges from the set have no common vertex. In order to select the maximum number of edges that keep this property, we perform an ascending order of the set of all edges from complementary graph. After that, we select, in a greedy manner, as many as possible, the first edges that have the property just specified. An alternative is to select a certain number of such edges, instead of taking as many edges as possible. Graph structure changes after every iteration, until we obtain a clique. The coloring number is equal to the order of this clique. The nature of operations performed successively on the graph ensures the fact that graph will become a clique.

3.2.1 *PEBAC* implementation scheme:

```
  NA/NT – number of agents/threads
  G – the initial graph
  A – the graph matrix

function ParallelContractingEAC()
{
```

```
    while (G is not a clique)
    {
        InitGlobalEdges(G);
        InitThreads(NT);
        foreach (thread T in Threads)
            T.start();
        JoinThreads();
        UpdateGlobalExperience(A);
        G' = ContractGraph(G);
        G = G';
    }   // end while
    colNumber = |G|;
    // (order of G)
} // end function
```

Let's explain now every line from the schema of *ParallelContractingEAC*() function.

The input graph *G* is a non-oriented, connected graph. An iteration from the *while* loop represents an iteration of the algorithm. Iterations end when graph becomes a clique, i.e. the coloring number provided by *PEBAC* is equal to the order of the clique (which is the last form of graph G, after the transformations during the number of iterations).

InitGlobalEdges(G):

The system memory is maintained and updated via the graph adjacency matrix *A*; negatives floating values are added on edges from the complement of *G* - a "more" negative value for $a_{i,j}$ (i non-adjacent with j in *G*) means that vertices *i* and *j* were colored with the same color in most of the previous iterations. *InitGlobalEdges(G)* determines all such edges *(i,j)* from the complement of the current graph *G* (having $a_{i,j} \leq 0$). Ants in their next colorings will modify only the values in the matrix *A* associated with these edges.

InitThreads(NT):

associates to every thread the function *ThreadAntWork()*, called when a thread starts. The 3.2.2 subsection presents this function.

Join Threads ():

simply performs a join of threads - a simple thread specific action

UpdateGlobalExperience(A)

it updates the system's experience, based on colorings performed by the ants in the current iteration and the previous iteration. After all the threads from the current iteration finish their job, the function code executes:

```
function updateGlobalExperience()
{   foreach (edge e in globalEdges)
        // i, j - incident vertices for e
        // ph - pheromone value on e
        if (u and v have the same color)
            a_{i,j} = ph
}
```

The corresponding matrix values receive the pheromone values associated with the edges from ***globalEdges***; in the next algorithm cycle, ***globalEdges*** resets based on the new graph structure, the system matrix keeps, instead, all the previous system experience. Given its simplicity, the computation needed for this function does not affect the system's performance. See also the 3.2.3 subsection.

ContractGraph(G):

selects a set C of edges from the complement of G, having the property:

$$\forall e_1,e_2 \in C, e_1=(i,j), e_2=(u,v), i,j,u,v \in V,$$
$$e_1 \notin E, e_2 \notin E, i \neq u, i \neq v \text{ and } j \neq u, j \neq v$$

The graph G contracts, by unifying the two incident vertices of every edge from C into a single vertex and performing a set of necessary matrix operations. These operations ensure a correct evolution of the global coloring algorithm. The new graph becomes the input graph in the next loop of while (the next iteration of the global algorithm).

Figure 1 illustrates the execution of the algorithm on a very simple graph. After the first iteration, the graph is colored with 3 colors: 1, 3, 6 - color 1; 2, 4 - color 2 and vertex 5 with color 3. The structure of the graph changes, after a number of vertices contractions. The figure shows the most important steps in the algorithm execution: colorings and graph transformations based on the system memory.

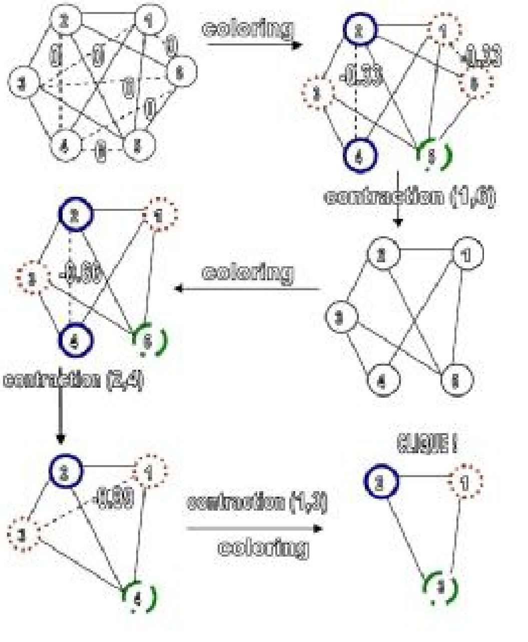


Figure 1. *PEBAC* execution for a trivial graph.

3.2.2 Specific colourings

When starting, every thread executes the code from the same function, *ThreadAntWork()*. A short description of this function:

```
function ThreadAntWork()
{
   Ants = NA/NT;
          // every thread handles the work
          // of the same number of agents
   foreach (ant a in Ants)
   {
      a randomly chose a coloring method *
      a colors the graph using the selected method
      foreach (edge in GlobalEdges)
         // see InitGlobalEdges (G) from 3.2.1
         add negative values on the edge,
         proportional with the coloring
         number found by current ant, a
   } // end foreach
} // end function
```

*A first scenario could be to use the same specific coloring algorithm for all ants. Another possibility, like in the following, is to use a number of specific coloring algorithms - for example *greedy*, *dsatur* (Brelaz, 1979) - our algorithm uses *experience based*, new versions of these algorithms; a number of ants (a part of total number *NA*) use the first coloring algorithm, another number of ants use the second algorithm and so on. The coloring algorithm used by an ant *a* in the current iteration will output a k_a-coloring of the graph.

This subsection presents, first, two from four *coloring specific methods (algorithms)* we used in the beta version of *EAC*; the final version of *EAC* uses other specific coloring algorithms, presented in the second part of this subsection. *PEBAC* uses a part of these specific coloring algorithms; using another combination of coloring methods can lead to different output for the algorithm.

An agent, in the current iteration of the global algorithm, can use any of the specific algorithms, in order to perform a coloring. The colorings depend on the previous *experience*. The first algorithm uses a *greedy-like technique* to establish an order of vertices, the second algorithm colors vertices based on an *ascending, degree based, order*; the third algorithm is a *dsatur adaptation*, that is, colors vertices in the descending order of saturation degrees. Finally, the fourth algorithm colors the vertices of the graph based on the matrix adjacency values.

```
Algorithm 1 (Greedy-based Coloring)
begin
   ant a picks the common matrix A
   nr_col ← 0 (the number of colors used)
  while (the coloring is not completed)
     ant a randomly choose a vertex v, not colored
     a finds the minimum color for v, colv
     color v with colv
     a choose the best vertex u for v (see * criteria)
     color u with colv
     if (col v = nr_col + 1)
        nr_ col ← colv
     endif
  end while
end
```

```
Algorithm 2 (Dsatur-based Coloring)
begin
    ant a picks the common matrix A
    nr_ col ← 0 (the number of colors used)
   while (the coloring is not completed)
            - ant a choose the first non colored vertex v
            from all vertices having maximum degree
            of saturation
            - a finds the minimum color for v, colv
            - color v with colv
            - a choose the best vertex u for v (see *)
            - color u with colv
            if (colv = nr col + 1)
                 nrcol ← colv
            endif
      end while
end
```

* Criteria - the ”best” vertex *u*, from all the vertices non-adjacent to *v*:

$$a_{vu} = \min\{ a_{vw} \mid \forall w \in V, w \neq v, \{v, w\} \notin E \}$$

The specific algorithms for the final version of EAC (used also for *PEBAC*):

```
ALGORITHM 1
(Experience based Greedy Coloring)
begin
    ant a picks the common matrix A
    foreach (vertex i in V ) (i)
            - let $S_1, S_2, ..., S_j$ be the current
            coloring sets (ii)
            - determine feasible coloring sets for i:
            $S_{i1}, S_{i2}, ..., S_{ik}$ (iii)
            if (there are no feasible sets for i)
                        - add i to a new set, $S_{j+1}$, as first
                        element of the set (iv)
            else
               foreach (set S in $S_{i1}, S_{i2}, ..., S_{ik}$)
                  - compute score for S: $\sum_t a_{it}$ ,
```

```
                for every $t$ in $S$
            endforeach
            - randomly choose $S'$ from
            best scored sets (v)
            - add $i$ to $S'$
        endif
    end foreach
end
```

(i): vertices are colored in order $1, 2... n$ or current vertex i is chosen randomly from the set of non-colored vertices

(ii): vertices $1, 2,..., i-1$ are already colored - S_t contains vertices colored with color t.

(iii): feasible coloring set for i - no vertex from this set is adjacent to i

(iv): i is colored with color $j + 1$, forming a new coloring set, S_{j+1}

(v): best scored sets - sets having the same, minimum value of the score

```
ALGORITHM 2
(Experience based Dsatur Coloring)
begin
    ant $a$ picks the common matrix $A$
    while (coloring is not completed) (i)
            - choose a non-colored vertex $i$ based on
            algorithm's criteria (ii)
            - let $S_1, S_2, ..., S_j$ be the current coloring sets
            - determine feasible coloring sets for $i$:
            $S_{i1}, S_{i2}, ..., S_{ik}$
            if (there are no feasible sets for $i$)
                - add $i$ to a new set, $S_{j+1}$,
                  as first element of the set
            else
                foreach (set $S$ in $S_{i1}, S_{i2}, ..., S_{ik}$)
                    compute score for $S$: $\sum_t a_{it}$ ,
                    for every $t$ in $S$
                endforeach
            - randomly choose $S'$ from best scored sets
            - add $i$ to $S'$
            endif
    end while
end
```

(i): Coloring not completed - there is at least one non-colored vertex left.

(ii): CRITERIA - from all vertices with maximum saturation degree, randomly choose a vertex from equally best-scored vertices set. The score of vertex i is $score_i = \sum_j a_{ij}$, for every j non-adjacent to i.

A best scored vertex means a vertex having the minimum score value.

3.2.3 System memory. Updating experience

As presented in 3.2.1, every ant adds pheromone on its path through graph, in other words, adds negative values on edges from the complement of G. For every such an edge, the two extremities were identically coloured by the ant. In this way, the co-operation between ants will determine the behaviour of algorithm in the next iterations. The advantage of the parallel nature of work performed by ants is obvious, the similar activities of ants are handled using threads. After all threads finish their job, an update of the global experience is performed, by joining all pheromone traces and modifying, appropriately, the global matrix A. The system memory is the key of the good functionality of the algorithm. It simply indicates the future behaviour of agents when generating their own partial solutions.

3.2.4 Algorithm convergence

In our previous implementation, *EAC*, the structure of the input graph does not change. Ants perform colorings of the same input graph. After a number of iterations, all the ants approximately find the same coloring number - *convergence*. In case of *PEBAC,* we see the repeated transformation of the graph as a specific type of algorithm *convergence*: the graph structure becomes simpler and simpler, after a number of iterations all the agents find the same coloring number for the current graph. The convergence depends, though, on the implementation of the pheromone trace, the algorithms' parameters and the agent/system level operations. In the current form of the *PEBAC* algorithm, the number of edges (from the complementary graph) selected for the graph contraction is proportional to the order of the current graph (the graph from the current iteration). This number of edges determines the quality of the convergence and the number of graph transformations to perform in the next iterations. By selecting a large number of edges for contraction, the graph structure changes significantly. In the next iterations, the colorings performed by ants will be almost identical, that is, the algorithm reaches a premature convergence. An advantage of this approach is the low iterations number. The algorithm converges, but the quality of solutions is, in many cases, poor. Limiting the number of edges for contraction, the number of graph transformations increases, the algorithm converges after a higher number of iterations. The quality of solutions increases because the group experience determines a high variety of partial solutions and a better management of the system experience.

4 Experimental results

PEBAC is a coloring algorithm emerged after a research period in which several hybrid coloring methods were tested, combining old known coloring methods like *greedy*, *dsatur* with ant-system-like metaheuristics. *PEBAC* is a parallel version of *cEAC* (contracting Experience Ant Coloring); both algorithms emerged from the *EAC* algorithm. The main reason of exploiting the parallel nature of the *cEAC* algorithm is the execution time. Using a parallel implementation, like in *PEBAC*, the algorithm execution time decreases, and the agents' cooperation becomes more obvious, like in the real life scenario.

We implemented the algorithm using Microsoft Visual Studio 2005 and the C# language, under the .NET platform.

For testing, we used input graphs from DIMACS Computational Challenge suite (DIMACS). Table 1 summarizes the experimental results obtained for PEBAC in comparison with *Dsatur* (Brelaz, 1979), and *ShortTabu* coloring algorithms (Glover, 1990). The table presents the results obtained only for a part of input graphs used for testing. For each coloring algorithm, the table lists the coloring number. Due to its probabilistic, approximate nature, PEBAC may provide different results in multiple runs. We obtained the coloring numbers after a number of tests, using the following values of the parameters:

NA = 1600 ants, NT = 200 threads

graph	*N*	*m*	χ	*#*	*EAC*	*DS*	*ST*
ana	138	986	11	11	11	11	11
david	87	406	11	12	11	11	11
DSJC 125.1	125	736	?	6	6	6	5
DSJC 125.5	125	3891	?	24	22	21	17
DSJC 250.5	250	15668	?	37	35	38	30
DSJC 250.9	250	27897	?	88	81	91	73
DSJC 500.9	500	224874	?	155	157	161	133
DSJR 500.5	500	58862	?	129	128	130	130
fpsol 2.i.1	496	11654	65	65	65	65	65
huck	74	301	11	11	11	11	11
le450 _5b	450	5734	5	13	12	9	5
le450 _5d	450	9757	5	7	7	11	5
queen 8_8	64	728	9	10	11	12	9
queen 8_12	96	1368	12	14	13	13	12
queen 10_10	100	2940	?	13	14	13	11
queen 12_12	144	5192	?	17	16	15	13
queen 13_13	169	6656	13	19	18	17	14
queen 15_15	225	10360	?	23	21	19	16
zeroin .i.2	211	3541	30	30	30	30	30

Table 1. Experimental results - comparison with other approaches for the graph coloring problem

Legend:

graph: the name of the input graph
N: the number of vertices in graph
m: the number of edges in graph
χ: the chromatic number of the graph
#: the coloring number for *PEBAC*
EAC: the coloring number for *cEAC*
DS: the coloring number for *Dsatur*
ST: the coloring number for *ShortTabu*

The results obtained are similar or better then the results obtained with heuristics like *Seq* (sequential heuristic) or *GA* (genetic algorithms), for the most part of the input graphs, and comparable with algorithms like *DSatur* or *ShortTabu*. The parameters' values decide the quality of solutions. Table 2 illustrates the influence of parameters in the *PEBAC* algorithm.

NT = 100 threads

		NA values (number of ants)				
graph	χ	100	400	800	1200	1600
ana	11	12	11	11	11	11
DSJC 125.1	?	10	9	8	8	7
DSJC 125.5	?	28	26	26	25	25
DSJC 250.9	?	92	93	92	90	89
queen 8_8	9	12	13	12	11	11
queen 8_12	12	16	16	15	15	14
queen 12_12	?	22	20	19	19	18
queen 13_13	13	23	21	22	22	20
queen 15_15	?	28	27	25	26	24

Table 2. The importance of parameters values for the quality of solutions. The table illustrates the results obtained for different numbers of agents, using the same number of threads. An increased number of ants cooperating in the global process produce solutions of higher quality.

Observations:

- for a fixed number of threads, increasing the number of agents does not assure in all the cases the improvement of the quality of the solution; some correlations exists between the two parameters that must be revealed by further experiments. The probabilistic nature of the algorithm implies, also, variations in the results obtained after a number of tests for the same input graph

- there are particular graphs for which the results obtained by *PEBAC* are partially unpredictable. After a number of tests, the motivation for these results seems to be the special internal structure of these graphs - "difficult graphs"

- the threads activity depends on the processor load; this can lead to partially unpredictable behaviors during the execution of the algorithm; the system's threads management influences the results obtained in many test cases

- adapting *PEBAC* for a multiprocessor platform can change radically the quality of the solutions by decreasing the execution time and having the possibility to use a huge number of agents, that is increasing the global cooperation.

A future version of *PEBAC* algorithm will adapt for executions on multiple machines in parallel. For the presented tests, we used a single machine, with single processor architecture. The threads' management is an important feature of *PEBAC*. The quality of solutions improves when the implementation appropriately exploits the parallel nature of the approach.

5 Conclusions

In this paper, a *new parallel ant-like algorithm* for the *graph-coloring problem* is proposed; it uses the *ant colony system* concept combined with experience based coloring techniques. A *colony of ants* collaborates for the goal of finding colorings and optimizing the quality of the solution. The most important factors are the *S-memory* (experience) and the *parallel nature* of the approach. The previous actions of the agents have an important role for the behavior of the agents in the *future* (the next iterations of the algorithm). *PEBAC* is a parallel ant-like algorithm that uses many *specific coloring* methods; the colorings are performed using *threads* (ants working in parallel), with the goal of *converging* to the final solution. *PEBAC* is an approximate, probabilistic algorithm, solving the coloring problem in polynomial time. The quality of the solutions depends on *parameters* value, due to the probabilistic nature of the algorithm. Based on the results we can say that *PEBAC*, a new ant system-based approach, represents a feasible approach for solving the graph-coloring problem.

For the future, the intention is to continue the work in the field of graph coloring and combinatorial optimization, searching for improved results and new ways to take advantage of the parallel nature of the cooperative heuristics for optimization problems.

References

[1] D.Brelaz. *New Methods to color vertices of a graph.* Communications of the ACM, 1979.

[2] A.Colorni, M.Dorigo, and V.Maniezzo. *Distributed optimization by ant colonies.* In Proceedings of the First European Conference on Artificial Life, Elsevier, 1992.

[3] D.Costa, A.Hertz, *Ants can colour graphs*, Journal of the Operational Research Society, 48 (1997) 295-305.

[4] C.Croitoru, G.Negara,
Experience-based Ant Coloring - A new ant-like graph-coloring algorithm, Scientific annals, "Al. I. Cuza" University, 15th serie, pages 110-123, 2006.

[5] M.Dorigo, V.Maniezzo, and A.Colorni. *Positive feedback as a search strategy.* Technical Report 91-016, Dipartimento di Elettronica, Politecnico di Milano, IT, 1991.

[6] P.Galinier, A.Hertz, N.Zufferey. *An Adaptive Memory Algorithm for the k-Coloring Problem*, Les Cahiers du GERAD, G200335, May 2003, Revised: April 2004.

[7] M.R.Garey, D.S.Johnson, *Computers and Intractability: a guide to the theory of NP-completeness*, 1979.

[8] F.Glover, Tabu Search - Part II, ORSA J. Comput. 2, 4–32, 1990.

[9] P.P.Grasse. *La reconstruction du nid et les coordinations interindividuelles chez bellicositermes natalensis et cubitermes sp. La theorie de la stigmergie: essai dinterpretation du comportement des termites constructeurs.* Insectes Sociaux, 6:4181, 1959.

[10] J.Shawe-Taylor, J.Zerovnik, *Ants and graph coloring*, ICANNGA 2001, 5th International Conference on artificial neural networks and genetic algorithms.

[11] DIMACS benchmark graphs. http://mat.gsia.cmu.edu/COLORING02/index.html.

Parallel, Asynchronous and Decentralised Ant Colony System

Enda Ridge⋆
⋆Department of Computer Science
University of York
ERidge@cs.york.ac.uk

Edward Curry†
†Department of Information Technology
National University of Ireland, Galway
EdCurry@acm.org

Daniel Kudenko⋆
Kudenko@cs.york.ac.uk

Dimitar Kazakov⋆
Kazakov@cs.york.ac.uk

Abstract

This paper describes a multi-agent system architecture that would permit implementing an established and successful nature-inspired algorithm, Ant Colony System (ACS), in a parallel, asynchronous and decentralised environment. We review ACS, highlighting the obstacles to its implementation in this sort of environment. It is suggested how these obstacles may be overcome using a pheromone infrastructure and some modifications to the original algorithm. The possibilities opened up by this implementation are discussed with reference to an elitist ant strategy. Some related exploratory work is reported.

1 Introduction and Motivation

It is often desirable to design large scale distributed applications in a parallel, asynchronous and decentralised (PAD) fashion. Furthermore, many emerging and proposed applications such as Grids and Peer-to-Peer systems will be constrained to operate in such environments. However, despite several notable successes (Brueckner, 2000), designing such systems still presents formidable challenges.

How do we enable decentralised control? How do we coordinate large numbers of agents? How do we communicate efficiently between so many agents? Natural systems of flocks and swarms provide evidence that these questions can be addressed. However, with some exceptions, the majority of established and successful applications based on nature are sequential, synchronous and centralised (e.g. the genetic algorithm, ant colony optimisation). Can we harness the power of these established algorithms in PAD environments and so help address these challenges?

This paper proposes an architecture for implementing a well-known and successful nature-inspired algorithm, Ant Colony System (ACS) in a parallel, asynchronous and decentralized environment.

The next section reviews the ACS algorithm for completeness. Section 3 describes how to implement this algorithm in a parallel, asynchronous and decentralised environment. Section 5 reviews related work. Section 6 concludes with a summary of some ongoing exploratory research and an outline of future work.

2 ACS Algorithm

The original ACS algorithm was applied to the Travelling Salesperson Problem (TSP). The TSP involves finding the shortest route between a set of cities such that no city is visited more than once. The TSP is often represented as a graph structure (Figure 1).

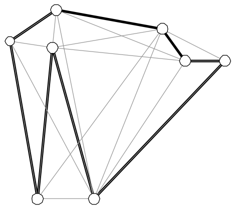


Figure 1: Illustrative TSP graph (some edges omitted for clarity). Darker edges illustrate a tour.

The Ant Colony System (ACS) algorithm is summarized in Figure 2. The ‘pheromone’ applied to edges is an abstraction of the chemical markers used by real ants. Edges with high pheromone levels are more attractive to ants. All ants build their tours using a probabilistic decision rule., The *local pheromone update* involves decaying the pheromone level on an edge traversed by an ant by a small amount. Once all ants have built a tour, pheromone is deposited along the best ant’s tour in a *global pheromone update*. The whole process then repeats.

```
Initialise pheromone on all edges
While (stopping criterion is not yet met)                     [Main Loop]
    Place each ant on a random city such that
    no two ants are placed on the same city
    For (each ant)                                            [Tour Building & local update]
        Choose the next city to visit according
        to a probabilistic decision rule
        Apply a local pheromone update to the
        chosed edge
    End For
    For (each ant)                                            [Update best tour]
        Compute the length of tour found
        If an improved tour was found, update
        the record of the best tour and its
        associated cost
    End For
    For (each edge on the best tour found)                    [Global update]
        Perform a global pheromone update by
        depositing pheromone on the edge
    End For
End While
```

Figure 2: Pseudocode for ACS

2.1 Issues

This paper addresses the two major obstacles to implementing this algorithm in a parallel, asynchronous and decentralised fashion.

- Firstly, the algorithm is highly synchronous. Each ant performs its own tour and local pheromone update in turn [1]. The algorithm waits for all ants to finish this phase before a global pheromone update is performed. Once the global pheromone update is complete, all ants begin the tour building phase again.
- Secondly, the global pheromone update requires a *centralised* comparison of all the ant tours to acquire the *global knowledge* of the system's best tour.

Clearly, a direct implementation of this algorithm would be expensive and inefficient in a parallel, asynchronous and decentralised environment. The next section proposes how these obstacles can be overcome.

3 Parallel, Asynchronous, Decentralised ACS

We propose a Multi-Agent System (MAS) platform as the basic framework on which the algorithm should be implemented. Common MAS platforms (Bellifemine et al., 2001) provide a convenient means of distributing computation over machines while coordinating with asynchronous messaging.

3.1 Pheromone Infrastructure for ACS

Brueckner first introduced the concept of a pheromone infrastructure in the context of manufacturing control (Brueckner, 2000). Briefly, this approach involves representing the environment by a topology of *Place agents*. These Place agents manage 4 pheromone functions: aggregation, evaporation, propagation and sensing (Parunak et al., 2004). Such an infrastructure can be used to methodically move the ACO algorithm into a PAD environment (Ridge et al., 2005).

Each city in the TSP (Figure 1) is represented by a Place agent. Each ant from ACO becomes a *Solution agent* that interacts with this pheromone infrastructure. Figure 3 is a schematic of our architecture overlaying the previous TSP graph of Figure 1.

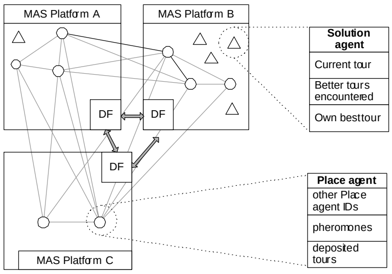


Figure 3: Architecture showing 3 MAS platforms. Nodes in the previous TSP graph are now Place agents distributed on the platforms. Some platforms are also hosting Solution agents.

3.2 Agent Interactions

In this scenario, all Place agents register with one another using a service description containing their city's coordinates. Place agents can then calculate the distance to any other Place agent. Place agents maintain a record of the pheromone levels on each link connecting to other Place agents.

---

[1] We have seen conflicting pseudocodes in the literature. Some ACS implementations construct solutions in parallel and not one after the other as seen here. We are currently investigating whether these differences have an effect on performance.

The Solution agent lifecycle consists of the following interactions with the pheromone infrastructure (Figure 4).

1. A Solution agent 'arrives' at a given Place agent with an INFORM message containing details of where the Solution agent has come from.
2. The Place agent performs a local pheromone decay on the relevant link.
3. The Place agent responds with (1) a list of other Place agent IDs (acquired from the Place agent registrations), (2) the calculated cost to each of the other Place agents and (3) the latest pheromone level on the links to each of the other Place agents.
4. The Solution agent uses this information to decide which Place agent to visit next.
5. The Solution agent informs the Place agent of its decision to move to a given destination Place agent.
6. The Place agent updates its pheromone value for that link using the equivalent of the algorithm's *local pheromone update*.
7. Life cycle returns to step 1, with the Solution agent arriving at the destination Place agent

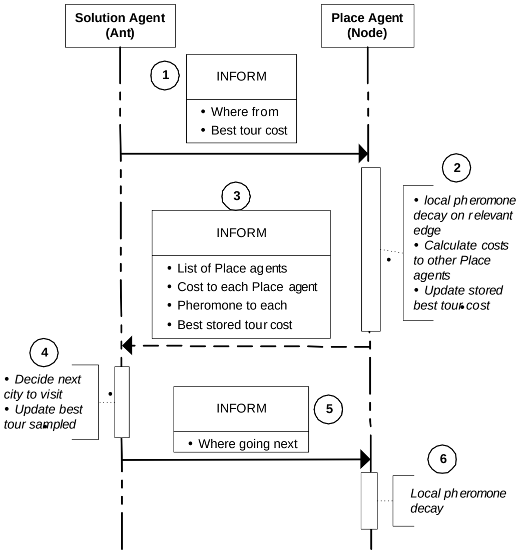


Figure 4: Solution agent interactions

3.3 Global Pheromone Update

Global pheromone updates are performed by Solution agents as follows. When a solution agent moves between Place agents (Step 4), it keeps track of the cost associated with that move and the Place agents involved in that move. Once it has visited all Place agents, it has a total cost for the tour it has completed. The Solution agent can then deposit the associated amount of pheromone with the Place agents and the Place agents can adjust their pheromone levels accordingly.

Thus far, we have described the mechanisms and interactions that permit a *direct* implementation of ACS on a pheromone infrastructure. Recall the two main issues identified previously: that of the *synchronous* tour building and global update phases and that of a global update that depends on the *centralised* global information of the best tour produced. We now address these issues with suggested modifications to the architecture and algorithm.

3.4 Modifications

Let each Place agent be capable of storing a description of a tour and that tour's cost. When Solution agents 'move' between Place agents (Steps 1 to 3), they deposit with the destination Place agent the cost of the best tour the Solution agent has ever performed. If the Place agent's current stored cost is worse (higher cost) than the deposited cost, the Place agent overwrites its stored cost. The Place agent informs the Solution agent of its stored cost. In this way, a measure of the globally best tour is distributed around the pheromone infrastructure of Place agents. There is no longer a centralised comparison requiring the global knowledge of all tours. Triggering a global update is a small addition to this procedure. When a Solution agent completes its tour, it checks to see what the lowest tour cost encountered was. If the Solution agent's best ever tour cost is less than the lowest tour encountered in the environment, the Solution agent knows its has the best tour. That Solution agent then performs a global update.

3.5 Further Possibilities

When we relax the constraint of remaining as true as possible to the original algorithm, our idea of depositing tour information in the environment introduces many possibilities. The 'Elitist' ant strategy has been found to improve Ant System, the precursor to ACS. This strategy permits e elitist ants to perform a global update on the best tour found so far. In our framework, Solution agents can count the number of better tours encountered in the environment and perform a global update if they are within the top t tours encountered. Alternatively, they could track the percentage difference between their own tour and the best tour encountered and perform a global update accordingly.

4 Preliminary Results

Currently, we are testing the effect of asynchronicity, parallelism and concurrency on the performance of the original Ant Colony System algorithm. There is an inherent bias in its standard implementation (Figure 2) in that ants build full tours one at a time and this *process order* is repeated for the lifetime of the algorithm. For the implementation investigated with this research, we should like to rule out that asynchronous and parallel tour building have any effect on algorithm performance. We should not assume this is the case. Recall that stigmergic mechanisms such as pheromones rely on sensing signals previously deposited in the environment.

In our experiments, we tested two independent variables—the process order used by ants when building tours and the number of steps towards a full tour during an ant's turn. We used two levels of process order—fixed and random. We also used two levels of the process size—an ant solves the whole problem on its turn or an ant makes one step towards solving the problem on its turn. The accuracy, speed and reliability of the algorithm were measured. We found no statistically significant difference in ACS's performance between all combinations of all levels of the independent variables. We are now performing similar experiments on the effect of the number of concurrently active ants on the original algorithm's performance.

5 Related Work

Randall and Lewis (Randall and Lewis, 2002) applied a parallelisation strategy to 8 TSPLIB problems ranging in size from 24 to 657 cities. Their results showed an improvement in speedup and efficiency of solution for problem sizes greater than 200 cities. However, their scheme had a high communication cost.

Stützle has experimented with parallel independent runs of Max-Min Ant System on 7 instances from TSPLIB ranging in size from 198 to 1291 cities (Stutzle, 1998). These experiments showed performance improvements over a single sequential run. Both Stützle's and Randall and Lewis' experiments were in the vein of a traditional parallel computing master/slave approach as opposed to the completely decentralised MAS approach described by this paper. The reader is referred to the literature (Janson et al., 2005) for a comprehensive overview of approaches to parallelisation of ACO.

6 Future Work

Our immediate future work will involve building and testing the architecture and ACS implementation described in this paper. We would also like to consider other ant colony variants such as Max-Min Ant System and other nature-inspired approaches such as Particle Swarm Optimisation.

References

F. Bellifemine, A. Poggi, and G. Rimassa. JADE: a FIPA2000 compliant agent development environment. In *Proceedings of the Fifth International Conference on Autonomous Agents*, pages 216–217. 2001.

S. Brueckner. *Return from the Ant: Synthetic Ecosystems for Manufacturing Control*. Phd, Humboldt University, 2000.

S. Janson, D. Merkle, and M. Middendorf. Parallel Ant Colony Algorithms. In *Parallel Metaheuristics*. Wiley, 2005.

H. V. D. Parunak, P. Weinstein, P. Chiusano, and S. Brueckner. Agents Swarming in Semantic Spaces to Corroborate Hypotheses. In *Proceedings of the Third International Joint Conference on Autonomous Agents and Multi-Agent Systems*, pages 1488–1489. 2004.

M. Randall and A. Lewis. A Parallel Implementation of Ant Colony Optimization. *Journal of Parallel and Distributed Computing*, 62(9):1421–1432, 2002.

E. Ridge, D. Kudenko, D. Kazakov, and E. Curry. Moving Nature-Inspired Algorithms to Parallel, Asynchronous and Decentralised Environments. In *Self-Organization and Autonomic Informatics*, pages 35–49. IOS Press, 2005.

T. Stutzle. Parallelization Strategies for Ant Colony Optimization. In *Proceedings of the Fifth International Conference on Parallel Problem Solving from Nature*, volume 1498 of *Lecture Notes in Computer Science*. 1998.

Exploration vs. Exploitation in Naturally Inspired Search

4th April 2006

Organisers

Mark Bishop, Goldsmiths College

Slawek Nasuto, University of Reading

Programme Committee

Peter Bentley, University College London
Mark Bishop, Goldsmiths College
Jim Kennedy, USA
Bernard Manderick, Vrije Universiteit Brussel
Slawek Nasuto, University of Reading
Vitorino Ramos, University of Lisbon
Luis Rocha, Indiana University
Chris Voudouris, BT Group
David Wolpert, NASA
Qingfu Zhang, University of Essex

Contents

Exploration vs. Exploitation in Naturally Inspired Search

Mark Bishop* and Slawek Nasuto†

*Department of Computing
Goldsmiths College, New Cross
London, UK
m.bishop@gold.ac.uk

†Department of Cybernetics
University of Reading, Reading
Berkshire, UK
s.j.nasuto@reading.ac.uk

Abstract

Nature Inspired Systems (NIS) such as genetic algorithms, particle swarm optimisation and ant colony algorithms are the state-of-the-art solution techniques for some search and optimisation problems. The NIS meeting at AISB06 examined nature inspired systems for parallel, asynchronous and decentralised environments in the context of a fundamental problem - the balance of system resources between exploration of the search space and exploitation of potentially good problem solutions. In this short introduction we will highlight some of the key issues raised in the workshop.

1 Introduction

Any method can solve a search problem given infinite time. This does not demonstrate intelligence; intelligent search is resource limited. Intelligent search must combine exploration of the new regions of the space with evaluation of potential solutions already identified. This necessitates in balancing exploration with exploitation. Too much stress on exploration results in a pure random search whereas too much exploitation results in a pure local search. Clearly, intelligent search methods must reside somewhere in the continuous spectrum in between these extremes.

2 Workshop papers

In addressing the workshop theme, Myatt et al. (4) discuss an efficient naturally inspired *Swarm Intelligence* search technique :- Stochastic Diffusion Search (2). Stochastic Diffusion Search, (SDS), is characterised by its use of objective function decomposition[1] and stochastic agent recruitment mechanisms which together help facilitate increased search efficiency. This paper highlights the effect of different recruitment mechanisms on the balance between exploration and exploitation and concludes by stressing the importance of objective function decomposition.

Another Swarm Intelligence method using objective function decomposition is described by Li and Zhang (5). They investigate the effect of a decomposition-based evolutionary strategy on a bi-objective optimisation problem - the leading ones and trailing zeros problem (LOTZ) and demonstrate that the success of their algorithm is grounded upon the decomposition of LOTZ into a number of scalar optimisation subproblems.

Evolutionary algorithms (EAs) provide an example of a population-based framework in which the issue of exploration vs exploitation has been explored for many years. EAs are often discussed as a Naturally Inspired Search methodology which is effective because it strikes the exploration/exploitation balance in a good position between local and random search. An example of a Machine Learning system using Evolutionary Algorithms is the Learning Classifier Systems (LCS) (3). LCS use Evolutionary Algorithms to learn a set of production rules required to solve a particular problem. In their paper, McMahon et al. (1) describe how the use of a parameter analogous to emotion enables a LCS to vary more efficiently the balance between exploration and exploitation in the woods environment test problem[2].

More recently, other population-based NIS algorithms - such as Ant Colony Optimisation (ACO), Mimetic Algorithms (MA), Particle Swarm Optimisation (PSO) etc - are also believed to find an effective exploration/exploitation ratio. In their cpntribu-

---

[1] By analogy to the partial information about the environment available to individuals in insect societies, the approach advocated here capitalises on the fact that many objective functions are *decomposable* into components that can be evaluated independently. An evaluation of only one or a few of these components – a *partial evaluation* of the objective function – may still hold enough information for search and optimisation purposes.

[2] The woods environment is a grid-based virtual environment in which grid positions are navigated by an agent avoiding obstacles and searching for reward.

tion to this symposium Selvarajah et al. (6) establish stability of the best particle of the Particle Swarm Optimiser under the continuous time condition. These results help characterise the exploration/exploitation balance of PSO systems.

3 Conclusion

A potentially interesting mechanism involved in exploration and exploitation balance highlighted by two of the symposium papers is the use of the decomposable objective functions. The symposium organisers conclude that more research should be done to better understand this mechanism.

References

[1] McMahon, A., Scott, D., Baxter P., Browne W., (2006), An Autonomous Explore/Exploit Strategy, Proc. of the AISB06 Symposium on Nature Inspired Systems.

[2] De Meyer, K., Nasuto, S.J., Bishop, J.M., (2006), Stochastic Diffusion Optimisation: the application of partial function evaluation and stochastic recruitment in Swarm Intelligence optimisation, Volume 2, Chapter 12 in Abraham, A., Grosam, C. and Ramos, V. (eds), (2006), Swarm intelligence and data mining, Springer Verlag.

[3] Goldberg, G., (1989), Genetic Algorithms in optimization and machine learning, Addison Wesley.

[4] Myatt, D., Nasuto, S.J., Bishop, J.M., (2006), Exploration and Exploitation in Stochastic Diffusion Search, Proc. of the AISB06 Symposium on Nature Inspired Systems.

[5] Li, H., Zhang, Q, (2006), A Decomposition-based Evolutionary Strategy for Bi-objective LOTZ Problem, Proc. of the AISB06 Symposium on Nature Inspired Systems.

[6] Selvarajah K., Kadirkamanathan, V., Flemming, P., (2006), Stability Analysis for the Stochastic best Particle Dynamics of a Continuous-time Particle Swarm Optimizer, Proc. of the AISB06 Symposium on Nature Inspired Systems.

Exploration and Exploitation in Stochastic Diffusion Search

D.R. Myatt*
*Department of Cybernetics
School of Systems Engineering
University of Reading
`d.r.myatt@reading.ac.uk`

S.J. Nasuto†
†Department of Cybernetics
School of Systems Engineering
University of Reading
`s.j.nasuto@reading.ac.uk`

J.M. Bishop‡
‡Department of Computing
Goldsmiths College
University of London
`m.bishop@gold.ac.uk`

Abstract

The issue of how decomposable objective functions can form an integral part of the analysis of exploration and exploitation is presented, with the aim of investigating how more efficient optimisation algorithms may be developed. Stochastic Diffusion Search is presented as an example algorithm, and is shown theoretically to be more efficient than simple optimisation techniques that do not employ partial evaluation. The exploration/exploitation balance in SDS is then analysed and methods for altering this balance discussed.

1 Introduction

This paper examines how the issue of exploration and exploitation is modified when considering the class of decomposable objective functions. Primarily, it examines how the ability to perform partial evaluations may be utilised to significantly reduce the computational expense associated with optimisation on this class of functions.

Many naturally inspired optimisation techniques such as Genetic Algorithms (Holland, 1975), Differential Evolution, (Storn and Price, 1997) and Particle Swarm Optimisation, (Kennedy, 1995) all rely upon individuals/particles fully evaluating the objective function at each iteration. In such cases, the issue of exploitation versus exploration is a question of how to efficiently exploit the topology of objective functions to scale minima while still retaining the ability to locate global optima.

However, when considering the class of decomposable objective functions another aspect is added to this debate: each point in the search space can be considered to be a resource for exploitation in its own right. For example, if the objective function may be decomposed into one hundred subfunctions, then many fewer than one hundred subfunction evaluations may be required in order to determine a good approximation of the overall objective function value.

In this way, for this class of functions, the overall computational expense may be reduced below that of a fully-evaluating method. As a result, the issues surrounding this form of exploitation will be focussed on rather than the more typical exploitation of search space gradients.

The algorithm used in this paper to examine the theoretical issues is Stochastic Diffusion Search (SDS). SDS, first described in Bishop (1989b), is an efficient probabilistic multi-agent global search and optimisation technique that has been applied to diverse problems such as site selection for wireless networks (Whitaker and Hurley, 2002), mobile robot self-localisation (Katevas et al., 1997), object recognition (Grech-Cini and McKee, 1993), and text search (Nasuto, 1999). Additionally, a hybrid SDS and n-tuple RAM (Aleksander and Stonham, 1979) technique has been used to locate eyes in video sequences (Bishop and Torr, 1992). Previous analysis of SDS has investigated its convergence, (Nasuto and Bishop, 1999; Myatt et al., 2004) and resource allocation (Nasuto, 1999) using Markov Chains and Ehrenfest Urn models under a variety of noise conditions.

One of the advantages of SDS is that its expected behaviour may be well-defined theoretically in a range of search spaces, and consequently it is well-suited to a theoretical analysis into exploration and exploitation.

Two main topics will be considered: the theoretical advantages that performing partial evaluations can yield over fundamental types of search, and also the factors determining the exploration and exploitation ratio in SDS.

2 Stochastic Diffusion Search

SDS is usually applied to pattern search and matching, where the search space is discrete but is generally a quantisation of some continuous manifold that contains meaningful gradients.

Such problems can frequently be cast in terms of optimisation by defining the objective function, $F(\mathbf{x})$, for a hypothesis $\mathbf{x}$ about the location of the solution, as the similarity between the target pattern and the corresponding region at $\mathbf{x}$ in the search space and finding $\mathbf{x}$ such that $F(\mathbf{x})$ is maximised. In general, SDS can most easily be applied to optimisation problems where the objective function is decomposable into components that can be evaluated independently:

$$F(\mathbf{x}) = \sum_{i=1}^{n} F_i(\mathbf{x}), \qquad (2.1)$$

where $F_i(\mathbf{x})$ is defined as the i^{th} partial evaluation of $F(\mathbf{x})$.

In order to locate the optima of a given objective function SDS employs a population of n agents, each of which maintains a hypothesis about the optima. The algorithm entails the iteration of Test and Diffusion Phases until agents converge upon the optimum hypothesis.

Test Function: The boolean Test Function returns whether a randomly selected partial evaluation of the objective function is indicative of a 'good' hypothesis. For example, in pattern matching the Test Function may return True if the i^{th} sub-feature of the target pattern is present at position $(\mathbf{x}, i)$ in the search space.

Test Score: The Test Score for a given hypothesis, $\mathbf{x}$, is the probability that the Test Function will return true, and is hence representative of the value of $F(\mathbf{x})$.

Initialisation: Typically the initial hypothesis of each agent is selected uniformly randomly over the search space. However, any information about probable solutions available *a priori* can be used to bias the initial selection of hypotheses.

Test Phase: Each agent applies the Test Function to its current hypothesis. If the Test Function returns true the agent becomes *active*, if not then it becomes *inactive*.

Diffusion Phase: Through intercommunication between the agents hypotheses are copied from active to inactive agents through some recruitment mechanism. Inactive agents that do not locate promising hypotheses through this communication randomise their hypothesis over the entire search space.

Convergence: As iterations progress clusters of agents with the same hypothesis form. At termination the largest cluster of agents defines the optimal solution.

3 Efficiency of SDS

This section compares the efficiency of SDS with the basic optimisation methods of fully-evaluating exhaustive and random search, thus demonstrating the effect that exploiting partial evaluation can have.

In order to make the comparison, some approximation of the computational expense of SDS in this search space is needed. SDS has been previously shown to have linear time complexity with respect to the size of the search space (Bishop, 1989a; Nasuto et al., 1998), and this property will be re-examined under the homogeneous background noise model to improve the estimate.

It is assumed that the objective function may be decomposed into k subfunctions, that there is a population of n agents, and the search space has s elements, of which one corresponds to the global optimum with Test Score α. All non-optimal solutions have Test Score β, corresponding to homogeneous noise.

The behaviour of SDS before the optimal solution has been located is under consideration, and therefore the observed behaviour will be equal to that observed in a modified search space with *no* optimal hypothesis. The stationary state in the modified search space would be simply β, and hence a consistent transient state with a mean activity level of β will be apparent in the true search space until the optimal solution is found.

It is necessary to calculate the expected proportion of agents that will generate random hypotheses every iteration. The agents that will reselect new hypotheses at random are inactive agents that select other inactive agents during the diffusion phase, and therefore as a proportion of the population this will be $(1-\beta)^2$. Therefore, the expected number of hypotheses randomly selected at each iteration will be $n(1-\beta)^2$.

Let the probability ρ be the acceptable probability of not having located the optimal solution after i iterations. A typical value of ρ would be 0.01, for example. It follows that

$$(1 - \frac{1}{s})^{in(1-\beta)^2} \leq \rho. \qquad (3.1)$$

Simple rearrangement of this expression in terms of i

produces the expected number of iterations as

$$i = \frac{\ln \rho}{n(1-\beta)^2 \ln(1 - 1/s)}. \tag{3.2}$$

For large search space size s, this can be further simplified as

$$\ln(1 - 1/s) \approx -\frac{1}{s}, \tag{3.3}$$

This value is quickly approached, even for small search spaces. This is easily shown by considering the expansion of $\ln(1 - 1/s)$:

$$\ln(1 + x) = \sum_{k=1}^{\infty} \frac{(-1)^{k-1} x^k}{k} \tag{3.4}$$

$$\therefore \ln\left(1 - \frac{1}{s}\right) = -\frac{1}{s} - \frac{1}{2s^2} - \frac{1}{2s^3} \dots \tag{3.5}$$

$$\ln\left(1 - \frac{1}{s}\right) \propto -\frac{1}{s} - O\left(\frac{1}{s^2}\right) \tag{3.6}$$

So, even for a very small search space of $s = 1000$, this approximation yields an error of only $\approx 5 \times 10^{-7}$. The final expression for the expected number of iterations is then

$$i = s\frac{\ln \frac{1}{\rho}}{n(1-\beta)^2}, \tag{3.7}$$

which is linear with respect to s.

SDS can also exhibit sublinear time complexity (Nasuto, 1999), but this is only true over the region where the agent population is comparable to the size of the search space. So, although using an appropriately large agent population would always technically yield sublinear complexity for small perturbations in the search space size, in reality, this would result in significantly more computational expense for a given search space size s.

3.1 Comparison of SDS with Exhaustive Search

This section demonstrates that few objective subfunctions are required in order to make SDS more efficient than an exhaustive search method. Intuitively, it is apparent that the more subfunctions an objective function can be decomposed into, the more efficient SDS will become, as each agent only needs to evaluate one partial evaluation per iteration.

The number of partial evaluations required for an exhaustive search, is trivially sk, as all objective subfunctions must be evaluated for each position within the search space.

$$\chi_{\text{exhaustive}} = sk \tag{3.8}$$

For SDS the number of partial evaluations required, assuming instantaneous convergence after the time-to-first-hit, is

$$\chi_{\text{SDS}} = ni, \tag{3.9}$$

where i is the expected number of iterations, and therefore the number of partial evaluations required for SDS can be accurately approximated by

$$\chi_{\text{SDS}} = s\frac{\ln \frac{1}{\rho}}{(1-\beta)^2}, \tag{3.10}$$

So, in order for SDS to be more efficient than exhaustive search,

$$\chi_{\text{SDS}} < \chi_{\text{exhaustive}}, \tag{3.11}$$

$$s\frac{-\ln \rho}{(1-\beta)^2} < sk. \tag{3.12}$$

Therefore, the minimum number of partial evaluations required, k_{min}, for which SDS is more efficient than exhaustive search, is

$$k_{\text{min}} > -\frac{\ln \rho}{(1-\beta)^2}. \tag{3.13}$$

For an acceptable possibility of failure $\rho = 0.01$ in the noiseless case, $k_{\text{min}} = 4.605$, and therefore an objective function decomposable into at least five subfunctions would be required in order to show an improvement over an exhaustive search. For a highly noisy case, with $\beta = 0.4$, $k_{\text{min}} = 12.792$, and therefore a minimum of thirteen partial evaluations would be required.

3.2 Comparison of SDS with Random Search

For a fully-evaluating random search technique, the probability ρ, of not having found a single optimal solution after i iterations is

$$\rho = \left(1 - \frac{1}{s}\right)^i, \tag{3.14}$$

and therefore

$$i = \frac{\ln \rho}{\ln\left(1 - \frac{1}{s}\right)} \approx -s \ln \rho. \tag{3.15}$$

So, the expected number of partial evaluations needed for the fully evaluating random search algorithm is

$$\chi_{\text{random search}} = -sk \ln \rho \tag{3.16}$$

And therefore SDS will have superior performance for

$$k_{\min} > \frac{1}{(1-\beta)^2}. \tag{3.17}$$

So, in the noiseless case SDS will always be superior to pure random search, under the assumption of instantaneous convergence and a zero probability of cluster failure. Conversely, in noisy search spaces, for example $\beta = 0.5$, at least four possible partial evaluations are necessitated.

3.3 Discussion

Although it was assumed that there was a zero probability of cluster failure and instantaneous convergence, the values derived in this section form a useful lower bound on the number of partial evaluations required to achieve increased efficiency, thus aiding identification of problems for which SDS may be superior.

Moreover, for the class of decomposable objective functions there will always be some minimum number of possible subfunctions for which SDS will be superior to pure random/exhaustive search.

In general, other heuristic optimisation algorithms can only give superior performance to random search if meaningful gradients are found within the search space. Therefore, showing that standard SDS can be more efficient than random search suggests increased performance over many heuristic algorithms in the absence of such gradients.

4 Comparative Efficiency of SDS

This section outlines how simple changes in agent behaviour affect the exploration and exploitation of SDS. The amount of exploration is dependent on the proportion of agents that select new solutions at each iteration. Exploitation, for SDS, is related to the number of agents assigned to a given point in the search space, while exploitation in most swarm algorithms is a function of the number of particles/individuals within the basin of attraction of a given minimum. In this sense, the sampling of an area around a minimum can also be seen as some sort of partial evaluation, as it is assumed that points nearby in the search space give an indication of the quality of the minimum.

Let us consider a large search space with a global optimum of Test Score α, homogeneous background noise β, and also a small number of local minima with varying Test Score high enough to induce convergence given β. It is likely that SDS will form a significant cluster on one of the other minima first before locating and converging on the global optimum. Moreover, several different transitory clusters may be formed as progressively better solutions are found.

What will be observed are several successive quasi-stationary states in the proportion of active agents. It will be shown that the proportion of resources available for exploration is purely a function of the Test Score of the best minima located so far, and thus independent of the homogeneous noise.

Since it is assumed that there are few such local minima in the search space, the behaviour of SDS once convergence to the optimal solution has occurred will not significantly differ from the case where the homogeneous noise only were present.

If the number of other search space positions with Test Scores greater than that of the homogeneous noise became non-negligible compared to the size of the search space, though, then this model would begin to break down.

4.1 SDS as a Discrete Dynamical System

This section presents a model of SDS that will facilitate a formal analysis of exploration and exploitation.

Although SDS is inherently stochastic, it has been demonstrated that examining only the mean behaviour of SDS, in terms of the expected cluster transitions, can yield useful results (Myatt et al., 2004).

Let the expected proportion of active agents in the optimal cluster in iteration i be $\bar{c}_i$. In iteration $i+1$, the expected value must be some function of $\bar{c}_i$, α and β, and thus can be written as

$$\bar{c}_{i+1} = f\left(\bar{c}_i, \alpha, \beta\right). \tag{4.1}$$

α and β are assumed constant over all iterations (non-dynamic search space) and, hence, f is an *iterated map function*, and the sequence $\bar{c}_{0\ldots i}$ describes a discrete dynamical system. Therefore, formulating SDS in this context allows investigation using the tools of dynamical systems theory.

Based on the observed value of $\bar{c}_i$ and the search space parameters α and β, Figure 1 summarises the expected proportion of the population in each state at iteration i.

Therefore, from Figure 1 it can be written that the expected 1-step optimal cluster size evolution will be:

$$\bar{c}_{i+1} \equiv f(\bar{c}_i, \alpha, \beta) = \alpha\left(\bar{c}_i + g_{\text{inactive}}(\alpha, \beta, \bar{c}_i)\bar{c}_i\right), \tag{4.2}$$

where

$$g_{\text{inactive}}(\alpha, \beta, \bar{c}_i) = \left[\frac{1-\alpha}{\alpha}\bar{c}_i + (1-\beta)\left(1 - \frac{\bar{c}_i}{\alpha}\right)\right]. \tag{4.3}$$

The stationary state, $\gamma \equiv c_\infty$, that will be approached after convergence may be determined by applying the condition

$$\gamma = f(\gamma, \alpha, \beta) \tag{4.4}$$

This yields the stationary state

$$\gamma = \begin{cases} \frac{\alpha(2-\beta)-1}{\alpha-\beta}, & \text{if } \alpha > \frac{1}{2-\beta}; \\ 0, & \text{Otherwise.} \end{cases} \tag{4.5}$$

It is important to note that the stationary state of the dynamical system model, γ, corresponds to a *quasi*-stationary state of the underlying stochastic process. Therefore, γ will generally be referred to as quasi-stationary to reflect the reality of the system.

4.2 Noise Dependence of Exploration/Exploitation Balance

This section demonstrates that the exploration/exploitation balance in standard SDS is purely a function the best minima located so far, and thus is independent of the remaining noise in the search space.

Consider the quasi-stationary state, γ, of the active agents within the optimal cluster, as defined in (4.5). The overall quasi-stationary state, of *all* active agents, γ', must be equal to

$$\gamma' = \gamma + \beta\left[1 - \frac{\gamma}{\alpha}\right], \tag{4.6}$$

as the expected number of active agent associated with the noise solutions (shown in Figure 1) is valid for any $\bar{c}_i \in [0, 1]$.

Therefore, it can be seen that the overall quasi-stationary state of active agents is equal to

$$\gamma' = 2 - \frac{1}{\alpha}, \tag{4.7}$$

which is independent of β. It is also the quasi-stationary state of active agents observed when $\beta = 0$, showing that the overall quasi-stationary state is always independent of β.

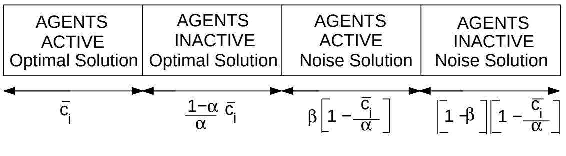


Figure 1: Illustration of the current state of the agent population in iteration i in terms of $\bar{c}_i$, α and β.

Table 1: A table demonstrating the expected percentage of agents in the population per iteration that are free for exploration ($100\varepsilon\%$) after SDS has reached a quasi-stationary state associated with a minima of Test Score α'.

α'	Exploration(%)	α'	Exploration(%)
0.51	92.3	0.75	11.1
0.55	66.9	0.8	6.3
0.6	44.4	0.85	3.1
0.65	29.0	0.9	1.2
0.7	18.4	0.95	0.3

Given that the overall activity is defined by (4.7), the proportion of agents ε selecting random hypotheses at each iteration will be $(1-\gamma')^2$, so

$$\varepsilon = \left(\frac{1}{\alpha} - 1\right)^2. \tag{4.8}$$

Table 1 shows the expected percentage of agents ε that generate new hypotheses at each iteration, given that convergence has taken place to one of the local minima i.e. the quasi-stationary state of activity has been reached. The Test Score of the local minima is denoted α'.

It can be seen that as the Test Score of the local minima increases, the proportion of exploration decreases rapidly - for $\alpha' = 0.95$, only 0.3% of agents will be free for exploration, corresponding to only three agents out of a typical population size of one thousand.

Obviously, if considering an objective function where the local minima may have a Test Score > 0.6, the explorational capabilities of standard SDS may be considered inefficient as the majority of resources (in the form of agents) will be concentrating on retaining and validating the current best solution.

4.3 Context-sensitivity

Context-sensitive SDS (Nasuto, 1999) is an alternative resource allocation mechanism designed to improve the explorational ability of SDS by reducing the proportion of resources allocated to currently known solutions.

During the Diffusion phase, each *active* agent also selects another agent at random (in contrast to only inactive agents selecting in standard SDS). If the selected agent is *also* active, and is associated with the same hypothesis, then the selecting agent becomes inactive.

Intuitively, it can be seen that this mode of resource allocation will commit less agents to the optimal solution, thereby leaving more agents free to explore the search space.

Inactive agents behave in exactly the same way as with standard SDS, and therefore context-sensitivity can be seen as purely an *addition* to standard SDS, rather than a modification.

It is therefore necessary to alter the discrete dynamical system developed for standard SDS to incorporate the new behaviour introduced by context-sensitivity.

Returning to the discrete domain, consider a cluster of k active agents in a population of n agents. Each agent in the cluster selects another agent uniformly randomly, so each such event is a Bernoulli trial with a probability of success of $\frac{k}{n}$ (success is defined as selecting another agent in the cluster). This process can therefore be described by a binomial probability distribution, and hence the expected number of successes would be $\frac{k^2}{n}$. Therefore, the proportion of the overall population that would become inactive is $\frac{k^2}{n^2}$. Generalising this to the continuous domain, $\bar{c}_i \equiv \frac{k}{n}$, and therefore the reduction in the size of the cluster would be $\bar{c}_i^2$.

The overall system equation for context sensitive SDS is consequently given by

$$\bar{c}_{i+1} = f_{\text{cs}}(\bar{c}_i, \alpha, \beta) = \alpha\left(\bar{c}_i + g(\bar{c}_i, \alpha, \beta)\bar{c}_i - \bar{c}_i^2\right) \tag{4.9}$$

which can be simplified to

$$f_{\text{cs}}(\bar{c}_i, \alpha, \beta) = \bar{c}_i[\alpha(2-\beta)] - \bar{c}_i^2[2\alpha - \beta]. \tag{4.10}$$

This expected stationary state is again found by applying the condition (4.4) - in this case, rearranging yields the result

$$\gamma_{\text{cs}} = \begin{cases} \frac{\alpha(2-\beta)-1}{2\alpha-\beta}, & \text{if } \alpha > \frac{1}{2-\beta}; \\ 0, & \text{Otherwise.} \end{cases} \tag{4.11}$$

The form of (4.11) is almost identical to that of the standard stationary state (4.5), except for the factor of $(2\alpha - \beta)$ in the denominator, rather than $(\alpha - \beta)$. Therefore, in the noiseless case the stationary state of context-sensitive SDS is always half that of standard SDS. However, in the noisy case this no longer holds and the stationary state behaviour will become more complex.

Table 2: A table showing the expected percentage of agents in the population per iteration that are free for exploration ($100\varepsilon\%$) after context-sensitive SDS has reached a quasi-stationary state associated with a minima of Test Score α'. A value of $\beta = 0.3$ is assumed.

α'	Exploration($\%$)	α'	Exploration($\%$)
0.51	60.2	0.75	31.6
0.55	54.3	0.8	27.8
0.6	47.5	0.85	24.4
0.65	41.4	0.9	21.6
0.7	36.2	0.95	19.1

For context-sensitive SDS, the expected overall activity level after convergence is

$$\gamma'_{\text{cs}} = \frac{2\alpha - 1 - \beta\left(2 - \alpha - \frac{1}{\alpha}\right)}{2\alpha - \beta}, \tag{4.12}$$

and therefore the proportion of agents available for exploration is

$$\varepsilon_{\text{cs}} = \left[\frac{1 + \beta\left(1 - \alpha - \frac{1}{\alpha}\right)}{2\alpha - \beta}\right]^2. \tag{4.13}$$

Due to the dependence of γ' on β, the proportion of agents available for exploration is no longer independent of noise, unlike with standard SDS. However, the same basic trend of monotonically decreasing exploration with respect to increasing α is observed.

Table 2 illustrates the contrast between the explorational capabilities of standard and context-sensitive SDS. It can be seen that a much larger proportion of resources are available for exploration - for a minima with Test Score of 0.95, this is now 19% rather than only 0.3%. Therefore, it is expected that context-sensitive SDS will be more computationally efficient in general.

However, 19% of resources remains a relatively small amount of the population, so further work could include attempting to develop agent behaviour that further reduces the exploitation aspect of minima and improving exploration. However, such a mechanism is likely to have a deleterious effect on robustness (the ability of SDS to converge to a minima of a specified Test Score with a given level of homogeneous noise).

5 Conclusion

This paper has discussed the effects on the question of exploration versus exploitation in decomposable objective functions, allowing the exploitation of individual points within the search space as well as gradients.

Stochastic Diffusion Search was highlighted as an example of a technique that exploits decomposable objective functions, the conditions for which a saving in computational expense relative to basic fully-evaluating methods were determined.

Subsequently, the search space factors affecting the exploration/exploitation balance of SDS were described, and it was demonstrated how this is solely dependent upon the best current minima with standard SDS. However, it was observed that the proportion of agents free for exploration could be small when faced with a high quality local minima, and consequently context-sensitive SDS was described in analysed in this context and shown to be superior.

More generally, the topic of exploitation is related to making optimal use of information within the objective function to increase the efficiency of optimisation algorithms. Usually, only information found within the gradients of the objective function landscape are considered, but this paper has demonstrated that partial evaluation may provide another useful tool for achieving this goal.

References

I. Aleksander and T.J. Stonham. A guide to pattern recognition using random-access memories. In *Computers and Digital Techniques*, volume 2, pages 29–40, 1979.

J.M. Bishop. *Anarchic Techniques for Pattern Recognition*. PhD thesis, University of Reading, 1989a.

J.M. Bishop. Stochastic searching networks. In *Proc. 1st IEE Conf. on Artificial Neural Networks*, pages 329–331, 1989b.

J.M. Bishop and P. Torr. *The Stochastic Search Network*, pages 370–387. Chapman and Hall, 1992.

H.J. Grech-Cini and G.T. McKee. Locating the mouth region in images of human faces. In *Proceedings of SPIE - The International Society for Optical Engineering*, volume 2059, Sensor Fusion VI, pages 458–465, 1993.

J.H. Holland. *Adaptation in Natural and Artificial Systems*. University of Michigan Press, 1975.

N.I. Katevas, N.M. Sgouros, S.G. Tzafestas, G. Papakonstantinou, P. Beattie, J.M. Bishop, P. Tsanakas, and D. Koutsouris. The autonomous mobile robot senario: A sensor-aided intelligent navigation system for powered wheelchairs. *IEEE Robotics and Automation*, 4(4):60–70, 1997.

Eberhart R. Kennedy, J. Particle swarm optimization. In *Proc. IEEE Int. Conf. on Neural Networks*, pages 1942–1948, 1995.

D.R. Myatt, J.M. Bishop, and S.J. Nasuto. Minimum stable convergence criteria for stochastic diffusion search. 40(2):112–113, 2004.

S.J. Nasuto. *Resource allocation of the Stochastic Diffusion Search*. PhD thesis, University of Reading, 1999.

S.J. Nasuto and J.M. Bishop. Convergence analysis of stochastic diffusion search. *Journal of Parallel Algorithms and Applications*, 14(2):89–107, 1999.

S.J. Nasuto, J.M. Bishop, and S. Lauria. Time complexity of stochastic diffusion search'. In *Proc. Neural Computation '98*, 1998.

R. Storn and K. Price. Differential evolution - a simple and efficient heuristic for global optimization over continuous spaces. *J. of Global Optimization*, 11:341–359, 1997.

R.M. Whitaker and S. Hurley. An agent based approach to site selection for wireless networks. In *Proc. ACM Symposium on Applied Computing*, pages 574–577, 2002.

A Decomposition-based Evolutionary Strategy for Bi-objective LOTZ Problem

Hui Li
*Department of Computer Science, University of Essex
Wivenhoe Park, Colchester, CO4 3SQ, UK
`hlil@essex.ac.uk`

Qingfu Zhang
†Department of Computer Science, University of Essex
Wivenhoe Park, Colchester, CO4 3SQ, UK
`qzhang@essex.ac.uk`

Abstract

In this paper, we present a decomposition-based evolutionary strategy to solve bi-objective leading ones and trailing zeros problem (mLOTZ). The proposed algorithm decomposes a MOP into a number of scalar optimization subproblems based on Tchebycheff scalarizing approach. The result shows that our algorithm is very promising.

1 Introduction

A multi-objective optimization problem (MOP) can be stated as follows:

$$\begin{array}{ll}\text{maximize} & f(\mathbf{x}) = (f_1(\mathbf{x}), f_2(\mathbf{x}), \ldots, f_p(\mathbf{x}))^T \\ \text{subject to} & \mathbf{x} \in S\end{array} \quad (1)$$

where $\mathbf{x} = (x_1, \ldots, x_n)^T$ is the decision variable vector. $f_i : \mathbf{R}^n \rightarrow \mathbf{R}, i = 1, \ldots, p$ are objectives functions. Let $\mathbf{x}, \mathbf{y} \in S$, $\mathbf{x}$ is said to be dominated by $\mathbf{y}$ if $f_i(\mathbf{y}) \geq f_i(\mathbf{x})$ for all $i = 1, \ldots, p$ and $f_j(\mathbf{y}) > f_j(\mathbf{x})$ for at least one index j. A solution $\mathbf{x}^* \in S$ is said to be pareto-optimal to (1) if there does not exist another solution $\mathbf{x}$ such that $\mathbf{x}^*$ is dominated by $\mathbf{x}$. $\mathbf{f}(\mathbf{x}^*)$ is then called a pareto-optimal objective vector. The set of all the pareto-optimal objective vectors is called the pareto-optimal front.

A number of multiobjective evolutionary algorithms (MOEAs) have recently been proposed. Compared with classical multi-objective optimization algorithms, MOEAs are able to find multiple non-dominated solutions in a single run. Decomposition is a basic strategy used in conventional multiobjective optimization. This strategy is rarely adopted in MOEAs. In this paper, we propose a new multi-objective evolutionary strategy based on decomposition, called MOES/D. The proposed algorithm decomposes a MOP into a number of scalar optimization subproblems based on Tchebycheff scalarizing approach. It optimizes these subproblems simultaneously. Each subproblem can have several neighboring subproblems. Information collected from optimization of a subproblem is used for optimization of its neighbors.

We compared our proposed algorithm with several other well-known MOEAs on four test instances of mLOTZ. The experimental result shows that our algorithm outperforms the other MOEAs under consideration.

2 The description of MOES/D

In MOES/D, The MOP (1) is decomposed into say, N, scalar optimization subproblems by using the weighted Tchebycheff approach. The goal of subproblem i is to minimize:

$$g^{(i)}(\mathbf{x}, \lambda^i) = \max_{j \in \{1,\ldots,p\}} \lambda^i_j \cdot |f^*_j - f_j(\mathbf{x})| \quad (2)$$

where $\lambda^i = (\lambda^i_1, \ldots, \lambda^i_p)$ is the weight vector with $\lambda^i_j \geq 0$ for all $j = 1, \ldots, p$ and $\sum_j \lambda^i_j = 1$. f^*_j is an upper approximation of the maximum of $f_j(\mathbf{x})$. If N is large enough and these weight vectors are uniformly distributed, then under a mild condition, the N optimal solutions to these subproblems will be a good approximation to the Pareto-optimal front. In the case of bi-objectives, $\lambda^i = (\lambda^i_1, \lambda^i_2)$ $i = 1, 2, \ldots, N$ can be set as:

$$\lambda^i_1 = (i-1)/(N-1), \quad \lambda^i_2 = 1 - \lambda^i_1.$$

Subproblem i is called a neighbor of subproblem j if λ^i is among the K closest weight vectors to λ^j. At each generation, MOES/D maintains a population of N solutions $\mathbf{x}^1, \ldots, \mathbf{x}^N$, where $\mathbf{x}^i \in S, i = 1, \ldots, N$, is the best solution to subproblem i found so far during search. The framework of MOES/D is described in Fig.1.

```
Step 1 Initialization
    Construct a set of $N$ evenly spaced weight vectors $\lambda^1, \ldots, \lambda^N$ and generate $N$ solutions $\mathbf{x}^1, \ldots, \mathbf{x}^N$ to form the initial population, and set the external population $EP$ to be empty set.

Step 2 For $i = 1 : N$
    Step 2.1 Mutation Mutate $\mathbf{x}^i$ and obtain $\mathbf{y}$.

    Step 2.2 Internal Update For $j = 1 : N$
    If subproblem $j$ is a neighbor of subproblem $i$ and $\mathbf{y}$ is better than $\mathbf{x}^j$ for subproblem $j$, then set $\mathbf{x}^j = \mathbf{y}$.

    Step 2.3 External Update Add $\mathbf{y}$ into any solutions in $EP$ if $\mathbf{y}$ is nondominated by $EP$ and remove all members of $EP$ dominated by $\mathbf{y}$.

Step 3 Termination
    If termination condition is met, then return $EP$; otherwise, goto Step 2.
```

Figure 1: Framework of MOES/D

3 Test Problem: mLOTZ

The two objectives to be maximized in mLOTZ are:

$$LOTZ1(x) = \sum_{i=1}^{n} \prod_{j=1}^{i} x_j, \tag{3}$$

$$LOTZ2(x) = \sum_{i=1}^{n} \prod_{j=i}^{n} (1 - x_j) \tag{4}$$

where $x = (x_1, \ldots, x_n) \in \{0, 1\}^n$.

The objective space of mLOTZ can be partitioned into the following $(n + 1)$ sets

$$Y_i = \{x | LOTZ1(x) + LOTZ2(x) = i\}, \quad i = 1, \ldots, n.$$

The Pareto-optimal front is Y_n. **mLOTZ** has been used for testing the performance of MOEAs by some researchers Laumanns et al. (2004).

4 Experimental Results

4.1 Experimental Settings

In our experiments, we used four instances - mLOTZ50, mLOTZ60, mLOTZ70, and mLOTZ80 with $m = 50, 60, 70, 80$ respectively. We compared MOES/D with SEMO Laumanns et al. (2004), MOGLS Jaszkiewicz (1998) and NSGA2 Deb et al. (2002) on these four instances. The initial population size of MOGLS is set to be n. The population size of MOES/D and NSGA2 is also set to be n. K in MOES/D is set to be 20. The size of the temporary population in MOGLS is set to 10. No extra parameter is needed in SEMO. All algorithms stop after $1000 \times n$ function evaluations. Besides, for the fair comparison, we only use one-bit mutation in all algorithms.

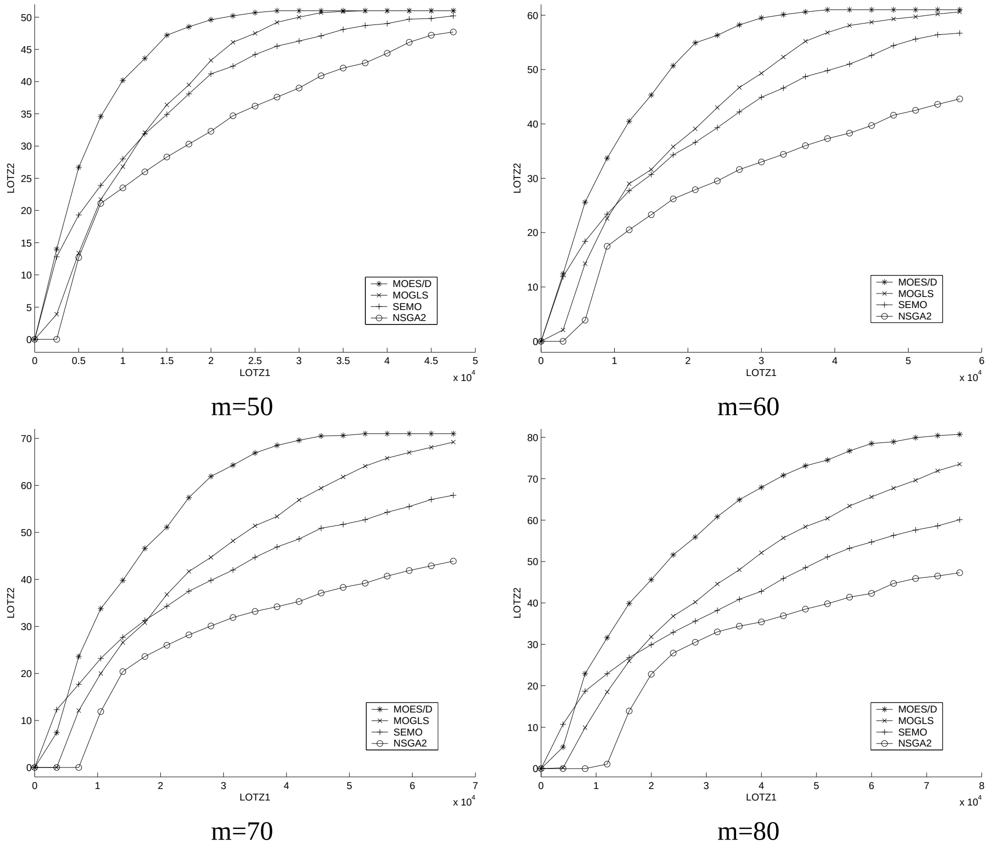
To measure the performances of these algorithms, we consider the cardinality-based metric - Overall Nondominated Vector Generation (ONVG), which counts the number of solutions found in the pareto-optimal front.

4.2 Experimental Results

Table 1 compares the ONVG values of four algorithms. We run each algorithm for each instance 10 times. This table gives the $best$, $mean$ and standard deviation (std) of the ONVG values of each algorithm in 10 runs. It is clear that MOES/D outperforms all other algorithms in terms of solution quality. To study the convergence speed of four algorithms, the evolution of the mean of the ONVG values of four algorithms with the number of function evaluations is given in Fig.2, which clearly shows that MOES/D is the fastest one.

Table 1: The comparison of ONVG of SEMO, NSGA2, MOGLS and MOES/D

OVNG(S,R)	Test Instance	mLOTZ50	mLOTZ60	mLOTZ70	mLOTZ80
SEMO	best	51	61	63	66
	mean	50.5	57.7	58.2	61.5
	std	1.08	5.98	3.22	2.46
NSGA2	best	50	55	58	65
	mean	48.1	45.5	45.2	48.2
	std	2.25	5.58	7.68	8.02
MOGLS	best	51	61	71	81
	mean	51	60.6	69.2	74.4
	std	0	0.84	1.81	5.83
MOES/D	best	51	61	71	81
	mean	51	61	71	80.9
	std	0	0	0	0.31



m=50

m=60

m=70

m=80

Figure 2: The evolution of $mean$ with the no of function evaluations in four algorithms: SEMO, NSGA2, MOGLS and MOES/D

5 Conclusions

We proposed a new MOEA based on decomposition strategy. The experimental results shows that our algorithm outperforms other three algorithms in terms of the solution quality and the speed on four **mLOTZ** test instances. In the future, we will study the behavior of the proposed algorithm and apply it to other MOPs.

References

Kalyanmoy Deb, Samir Agrawal, Amrit Pratap, and T. Meyarivan. A fast and elitist multiobjective genetic algorithm: Nsga-ii. *IEEE Trans. Evolutionary Computation*, 6(2):182–197, 2002.

A. Jaszkiewicz. Genetic local search for mulitple objective combinatorial optimization. Technical Report RA-014/98, Institute of Computing Science, Poznaň University of Technology, 1998.

Marco Laumanns, Lothar Thiele, and Eckart Zitzler. Running time analysis of multiobjective evolutionary algorithms on pseudo-boolean functions. *IEEE Trans. Evolutionary Computation*, 8(2):170–182, 2004.

An Autonomous Explore/Exploit Strategy

Alex McMahon
*University of Reading,
Berkshire
RG6 6AY, UK
siu01ajm@rdg.ac.uk

Dan Scott
University of Reading,
Berkshire
RG6 6AY, UK
siu01ds@rdg.ac.uk

Paul Baxter
University of Reading,
Berkshire
RG6 6AY, UK
p.e.baxter@rdg.ac.uk

Dr. Will Browne*
w.n.browne@rdg.ac.uk

Abstract

In reinforcement learning problems it has been considered that neither exploitation nor exploration can be pursued exclusively without failing at the task. The optimal balance between exploring and exploiting changes as the training progresses due to the increasing amount of learnt knowledge. This shift in balance is not known a priori so an autonomous online adjustment is sought. Human beings manage this balance through logic and emotions based on feedback from the environment. The XCS learning classifier system uses a fixed explore/exploit balance, but does keep multiple statistics about its performance and interaction in an environment. Utilizing these statistics in a manner analogous to logic/emotion, autonomous adjustment of the explore/exploit balance was achieved. This resulted in reduced exploration in simple environments, which increased with the complexity of the problem domain. It also prevented unsuccessful 'loop' exploit trials and suggests a method of dynamic choice in goal setting.

1 Introduction

“The dilemma is that neither exploration nor exploitation can be pursued exclusively without failing at the task.” (Sutton and Barto, 1998). The optimal balance between exploring and exploiting changes as the training progresses due to the increasing amount of learnt knowledge, which is not known a priori. Many reinforcement learning algorithms, such as XCS (Wilson, 1995), use a fixed explore/exploit ratio that may lead to unnecessary exploration steps, repetitive loops in exploit trials and an increased amount of time needed for training. Additionally, Whitley (1997) identifies that repetition (resampling) is important to avoid in learning algorithms, which is to be considered in this work.

The basic hypothesis was to adapt the technique of 'confusion matrices' (Holmes, 1997) from the medical domain. A learning system containing a matrix with high true positives and true negatives would require much less exploration than one with many false classifications. This method would struggle in multistep domains with delayed rewards. It would also not direct the learning towards areas of misclassification and thus resampling would occur.

Therefore, inspiration and analogy was sought from nature as humans use logic coupled with emotions to balance the learning task.

1.1 Emotions and explore/exploit strategies

One theory regarding the purpose of emotions is that it provides a means for enabling animals to cope with a complex and unreliable environment (Fellous, 2004). The benefits of emotions can be discussed in terms of the individual, “from fast perceptions of threats, to focusing and redirecting attention, to influencing memory storage and retrieval” (Scheutz, 2004) or in terms of communication techniques within societies; “whether they are vocal, postural or facial, emotional expressions are compact messages exchanged between individuals.” (Fellous, 2004). There is also, however, a great deal of confusion and ambiguity as to what constitutes emotions, with many phenomena being grouped under the banner of emotions. Despite this, a general definition of emotion could include the following: “emotions are states elicited by rewards or punishers, including changes in rewards or punishments” (Rolls, 1999).

A number of cognitive science studies have examined emotions in humans, and have explored the physical processes that underlie them. Accordingly, the limbic system is known to have a vital role. In addition to this, it is known that the hippocampus and the anterior cingulate cortex, key elements of the limbic system, play important parts in the functioning of memory processes (Bush, et al., 2000, Cowan, 1997, Squire, et al., 1992). It can then be said that emotions play a central role in normal cognitive functioning, as memory is arguably the basis of all cognition. Marvin Minsky's suggestion that emotions are an essential part of intelligence, rather than merely a set of supporting functions (Minsky, 1986), is thus supported by biological evidence. It is this theory that motivates the consideration of emotions in the field of artificial intelligence (AI).

In developing an emotion based strategy, it is worth considering what purpose it would serve in the AI architecture, and how it would be implemented. Wright's (Wright, 1996) proposal that emotions allow an agent to respond to urgent needs in real time strongly indicates that directing the explore/exploit balance online would be a useful strategy. It must be noted at this point that because of the ambiguity concerning the nature of emotions, this work does not claim to replicate them, but instead utilizes some of the associated concepts. As such, the word 'emotion' is utilized as a useful analogy; the phrase 'parameters to determine the level of exploration' could be substituted.

1.2 Aims and Objectives

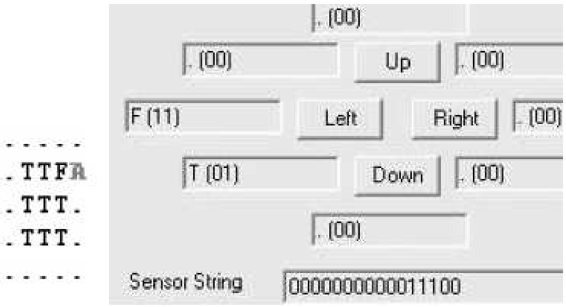
The aim of this project is to investigate whether the simulation of emotions can be used to create an autonomous explore/exploit strategy. The genetics-based machine learning technique of learning classifier systems, namely XCS, will be used as a base platform due to the statistics that it keeps on its interaction with the environment. Both Markov decision processes (MDP) and partially observable Markov decision processes (POMDP) will be tested to observe the effect of the strategy (Butz, 2004).

The structure of the paper continues with an overview of the important types of environment that an artificial agent (animat) may encounter. Existing AI architectures relevant to this work are described. The explore/exploit strategy based on the emotional architecture is detailed prior to testing on multistep domains. The results are discussed in terms of the explore/exploit strategy and the analogy with emotions.

2 Types of Environment

2.1 Single/Multi step

In a single step environment the choice of action from any given state has no bearing on any future state of the environment, with the result of the action being an instantaneous reward level. If the choice of action affects the future states of the environment, with reward being (potentially) received after multiple steps then the environment is considered multi-step. The predominant multi-step environment used in Learning Classifier System (LCS) research is the 'Woods' environment (e.g. figure 1).



. (00)

. (00) . (00)

F (11) . (00)

T (01) . (00)

. (00)

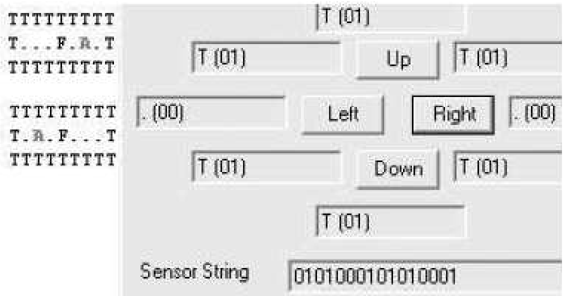
Sensor String 0000000000011100

Figure 1: Woods 1 environment (animat: A)

The Woods environments are cell based environments where each cell contains one of several objects which may have a reward associated with it; the AI agent (or animat) must navigate this environment to find the reward. In the most common woods environment there are 2 types of objects: T (tree) is an impassable object with zero reward, F (food) bestows a positive reward. Empty cells are indicated by a full stop and have zero associated reward. The animat has the ability to sense the objects in the surrounding 8 cells.

2.2 Markovian/non-Markovian

Within multi-step environments there is a distinct division between two types of environment, "If an environment has the Markov property, then its one-step dynamics enable us to predict the next state and expected next reward given the current state and action." (Sutton and Barto, 1998).



```
TTTTTTTTT
T...F.A.T
TTTTTTTTT

TTTTTTTTT
T.A.F...T
TTTTTTTTT
```

T (01)
T (01)
T (01)
. (00)
. (00)
T (01)
T (01)
T (01)

Sensor String 0101000101010001

Figure 2: Woods100, 2 positions with identical sensor strings

This is perhaps best illustrated with an example; in Woods1 (see figure 1) each possible position of the animat results in a unique sensor string (note that the environment wraps around on each side) and so after learning the animat can tell instantly which cell it is in. However in Woods 100 (see figure 2), there are two positions that give identical sensor readings, but with different subsequent states (and eventual rewards); this is therefore a POMDP environment.

2.3 Static/Dynamic

In the standard woods environment nothing changes over time; to test more advanced aspects of AI the environment can be made to be dynamic, for example the objects could move over time, or certain properties of objects could change. There is also the possibility of having a Multi-Agent System (MAS) which would introduce the possibilities of competition and collaboration. Initially the system will be tested in Static environments only, although it is hypothesized that dynamic environments would require the simulation of more complex emotions, such as apprehension (Rolls, 1999).

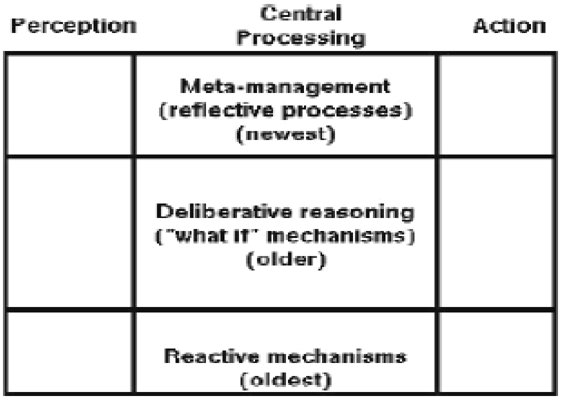
3 AI Architectures

3.1 Learning Classifier Systems

Learning Classifier Systems are rule-based evolutionary learning systems. The fact that they are rule-based gives them an advantage over some learning systems in that they have a high level of knowledge transparency, as the rules governing the LCS can be interpreted relatively easily. The original LCS (Holland, 1976) has been adapted and improved by various researchers, of key importance is XCS (Wilson, 1995), which (amongst other things) modifies Holland's LCS to use an accuracy-based fitness approach. There has also been relevant further work extending LCS to anticipate the effect of its actions, in the shape of Anticipatory Classifier Systems (ACS) (Butz, 2002). Wilson outlined both global and local explore/exploit strategies and concluded much further work is needed as they are crucial for true autonomy (Wilson, 1996).

3.2 Emotions in AI

In 'The Society of Mind' Minsky (1986) develops a theory of cognition which attempts to reduce the complexity of the human mind down to manageable and understandable modules which work together to allow for emergent behaviors to be produced.



Perception	Central Processing	Action
	Meta-management (reflective processes) (newest)	
	Deliberative reasoning ("what if" mechanisms) (older)	
	Reactive mechanisms (oldest)	

Figure 3: CogAff schema component grid

The CogAff schema (Sloman and Chrisley, 2004), see figure 3, is a method used to describe AI in a generic framework which can be considered as separating intelligence into 3 levels. It has also been shown in Sloman and Chrisley (2004) how this framework can be used to explain three different types of emotions.

3.3 Hybrid Emotional Classifier System

The proposed system is a combination of a standard implementation of XCS (TSI Artificial Intelligence) and the novel 'emotional' explore/exploit strategy. It supports the principle behind reinforcement learning in that an intelligent agent should learn from the interaction with its environment, and not from an explicit teacher. This leads to a Hybrid Emotional Classifier System (HECS).

4 Emotional Architecture

The proposed emotional architecture for HECS bases the emotional level of the animat on three factors (accuracy, reward and time), whilst storing the emotional level as two separate real variables ranging from -1 to 1 (with 1 being positive emotion). The equations proposed introduce several nonlinearities to the system, see figure 4. The precedence for the use of non-linear equations in LCS can be seen in their use for several other calculations, such as accuracy calculation, particularly where a sharp discrimination is required between similar inputs.

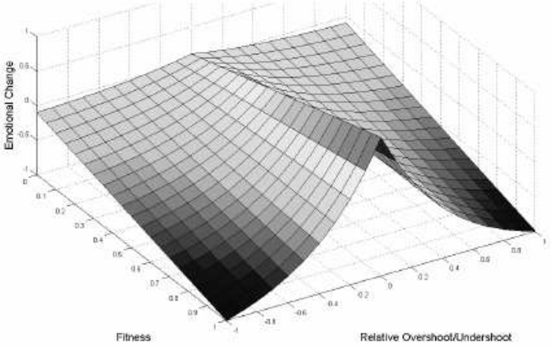


Figure 4: Graph of emotional change based on prediction fitness and overshoot/undershoot.

4.1 Accuracy Induced Emotion

The proposed emotional framework takes into account the accuracy of the predictions made by the animat, this calculation is based on the fitness of the prediction (how confident the animat is in the accuracy of its prediction) and the amount of overshoot/undershoot of the prediction (relative to maximum reward/punishment):

$$\Delta E_A = f(F) \times [g(M_{PA}) \times (e^{-|O_S| \times M_{PA}} - 1) + 1] \quad (1)$$

The function $f(F)$ is dependent on the fitness of the prediction (F) and a parameter that limits the maximum change in emotion (0-1).

$$f(F) = \Delta E_{Max} \times F \qquad (2)$$

MPA is a parameter which defines how close to 'perfect accuracy' a prediction has to be for a positive emotional increase (affects the gradient of the curve).

$$g(M_{PA}) = \frac{2}{1 - e^{M_{PA}}} \qquad (3)$$

O_S is the scaled overshoot:

$$O_S = \frac{O}{R_{range}} \qquad (4)$$

Where:

ΔE_{max} is the parameter which limits the maximum change in emotion (0-1).

F is the Fitness of prediction.

O is the Reward Overshoot, (positive value indicates reward was greater than prediction).

R_{range} is the animat's calculation of the range of rewards (max reward-max punishment).

M_{PA} is a parameter that defines the tolerance to perfect accuracy for a prediction to the reward obtained.

4.2 Reward Induced Emotion

When the animat receives immediate positive reward an emotional level is increased by an amount related to the amount of reward received relative to the maximum reward available, if the animat receives negative reward (or punishment) then the emotional level is decreased by an amount relative to the maximum punishment. The equation relating immediate reward to emotional change is as follows, and can be seen in figure 5.

$$\Delta E_R = h(M_{RS}) \times sign(R_S) \times (e^{|R_S| \times M_{RS}} - 1) \quad (5)$$

$$h(M_{RS}) = \Delta E_{\max} \times \frac{1}{e^{M_{RS}} - 1} \qquad (6)$$

Where:

R_S is the scaled reward/punishment.

M_{RS} is a scaling factor which affects the slope of the curve.

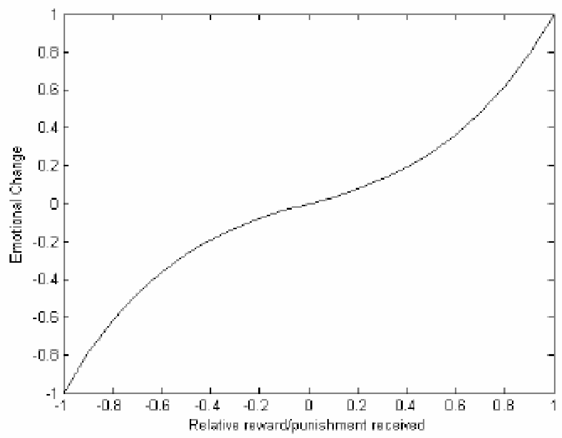


Figure 5: Graph of emotional change against immediate reward

4.3 Effect of Time on Emotions

The effect of reward on the animat's emotions is made to decay over time so that as time proceeds the animat becomes less satisfied with the reward it received in the past. The opposite occurs with negative reward so that the animat begins to forget the negative reward it received in the past. This effect of decay has been characterized in psychological studies, and is known as habituation. Habituation is an important mechanism in memory and attention, as reviewed by Cowan (1988, 1997). It occurs when a stimulus is repeatedly presented to a subject: the allocation of attention to that stimulus decays over time. As has been seen, emotional processes play an important role in attention and memory; it is thus postulated that a similar process of habituation takes place with emotions.

For positive rewards this decay is delayed until the animat has made more steps than it has estimated as being the maximum amount required to reach a reward (n_{max}) based on the minimum (P_{min}) and maximum (P_{max})predictions in the population, and the discount factor (γ) (see figure 6):

$$n_{\max} = \log_{\gamma}\left(\frac{P_{\min}}{P_{\max}}\right) \tag{7}$$

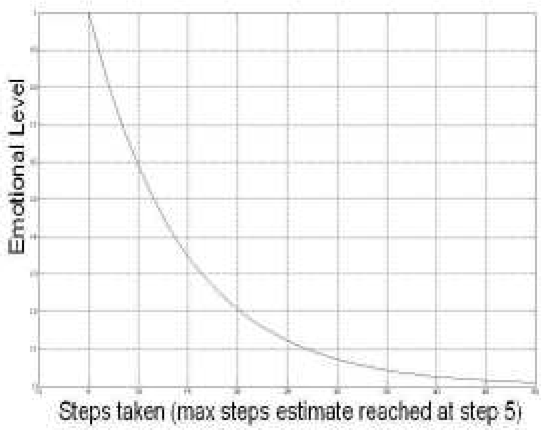


Figure 6: emotional decay level as steps taken passes maximum steps estimate

4.4 Effect of Emotions

The two emotional levels are used for two purposes: Firstly at the start of every trial the animat chooses whether to explore or exploit during this trial. This choice is based on the accuracy emotional level; if the animat is unhappy with its accuracy it should explore more so as to improve its knowledge (and therefore accuracy), if however the animat has been largely accurate then it can assume that the environment has been learnt to a sufficient level and therefore the best choice of action is to maximize the immediate reward by exploiting this knowledge.

$$P_{\exp lr} = \left(0.5 - \frac{0.9 \times E_A}{2}\right) \tag{8}$$

Note that in equation (8) the factor of 0.9 is an arbitrary value between 0 and 1 which means there is always a chance of both exploration or exploitation occurring (see figure 7).

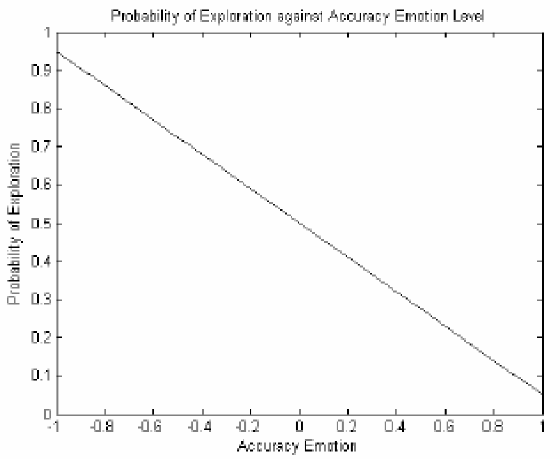


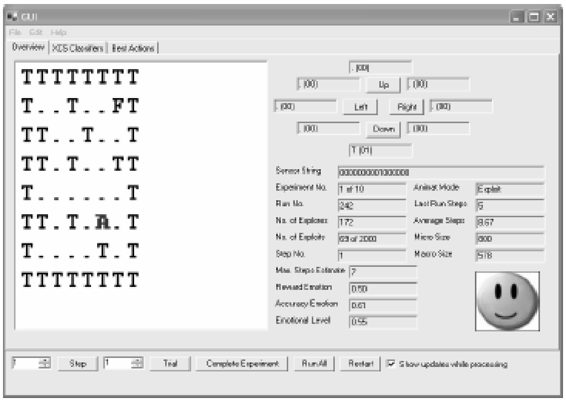
Figure 7: Graph showing the probability of the animat exploring at any given trial

Secondly the animat uses the reward induced emotional level to ‘escape’ from unsuccessful exploit trials; at the start of every exploit step the animat has the possibility of switching to explore mode if its emotional level is too close to zero; as the emotional level does not begin to decay until the estimated maximum number of steps have been taken there is little chance of switching until several exploit steps have failed to find a reward.

$$P_{SwitchToExplr} = 1 - |\, E_R \,| \tag{9}$$

5 Results

A screen capture of the HECS program, which was written in C#, is shown in figure 8. The reward emotion (E_R) and accuracy emotion (E_A) are combined to form an overall emotional level that acts as a guide to the user on the progress of training. An indication of generalization is given by comparing the Macro classifier size with the overall Micro classifier population size.



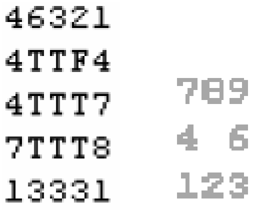
```
TTTTTTTT
T..T..FT
TT..T..T
TT.T..TT
T......T
TT.T.A.T
T....T.T
TTTTTTTT
```

Sensor String: 0000000001000000
Experiment No.: 1 of 10
Animat Mode: Exploit
Run No.: 242
Last Run Steps: 5
No. of Explores: 172
Average Steps: 8.67
No. of Exploits: 69 of 2000
Micro Size: 800
Step No.: 1
Macro Size: 578
Max. Steps Estimate: 7
Reward Emotion: 0.50
Accuracy Emotion: 0.61
Emotional Level: 0.55

Figure 8: HECS screenshot showing emotional exploration of Maze4

5.1 Woods XCS

The implementation of XCS in HECS was verified as it correctly reproduced known results in the Woods1 environment (TSI Artificial Intelligence). HECS successfully learnt the best actions for each of the available positions (see figure 9).

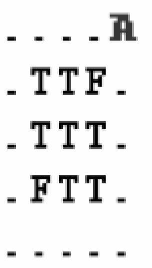


```
46321
4TTF4     789
4TTT7     4 6
7TTT8     123
13331
```

Figure 9: Woods1 (and key)

However, there appeared to be a slight problem with HECS in the woods1 environment; after some time fully general classifiers always appear, usually with the action of "up-right."

These classifiers tend to have low error and high fitness values. Initially this seemed likely to be an error in the code, however after some analysis it seems probable that this is an artefact of the combination of XCS with the Woods1 environment; The fully general classifiers have a low error because they predict a payoff after 2 further steps (P=810 if reward is 1000 and discount factor is 0.9), this is because for all positions a choice of "up right" does indeed leave you 2 steps away from the reward. The fitness of the classifier also remains high as fitness is measured relative to the action-set of the classifier (the set of all classifiers with matching condition and action), since the action "up right" is never the 'best' action to take there are no 'better' classifiers with which to compete in that action set (if you are going to go in that direction it does not matter where you are, it will always have the same reward as shown by the fully general classifiers). HECS still manages to choose the best actions, as the exploit algorithm takes a combination of fitness and predicted payoff, since the fully general classifiers have a low predicted payoff, they are always over-ruled by high fitness, high prediction classifiers. To test this theory a new environment was created (see Figure 10) where the up-right action is a good choice in certain positions, and not in others; when this new environment was tested there were no fully general classifiers in the final population.



```
....A
.TTF.
.TTT.
.FTT.
.....
```

Figure 10: Woods1b – the addition of food in the bottom left, makes 'up-right' a useful action at times

5.2 Woods1 – XCS + Emotional Exploration

Once the soundness of the basic XCS was established, the emotional side of the architecture was enabled and the results compared. Each experiment was run 10 times with the average values shown in table 1. The deviation between each run was not significant.

With emotions enabled the animat remained largely in explore mode for a period at the start, until it was happy' with its accuracy, at which point it switched mainly to the exploit mode with occasional exploration trials. This suggests that emotions enable the animat to decide when to exploit effectively.

Table 1: Woods1, effect of emotion on performance

Emotions	Off	On
Average Steps from start of training	2.792	2.727
Average Steps last 100 trials (optimum 1.6875)	1.708	1.725
Ave. No. Of Explores	2000	138
Actual Time taken (s)	402	65

Over the complete training cycle, 2000 exploit trials, the system with emotions took fewer steps, which may be important during online learning where resources need to be conserved. The average steps taken to reward at the end of training was close to the optimum value, which was confirmed as the optimal policy was present in both systems.

The number of explore trials was significantly reduced by the emotional explore/exploit strategy. This resulted in a significant timesaving for the overall algorithm.

The graph for macro population size, see figure 11, shows that with emotions turned on, HECS is still able to generalize, albeit less effectively than without emotions. The graph is somewhat misleading in terms of the x axis, as HECS uses far fewer exploration trials and achieves a level of generalization very close to that achieved with XCS after a greater number of exploration trials.

This domain is considered a simple environment, where simply alternating between exploring and exploiting is an effective strategy for XCS. HECS can learn the best actions for each state and reduce the number of explore trials. Generalization is not as good with emotions on, although it might be possible to add emotional pressure to generalize.

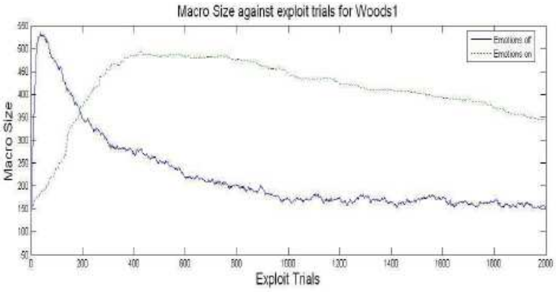


Figure 11: Woods1 – graph showing effect of emotions

5.3 Maze4

'Maze4' see figure 8, is also a static Markovian environment, but is a significantly larger environment than Woods1. This domain was used to test whether the explore/exploit benefits through the use of emotions could be scaled to larger environments.
HECS managed to greatly improve its performance by using emotional exploration as can be seen in the results, table 2, where both the number of explore trials and actual time taken were significantly reduced. The 'average test steps' is approximately the same. This environment is more complex than Woods1 due to the increase in size, increase in distance to reward, increase in maze density (Bagnall and Zatuchna, 2005), and less generalization being possible. HECS was also able to generalize to a similar degree with emotions on or off.

Table 2: Results for Maze4

Emotions	Off	On
Average Steps from start of training	8.18	4.74
Ave. No. Of Explores to optimum. Max emotional change = 1.0	2000	208
Actual Time taken (s)	1765	225
Ave. No. Of Explores to optimum. Max emotional change = 0.5	2000	3852

Exploration could be increased in HECS by reducing the effect of emotions, which shows that the explore/exploit strategy can be adapted to an environment. Despite this greater exploration, the time taken for the experiment was only slightly longer than XCS due to the emotions preventing the animat becoming stuck in long exploit trials, which occurred 147 times in XCS.

It is noted that similar levels of generalization were achieved partly due to the low levels of generalization possible, see figure 12. This figure also shows with emotions the system found a good strategy quickly, whereas without emotions the system kept trying alternative policies.

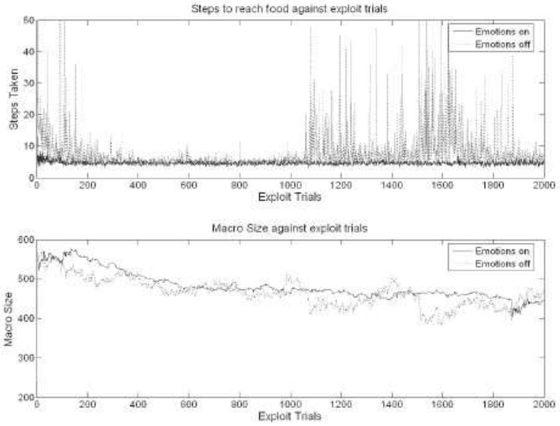


Figure 12: Graphs showing steps taken and macro population size for Maze4

5.4 Other Environments

HECS was tested in Woods100, as this environment is a POMDP. Standard XCS is unable to learn rules that give the best actions. Extensions to the standard XCS, such as memory, or a different LCS architecture, such as ACS, are required in such environments. Therefore, it is not surprising that HECS failed to evolve appropriate classifiers. Emotions could not make an improvement, so the animat was always 'unhappy' with its accuracy induced emotional state, and so very rarely exploited.

This does not indicate a problem with the implementation of emotions, instead it suggests that emotions could be used to change algorithmic methods or algorithm goals. It may allow a system to recognize that its environment is not a simple Markovian and adjust behaviours accordingly.

5.5 Effect of Emotion Parameters

The introduction of emotions has increased the number of parameters required for HECS to run; it is worth noting which of these parameters HECS is most sensitive to, and which parameters are very robust. The ‘Reward Slope Factor’, M_{RS} in equation (5) seems to be fairly robust to changes in value. Increasing the ‘Perfect Accuracy Weighting’, M_{PA} in equation (5), improved the generalization ability in Woods1, however the same increase in Maze4 caused HECS to do many more exploration trials, never reaching a satisfactory accuracy level to begin exploitation. This suggests that this parameter is not very robust in the current implementation, although perhaps it could be adapted in relation to the estimate of the environment size.

An interesting parameter is the factor which multiplies the accuracy induced emotional level in equation (8), currently set at 0.9. If this factor is removed then the animat performs well in terms of requiring less explore trials (in Woods1), however it fails to generalize very much, as in effect it finds a set of classifiers which are accurate and lead to a reward from every location, and so it stops exploring the environment and simply exploits its knowledge. Introducing this factor forces the animat to explore occasionally, and so causes the animat to continue to learn, whilst reducing the number of exploration trials required (compared to an unbiased explore/exploit choice).

6 Discussion

This explore/exploit strategy is novel as it uses nonlinear combinations of feedback from reward, prediction accuracy and temporal performance. Although there are similarities to the explore/exploit strategy suggested by Wilson (Wilson, 1996) nearly 10 years ago, the strategies have not been widely adopted. This is partly due to XCS without advanced explore/exploit strategies performing very well in terms of final 'average steps to reward' and rule generalization where appropriate. However, the novel autonomous explore/exploit strategy enabled HECS to significantly reduce the number of exploration steps (and hence time taken) in MDP remains, whereas XCS is stuck with its *a priori* explore/exploit balance. The algorithm also adjusted to the needs of the system as training progressed, with the balance weighted towards exploration at the start of training.

The use of a temporal measure based on the expectation of reward was successful in preventing loop 'exploits' trials, which were a problem for XCS. Setting a limit of 50 consecutive unsuccessful exploit trials in XCS would reduce the time taken, but requires domain knowledge.

When tested in a POMDP domain, both systems failed as they did not acquire the required structure. A different structure, such as ACS, could be triggered by the explore/exploit strategy, although this would be a sequential process. The use of memory has been proposed to improve performance in such problem domains. A modified strategy could allow HECS to consider its memory register only when considering a state where rules had low confidence of action.

Implementing a memory system could also help prevent resampling in two ways. When a message is presented to the system it could be compared with a list of known difficulties, so exploration can be triggered. Secondly, a set of past training examples may be kept in memory along with associated rewards. Memory could be used so that the system exploits information until an unknown area of search space is reached. The system would actively search for areas that need greater exploration.

Care must be taken not to autonomously tune one parameter by introducing two parameters that require tuning. In the case of this explore/exploit strategy, the parameters introduced are reasonably robust and can lead to significant time savings.

6.1 Emotions within HECS

Although 'emotions' were used as an analogy in HECS, it is worth considering how far this analogy can be taken. The emotions considered were instant (e.g. happy to have gained a reward) or predictive (e.g. happy as this action will lead to a reward), but not anticipatory (e.g. happy as I know the future will be better than the past). Temporal emotions have only been used in a simplistic manner by creating 'boredom' when stuck in exploit loops. “Emotional states, such as being anxious, apprehensive, relieved, and pleasantly surprised depend on deliberative capabilities in which plans can be created, inspected and executed.” (Holmes, 1997). In order to make the animat more deliberative, the range of temporal emotions must be expanded by including policy modelling. Once an internal world model of the domain is possible within an LCS, complex emotions due to the presence (or absence) of positive (or negative) reinforcers (Rolls, 1999) can be utilized to direct search.

Wright defines another set of emotions, which depend on the ability of the AI to deliberate on its own internal state, “Emotional states such as feeling humiliated, infatuated, guilty, or full of excited anticipation, in which attempts to focus attention on urgent or important tasks can be difficult or impossible, require an architecture with meta-deliberative control.” (Wright, 1997). The advantages of identifying an LCS as 'infatuated' are however highly debatable and approaches the realms of science fiction.

7 Conclusions

A novel explore/exploit strategy has been introduced that provides a dynamic choice for each trial and escapes from unsuccessful exploit trials. This was achieved through the analogy of emotions and results in behaviour that offers performance benefits where domain knowledge cannot be used to set up a standard XCS.

Although tested in a limited number of domains, HECS appears to scale well and have reasonably robust parameter settings. Further investigation is required; including how far the analogy of emotions used to guide LCS performance can be taken.

7.1 Future Work

It is desirable to test all possible explore/exploit strategies, based on feedback from the environment, in multiple types of domain. However, the motivation for HECS was to determine whether the simulation of emotion could be used to form a successful autonomous explore/exploit strategy. Further work needs to be conducted on how far this analogy may be taken.

Initially, there are many areas where HECS could be improved. At the moment the factor used in equation (4) is arbitrary. It would be interesting to see whether, instead of requiring this factor, the animat could be made to simulate ‘boredom’, so it could tire of getting reward all the time, and therefore decide to explore the world to see if there are any greater rewards available, or if there are better ways of getting to the rewards.

Resampling is a problem as an exploring LCS will consider all parts of the domain equally, including already successfully learnt parts. It may be possible for HECS to partially base the decision on the Match-Set classifiers and to therefore bias exploration to niches in the environment which have been insufficiently explored. It may also be possible to combine this with an ACS implementation to allow exploration to be directed towards future states which have been insufficiently explored.

The next stage in the development of HECS will be to implement an ACS module which will have greater deliberative abilities, and investigate further possibilities of emotional behaviour with this layer. After this has been completed a third intelligence module could be added with meta-deliberative abilities. The system architecture has been designed in a modular way so that any strategy or method can be implemented or modified independently.

Acknowledgements

The authors would also like to thank Jan Drugowitsch for his useful advice.

References

Bagnall, A.J. and Z.V. Zatuchna. On the classification of maze problems. In: *Foundations of Learning Classifier Systems*, Bull, L. and T. Kovacs (eds.), 307-316, Springer, 2005

Bush, G., P. Luu and M.I. Posner. Cognitive and emotional influences in anterior cingulate cortex. *Trends in cognitive sciences*, 4(6):215-222, 2000

Butz, M.V. Anticipatory Learning Classifier Systems. Boston, MA: Kluwer Academic Publishers, 2002

Butz, M.V. Rule-base evolutionary online learning systems: learning bounds, classification and prediction. University of Illinois, Illinois, PhD thesis, 2004

Cowan, N. Evolving Conceptions of Memory storage, selective attention, and their mutual constraints within the Human Information-Processing system. *Psychological Bulletin*, 104(2):163-191, 1988

Cowan, N. Attention and Memory: and integrated framework. Oxford: Oxford University Press, 1997

Fellous, J. From Human Emotions to Robot Emotions. *Proceedings AAAI 2004 Spring Symposium on Architectures for Modelling Emotion: Cross-Disciplinary Foundations*, 37-47, 2004

Holland, J.H. Adaptation. In: *Progress in Theoretical Biology*, Rosen, R. and F. Snell (eds.), 263-293, New York: Academic Press, 1976

Holmes, J.H. Discovering risk of Disease with a Learning Classifier System. *Proceedings 7th International Conference on Genetic Algorithms*, 426-433, Morgan Kaufmann, USA, 1997

Minsky, M. The Society of Mind. New York: Simon and Schuster, 1986

Rolls, E.T. The Brain and Emotion. Oxford: Oxford University Press, 1999

Scheutz, M. An Artificial Life Approach to the Study of Basic Emotions. *Proceedings CogSci*, 1203-1208, Chicago, August 5-7, 2004

Sloman, A. and R. Chrisley. More things than are dreamt of in your biology: Information Processing in Biologically-Inspired Robots. *Cognitive Systems Research*, 6(2):145-174, 2004

Squire, L.R., J.G. Ojemann, F.M. Miezin, S.E. Petersen, T.O. Videen and M.E. Raichle. Activation of the hippocampus in normal humans: a functional anatomical study of memory. *Proceedings of the National Academic Sciences*, 89:1837-1841, 1992

Sutton, R. and A. Barto. Reinforcement learning: An Introduction. Cambridge, MA: MIT Press, 1998

TSI Artificial Intelligence, Animat performance of ZCS, XCS and ACS. http://www.ai.tsi.lv

Whitley, L.D. Representation, Search and Learning. *Abstract of Proceedings of 3rd International Conference on Artificial Neural Networks and Genetic Algorithms (ICANNGA97)*, 624, Wein, New York, 1997

Wilson, S. Classifier fitness based on accuracy. *Evolutionary Computation*, 3(2):149-175, 1995

Wilson, S. Explore/Exploit strategies in Autonomy. *From Animals to Animats 4. Proceedings of the fourth international conference on Simulation of Adaptive Behaviour*, 325-332, Cape Cod, Massachusetts, 1996

Wright, I. Reinforcement learning and animat emotions. School of Computer Science, University of Birmingham, UK, Technical Report CSRP-96-4, 1996

Wright, I. Emotional Agents. School of Computer Science, University of Birmingham, Birmingham, UK, PhD thesis, 1997

Stability Analysis for the Stochastic best Particle Dynamics of a Continuous-time Particle Swarm Optimizer

K.Selvarajah[⋆†]
[⋆]Department of Automatic Control and Systems Engineering
University of Sheffield, UK
cop02ks@shef.ac.uk

V.Kadirkamanathan[†]
[†]Department of Automatic Control and Systems Engineering
University of Sheffield, UK
visakan@shef.ac.uk

P.J.Fleming[‡]
[‡]Department of Automatic Control and Systems Engineering
University of Sheffield, UK
p.flemiing@shef.ac.uk

Abstract

Previous stability analysis of the Particle Swarm Optimizer (PSO) were based on the assumption that all parameters are non-random, in effect a deterministic particle swarm optimizer. In this paper, we analyze the stability of the particle dynamics without this restrictive assumption using Lyapunov stability analysis and the concept of passive systems for the continuous time version of the particle Swarm optimizer. Sufficient conditions for stability are derived and an illustrative example given.

1 Introduction

Particle swarm optimization (PSO) is a Swarm intelligence technique developed by Eberhart and Kennedy (1995), inspired by social behavior of bird flocking and fish schooling. PSO is a population based search process where individuals, referred to as particles, initialize with a population of random solutions and search a potential solution to an optimization problem. Particles change their state by searching around in a multi-dimensional search space until a relatively unchanging state has been encountered or until computation limitations are exceeded (Ozcan and Mohan (1999)). PSO has been shown to be a very effective optimizer, especially in large convoluted search spaces (Eberhart and Shi (1998a)). Empirical evidence has accumulated that the algorithm is a useful tool for optimization (Shi and Eberhart (1999)). PSO has been applied to many optimization problems in real world (Eberhart and Hu (1999), Yoshida et al. (2000), Ciuprina et al. (2002)). On the algorithmic front, extensions have been made to deal with a dynamically changing environment in Eberhart and Hu (2002a). More recently, PSO has been successfully applied to multi-objective optimization problem (Eberhart and Hu (2002b), Parsopouls and Vrahatis (2002), CoelloCoello et al. (2004)) and artificial neural network training (Eberhart and Shi (1998a), van den Bergh (2002)). Additional operators have been incorporated into the basic particle swarm optimization scheme such as the selection operator in genetic algorithms in Angeline (1998) and neighborhood operator in Suganthan (1999). Owing to the similarity between a population of particles in swarm optimization and population of genotypes in genetic algorithms a comparison has resulted between the two in Eberhart and Shi (1998a). Recently, a hybrid of genetic algorithm and particle swarm optimization technique was applied to recurrent network design problem in Juang (2004).

The first analysis of the simplified particles behavior was carried out by Kennedy (1998) who showed the different particle trajectories for a range of design choices for the gain through simulations. In Ozcan and Mohan (1999), the authors showed that a particle in a simple one dimensional PSO system follows a path defined by a sinusoidal wave, randomly deciding on both its amplitude and frequency. The first formal analysis of the stability properties of the algorithm was carried out in Clerc and Kennedy (2002). Essentially, the analysis required the reduction of the standard stochastic PSO to a deterministic dynamical system by treating the random coefficients as constants. The resulting system was a second order linear dynamical system whose stability depended on the system poles or the eigenvalues of the state matrix. A similar analysis based on the deterministic version of the PSO was also carried out in identifying regions in the parameter space that guarantees stability (van den Bergh (2002)). The issue of convergence and parameter selection was also addressed in Eberhart and Shi (1998b), Trelea (2003). However, these do not take the stochastic nature of the algorithm into account.

Bacterial Swarm Optimization is another optimization method based on swarming behavior of bacteria, developed in Passino (2002). A stability analysis of social foraging swarms was carried out in Gavi and Passino (2004), where the authors used an individual-based continuous time model for the stability analysis. Here, authors analysed the stability properties of the collective behavior of the swarm for different profiles and provide conditions for collective convergence to more favorable regions of the profile. However, their functionality and dynamical equation are different to the PSO considered here.

In this paper, we provide a stability analysis of the continuous- time stochastic particle dynamics. The analysis is made feasible by representing the particle dynamics as a nonlinear feedback controlled system as formulated by Lure (Dosoer and Vidyasagar (1975), Vidyasagar (1993)). Such systems have a deterministic linear part and a nonlinear and/or time varying gain in the feedback path. It is well known that the stability of such feedback systems cannot be concluded by analyzing the stability of all possible linear feedback control systems with the nonlinear and/or time varying gain replaced by a linear one spanning the range of the nonlinear or time varying gain (Vidyasagar (1993)). Known as *Aizerman's conjecture*, its implication is that the stability conditions derived by treating the particle dynamics as deterministic, is not valid for the stochastic case in general.

The paper is organized as follows: In section 2, the basic continuous-time PSO algorithm is derived. In section 3, some characteristics of the particle dynamics are elucidated. In section 4, the main stability analysis result is derived. In section 5, an illustrative example is given followed by conclusions of the paper.

2 Particle Swarm Optimization

Particle swarm optimization (PSO) is a parallel evolutionary computation technique developed by Kennedy and Eberhart Eberhart and Kennedy (1995). The PSO formulation defines each particle as a potential solution to a problem in d-dimensional space with memory of its previous best position and the best position amongst all particles, in addition to a velocity component. At each iteration, the particles are combined to adjust the velocity along each dimension which in turn is used to compute the new particle position. Since each dimension is updated independently of others and the only link between the dimensions of the problem space are introduced via the objective functions, analysis can be carried out on the one-dimensional case without loss of generality. The original version was found to lack precision in the local search solution. This led to the introduction of an inertia factor in the velocity update in Eberhart and Shi (1998b), giving rise to the commonly used form of the PSO. The particle dynamics in one dimensional search space is given by,

$$\begin{aligned} v_{t+1} &= wv_t + \alpha_t^{(l)}(p^{(l)} - x_t) + \alpha_t^{(g)}(p^{(g)} - x_t) \\ x_{t+1} &= x_t + v_{t+1} \end{aligned} \qquad (1)$$

where v_t is the particle velocity at the tth iteration, x_t is the particle position at the tth iteration, $p^{(l)}$ is the best local position or the particle's best position thus far, $p^{(g)}$ is the best global position or the best solution amongst all particles, w is the inertia factor and $\alpha_t^{(l)} \sim \mathcal{U}[0, c_1]$, $\alpha_t^{(g)} \sim \mathcal{U}[0, c_2]$ are the random or stochastic parameters with uniform distributions.

The following statements can be derived from the particle dynamics of (1):

- The system dynamics is stochastic and is of order 2.
- The system does not have an equilibrium point if $p^{(g)} \neq p^{(l)}$.
- If $p^{(g)} = p^{(l)} = p$ is time invariant, there is a unique equilibrium point at $v_* = 0$, $x_* = p$.

Equilibrium point thus exists only for the best particle whose local best solution is the same as that of the global best solution. If asymptotic stability of the dynamics for the best particle can be guaranteed, then this particle is guaranteed reach the equilibrium point relating to the best solution. The analysis of the non-best particle is more challenging and is beyond the scope of this paper. Clearly, the conditions outlined for the existence of an equilibrium point does not hold true for any particle at all times in the particle swarm optimization. There are two points to be made with regard to this: Firstly, convergence to a fixed equilibrium point requires time invariance of the best solution position. Secondly, particles stop improving upon their locally best solution after a finite number of iterations so that beyond this point, the conditions can be deemed to hold.

We thus proceed to consider the particle dynamics associated with the best particle,

$$\begin{aligned} v_{t+1} &= wv_t + \alpha_t(p - x_t) && (2) \\ x_{t+1} &= x_t + v_{t+1} && (3) \end{aligned}$$

where $\alpha_t = \alpha_t^{(l)} + \alpha_t^{(g)}$. The combined stochastic parameter is no longer uniformly distributed but satisfies the following inequality:

$$0 < \alpha_t < K \tag{4}$$

where $K = c_1 + c_2$. Note that the use of (2) with p as a constant is not valid for non-best particle dynamics.The following expression used in Clerc and Kennedy (2002), Trelea (2003), for the deterministic PSO, given that

$$p = \frac{\alpha_t^{(l)} p^{(l)} + \alpha_t^{(g)} p^{(g)}}{\alpha_t} \tag{5}$$

is generally time varying if $p^{(l)} \neq p^{(g)}$ and $\alpha_t^{(l)}, \alpha_t^{(g)}$ are random.

By treating the random variable α_t as a constant, essentially a deterministic particle dynamics, the system dynamics is reduced to a simple time invariant linear second order dynamic model. Stability of such a deterministic particle dynamics can be concluded based on the eigenvalues of the state matrix in (9), as shown in Clerc and Kennedy (2002), van den Bergh (2002), Trelea (2003). The deterministic PSO model of Clerc and Kennedy has been analyzed assuming a continuous time process by considering a classical second-order differential equation Clerc and Kennedy (2002). As we shall see later in section 4, stability of the deterministic particle dynamics cannot be used to infer the stability of the stochastic particle dynamics. Further, noting that

$$x_{t+1} = x_t + wv_t + \alpha_t(p - x_t) \tag{6}$$

without loss of generality we can approximate the particle dynamics in a continuous time setting as follows,

$$\begin{aligned} \dot{x}_t &= wv_t + \alpha_t(p - x_t) && (7) \\ \dot{v}_t &= (w - 1)v_t + \alpha_t(p - x_t) && (8) \end{aligned}$$

Equations can be combined and written in the state space form ease of analysis as follows:

$$\begin{pmatrix} \dot{x}_t \\ \dot{v}_t \end{pmatrix} = \begin{pmatrix} 0 & w \\ 0 & w-1 \end{pmatrix} \begin{pmatrix} x_t \\ v_t \end{pmatrix} + \begin{pmatrix} \alpha_t \\ \alpha_t \end{pmatrix} (p - x_t) \tag{9}$$

3 System Characteristics

We note that the stability analysis of the particle dynamics is challenging with the stochastic or random component. It can be mapped to the problem of absolute stability of nonlinear feedback systems, known as Lure's stability problem. The stochastic particle dynamics is thus represented as a feedback controlled dynamic system as shown in Figure 1. The feedback control system representation depicts a time invariant linear plant in the forward path and an output control with time varying gain in the feedback path.

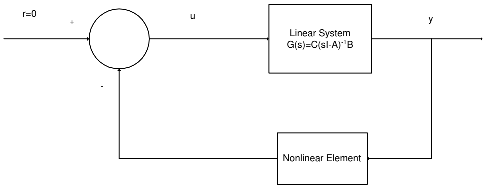


Figure 1: Feedback Control System Representation of Particle Dynamics

The equations governing the dynamics in this new representation can be expressed in the feedback form:

$$\begin{aligned} \begin{pmatrix} \dot{x}_t \\ \dot{v}_t \end{pmatrix} &= \begin{pmatrix} 0 & w \\ 0 & w-1 \end{pmatrix} \begin{pmatrix} x_t \\ v_t \end{pmatrix} + \begin{pmatrix} 1 \\ 1 \end{pmatrix} u_t && (10) \\ y_t &= \begin{pmatrix} 1 & 0 \end{pmatrix} \begin{pmatrix} x_t \\ v_t \end{pmatrix} && (11) \\ u_t &= -\alpha_t(y_t - p) && (12) \end{aligned}$$

where u_t is interpreted as the feedback control input signal. Under the conditions of p being time invariant, the dynamical system equation can be simplified further by introducing the state vector as follows:

$$\xi_t = \begin{pmatrix} x_t - p \\ v_t \end{pmatrix} \tag{13}$$

Defining $z_t = (y_t - p)$, the resulting state space representation is thus,

$$\dot{\xi}_t = A\xi_t + Bu_t \tag{14}$$

$$z_t = C\xi_t \tag{15}$$

$$u_t = -\alpha_t z_t \tag{16}$$

where the state matrix A, input matrix B and the output matrix C are given by,

$$A = \begin{pmatrix} 0 & w \\ 0 & w-1 \end{pmatrix}, B = \begin{pmatrix} 1 \\ 1 \end{pmatrix}, C = \begin{pmatrix} 1 & 0 \end{pmatrix} \tag{17}$$

The transfer function of the linear plant is then, $G(s) = C(sI - A)^{-1}B$.

The stability analysis method applied in the next section requires that the linear plant has a Hurwitz matrix A, i.e a matrix with eigenvalues strictly in the negative half circle in the complex-plane or equivalently $\lambda_i\{A\} < 0$ for all i. $\lambda_i\{\cdot\}$ represents the ith eigenvalues of A. Here, Noting that the feedback gain $\alpha_t > 0$, this problem can be circumvented by considering

$$A_\delta = A - \delta BC \tag{18}$$

Where $\delta > 0$ is arbitrarily small.

The state space model is adjusted as follows

$$\dot{\xi}_t = A_\delta \xi_t + Bu_t \tag{19}$$

$$z_t = C\xi_t \tag{20}$$

$$u_t = -(\alpha_t - \delta) z_t \tag{21}$$

where,

$$A_\delta = \begin{pmatrix} -\delta & w \\ -\delta & w-1 \end{pmatrix} \tag{22}$$

The resulting matrix A_δ can be shown to have eigenvalues in the open negative half plane of the complex plane.

The eigenvalues are the roots of the characteristic equation.

$$(\lambda I - A_\delta) = \begin{bmatrix} \lambda + \delta & -w \\ \delta & \lambda + 1 - w \end{bmatrix} \tag{23}$$

$$\mathbf{Det}(\lambda I - A) = \lambda^2 + (1 - w + \delta)\lambda + \delta \tag{24}$$

The roots of the characteristic equation λ_1,λ_2 are given by the solution of

$$\lambda_1 + \lambda_2 = -(1 - w + \delta) \tag{25}$$

$$\lambda_1 \lambda_2 = \delta \tag{26}$$

$Re\{\lambda_1, \lambda_2\} < 0$ if $w < 1$ and $\delta > 0$

The transfer function of the linear plant is given by

$$G(s) = C(sI - A_\delta)^{-1}B = \frac{s+1}{s^2 + (1 - w + \delta)s + \delta} \tag{27}$$

where s is the Laplace operator associated with the Laplace-Transform.

The following system properties can be derived readily:

- The particle dynamics specified in (14-16) has a unique equilibrium point at the origin in the ξ state space.
- The linear plant matrix A_δ is Hurwitz if $w < 1$ and is unstable if $w \geq 1$. Poles are also the eigenvalues of A_δ.
- The linear plant pair $\{A_\delta, B\}$ is controllable and pair $\{A_\delta, C\}$ is observable.
- The time varying memoryless feedback gain satisfies the so called sector condition $(\alpha_t - \delta) \in (0, K)$ and hence satisfies the following inequality:

$$u_t^2 + K u_t y_t \leq 0 \tag{28}$$

4 Stability Analysis

In this section, we use of the particle dynamics concept of passive systems and Lyapunov stability to analyse the stability of the particle dynamics. We begin this treatment by giving some basic concepts and their interpretations. A_δ, if its eigenvalues lie strictly in the open negative half circle of the complex plane or equivalently $\Re\{\lambda_i\{A_\delta\}\} < 0$ for all i. Here $\lambda_i\{\cdot\}$ represents the ith eigenvalues of A_δ and $\Re\{\cdot\}$ represents real part of eigenvalue.

A dynamical system is said to be *passive* if there is a non-negative scalar function $V(\xi)$ with $V(0) = 0$ which satisfies

$$\dot{V}(\xi_t) \leq y_t u_t \tag{29}$$

The equation above can be interpreted as the increase in stored energy is less than or equal to the energy input so that energy is lost in passive systems.

Lyapunov stability theorem states that with $\xi = 0$ being equilibrium point of the system, is asymptotically stable if there exists a non-negative scalar function $V(\xi)$ with $V(0) = 0$ which satisfies

$$\dot{V}(\xi_t) < 0 \tag{30}$$

Lyapunov stability analysis is based on the idea that if the total energy in the system continually decreases, then the system will asymptotically reach the zero energy state associated with an equilibrium point of the

system.
The passivity idea and the Lyapunov stability idea are combined to analyse the Lure stability problem whereby if all subsystems in a feedback system are passive, then the total energy can only decrease in an autonomous system (with zero input energy).

For linear systems, the passivity property can be related to a condition in the frequency domain known as positive real transfer functions.

The transfer function $H(s)$ of a linear dynamical system is said to be strictly positive real if and only if

- $\Re\{\lambda_i\{A_\delta\}\} < 0$ for all i, or equivalently A_δ has no eigenvalue in the right half-plane. Here $\lambda_i\{\cdot\}$ represents the i th eigenvalues of A_δ.
- $\Re\{G(j\varpi)\} \geq 0$ for all $\varpi \in (-\infty, \infty)$, where $\Re\{\cdot\}$ indicates the real part of its argument, $j = \sqrt{-1}$ is the imaginary number.

The transfer function $G(s)$ representing the linear plant in the particle dynamics is not a positive real transfer function. However, a lower limit for $\Re\{G(j\varpi)\}$ exists for vanishingly small δ and is given by,

$$\inf_{\varpi\in(-\infty,\infty)} \lim_{\delta\to 0} \Re\{G(j\varpi)\} > -\frac{w}{(1-w)^2} \qquad (31)$$

$$forall \quad \varpi \in (-\infty, \infty)$$

An important result that is necessary for the stability analysis is the continuous-time positive real lemma which links the concepts of positive real transfer functions and the existence of a Lyapunov function.

Positive Real Lemma can be stated as follows: Let $H(s) = C(sI - A_\delta)^{-1}B + D$ be a transfer function, where A_δ is Hurwitz matrix with$\{A_\delta, B\}$ being controllable, and $\{A_\delta, C\}$ being observable. Then $H(s)$ is strictly Positive Real if and only if there exist a symmetric positive definite matrix P, matrices W and L, and a positive constant ε such that (Khalil (1992)),

$$\begin{aligned} A_\delta^T P + P A_\delta &= -L^T L - \varepsilon P && (32)\\ B^T P &= C - W^T L && (33)\\ D + D^T &= W^T W && (34) \end{aligned}$$

Now we are ready to state the main result of this paper which specifies the conditions that must be satisfied by the design parameters w and K in order to guarantee the stability of the particle dynamics.

The particle dynamics specified in (14) is asymptotically stable if $w < 1$ and

$$K < \left(\frac{(1-w)^2}{w}\right) \qquad (35)$$

Consider the Lyapunov function

$$V(\xi_t) = \xi_t^T P \xi_t \qquad (36)$$

The decrease in the system energy as represented by the Lyapunov function at time t is given by,

$$\dot{V}(\xi_t) = \xi_t^T(PA_\delta + A_\delta^T P)\xi_t - 2\xi_t^T PB\alpha_t z_t \qquad (37)$$

Since $-2\alpha_t z_t(\alpha_t z_t - Kz_t) \geq 0$, if we add this component to the right-hand side of the equation, we get

$$\begin{aligned} \dot{V}(\xi_t) &\leq \xi_t^T(PA_\delta + A_\delta^T P)\xi_t - 2\xi_t^T \alpha_t z_t PB \\ &- 2\alpha_t z_t(\alpha_t z_t - Kz_t) && (38)\\ &= \xi_t^T(PA_\delta + A_\delta^T P)\xi_t \\ &+ 2\xi_t^T(KC^T - PB)\alpha_t z_t - 2(\alpha_t z_t)^2 && (39) \end{aligned}$$

We can show that the right-hand side is negative by completing a square term if the following matrix equations are satisfied

$$\begin{aligned} PA_\delta + A_\delta^T P &= -L^T L - \varepsilon P && (40)\\ PB &= KC^T - \sqrt{2}L^T && (41) \end{aligned}$$

Comparing these with the relationship established in the *Positive Real Lemma* above indicates that if and only if the linear system with the transfer function

$$\tilde{H}(s) = KC(sI - A_\delta)^{-1}B + 1 \qquad (42)$$

satisfies all the conditions stated in the positive real lemma, then (40)–(41) hold.

It is straight forward then to show that $\tilde{H}(s)$ satisfies the conditions in the Positive Real Lemma, if
A_δ Hurwitz
and

$$w < 1 \qquad (43)$$

and

$$K < \left(\frac{(1-w)^2}{w}\right) \qquad (44)$$

Then, substitute of (40) and (41)in (38).

$$\begin{aligned} \dot{V}(\xi_t) &\leq -\xi_t^T L^T L \xi_t + 2\sqrt{2}\alpha_t z_t \xi_t L^T \\ &- 2(\alpha_t z_t)^2 - \varepsilon \xi_t P \xi_t \\ &= -(L\xi_t - \sqrt{2}\alpha_t z_t)^T(L\xi_t - \sqrt{2}\alpha_t z_t) \\ &- \varepsilon \xi_t P \xi_t \\ &\leq -\varepsilon \xi_t^T P \xi_t && (45) \end{aligned}$$

Since the derivative in the Lyapunov function is non-increasing, the particle dynamics is guaranteed to be stable, according to the Lyapunov stability theorem.

In fact, asymptotic stability can be guaranteed by observing that when $\dot{V}_t = 0$, the particle dynamics is such that at the next time point, it will be non-zero – unless of course the particle has reached equilibrium.

Lyapunov stability results give only sufficient conditions and hence can be very conservative. Violation of the stability conditions do not imply instability – rather that stability cannot then be guaranteed.

The stability conditions derived above is illustrated graphically in Figure 2 which shows the maximum gain for a chosen inertia factor. Note that the maximum gain that guarantees the stability of particle dynamics decreases with the increase in inertia factor. This is in contrast to the results derived in van den Bergh (2002), Trelea (2003) under non-random constant gain assumptions where the maximum gain increased with the inertia factor.

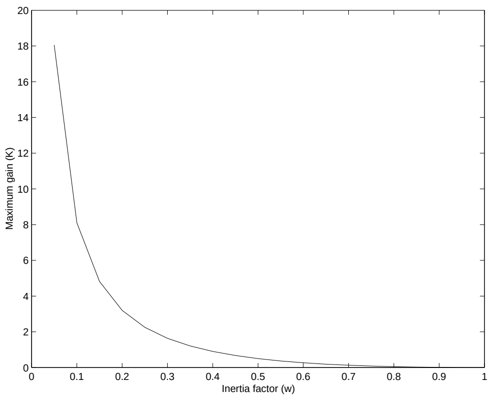


Figure 2: Maximum gain vs inertia factor for stability

5 Illustrative Examples

The stability analysis given in this paper can be interpreted in the frequency domain and time domain. Through an illustrative example, we demonstrate their utility and insight.

5.1 Nyquist Plot and Circle Criterion

The main stability theorem and the proof are based on the so called circle criterion, which can be used as a frequency domain graphical method for stability analysis. The results derived here is a special case when the lower limit for the feedback gain α_t is zero.

The circle criterion when applied to the stability of particle dynamics simply states that the Nyquist plot of the linear plant in the feedback system representation should lie to the right side of the point $-\frac{1}{K} + j0$ in the S-plane. This would then give a condition on K that is equivalent to that derived in the main theorem.

For the general particle dynamics as represented in (9), the Nyquist plots with the inertia factor (design parameter) $w = 0.8$ is given in Figure 3 and with $w = 0.2$ given in Figure 4. The dotted lines on the Figures show the condition for the positive realness,

$$\begin{array}{lll} -1/K & < -20 & \textbf{for} \quad w = 0.8 \\ -1/K & < -0.3125 & \textbf{for} \quad w = 0.2 \end{array}$$

The graphical results match those obtained from the results from the main theorem as expected.

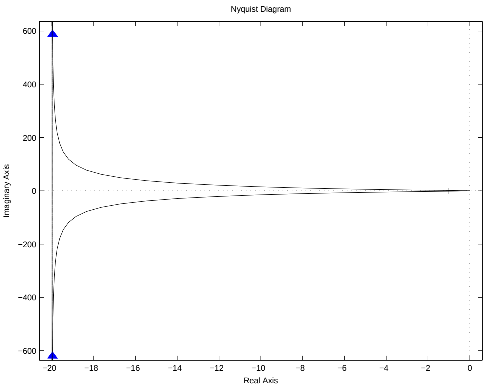


Figure 3: Continuous-time Nyquist plot for inertia factor=0.8 and the limit value for its real part

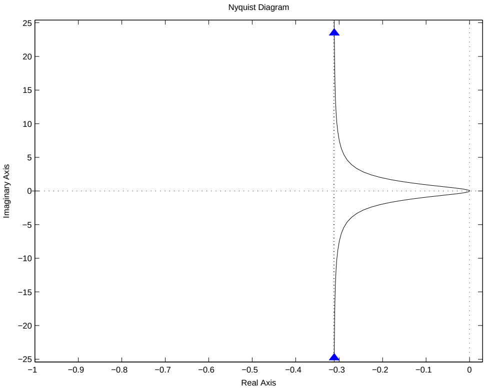


Figure 4: Continuous-time Nyquist plot for inertia factor=0.2 and the limit value for its real part

Note however, the circle criterion can be applied to general sector conditions such as $\alpha_{\min} \leq \alpha_t \leq \alpha_{\max}$

and thus provides us flexibility in designing further parameters.

5.2 Lyapunov Matrix and Particle Trajectories

The stability conditions derived here are based on Lyapunov stability analysis and hence are overly conservative. It is therefore important to analyse the impact on the particle dynamics of the choices for the design parameters. In particular, it is of interest to analyse the case when the derived stability conditions are violated.

First, we will determine a candidate positive definite matrix P in the Lyapunov function for the chosen inertia factor w. Consider the system with $w = 0.8$ then, the system state matrix A_δ can be considered as follows because the effect of δ in the calculation is negligible.

$$A = \begin{pmatrix} 0 & 0.8 \\ 0 & -0.2 \end{pmatrix} \tag{46}$$

For this case, stability requires $K < 0.05$. A choice of $K = 0.04$ is made for the analysis of this particle.

By solving for P from (40)–(41), the solutions are given by,

$$P_1 = \begin{pmatrix} 0.008 & 0.032 \\ 0.032 & 0.4542 \end{pmatrix}, P_2 = \begin{pmatrix} 0.008 & 0.032 \\ 0.032 & 0.2819 \end{pmatrix}$$

Likewise, for the system with $w = 0.2$, state matrix is

$$A = \begin{pmatrix} 0 & 0.2 \\ 0 & -0.8 \end{pmatrix} \tag{47}$$

For the stability guaranteeing choice of $K = 3$, the solutions for the positive definite matrix P are given by,

$$P_1' = \begin{pmatrix} 2.4 & 0.6 \\ 0.6 & 1.4 \end{pmatrix}, \quad P_2' = \begin{pmatrix} 2.4 & 0.6 \\ 0.6 & 0.6 \end{pmatrix}$$

The trajectory of the particle for the two cases above are also given in Figures 5 and 6 demonstrating asymptotic stability of the particle dynamics.

6 Conclusion

We have provided a stability analysis of the best particle of a continuous-time stochastic PSO. The passivity theorem and Lyapunov stability methods were applied to the particle dynamics in determining sufficient conditions for asymptotic stability and hence convergence to the equilibrium point. Since the results are based on the Lyapunov function approach, they are conservative and hence violation of these conditions do not imply instability. Illustrative examples were given to demonstrate the application of the technique. The commonly used PSOs are set in discrete-time and its stability analysis is currently under investigation, using similar approach.

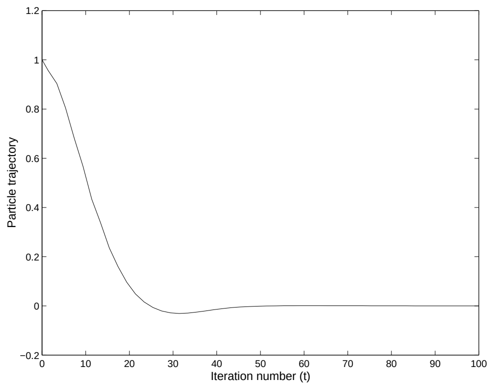


Figure 5: Particle trajectories with $K = 0.04$ and $w = 0.8$

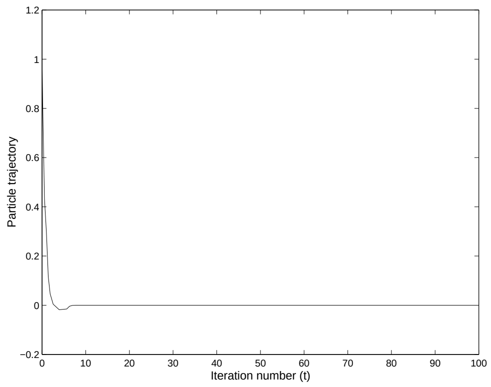


Figure 6: Particle trajectories with $K = 3$ and $w = 0.2$

References

P. J. Angeline. Using selection to improve particle swarm optimization. In *Proc. IEEE Congress on Evolutionary Computation*, Anchorage, Alaska, USA, 1998.

G. Ciuprina, D. Ioan, and I. Munteanu. Use of intellignet-particle swarm optimization in electromagnetics. *IEEE Trans. Magnetics*, 38(2):1037–1040, Mar 2002.

M. Clerc and J. Kennedy. The particle swarm-explosion, stability and convergence in a multidimensional complex space. *IEEE Trans. Evolutionary Computation*, 6:58–73, Feb 2002.

C. A. CoelloCoello, G. T. Pulido, and M. S. Lechuga. Handling multiple objectives with particle swarm optimization. *IEEE Trans. Evolutionary Computation*, 8(3):256–279, 2004.

C. A. Dosoer and M. Vidyasagar. *Feedback Systems: Input – Output Properties*. Electrical Science Series. Academic Press, 111 Fifth Avenue, New York, 1975.

R. C. Eberhart and X. Hu. Human tremor analysis using particle swarm optimization. In *Proc. IEEE Congress on Evolutionary Computation*, pages 1927–1930, Washington, DC, Mar 1999.

R. C. Eberhart and X. Hu. Adaptive particle swarm optimization: detection and response to dynamic systems. In *Proc. IEEE Congress on Evolutionary Computation*, pages 1666–1670, Hawaii, USA, May 2002a.

R. C. Eberhart and X. Hu. Multiobjective optimization using dyamic neighborhood particle swarm optimization. In *Proc. IEEE Congress on Evolutionary Computation*, pages 1677–1681, Hawaii, USA, May 2002b.

R. C. Eberhart and J. Kennedy. A new optimizer using particle swarm theory. In *Sixth International Symposium on Micromachine and Human Science*, volume 1, pages 39–43, Mar 1995.

R. C. Eberhart and Y. Shi. Comparision between genetic algorithms and particle swarm optimization. In *Proc. 7th Conference on Evolutionary Programming*, volume 1447, pages 611–616, Mar 1998a.

R. C. Eberhart and Y. Shi. Parameter selection in particle swarm optimization. In *Proc. 7th Conference on Evolutionary Programming*, volume 1447, pages 591–600, Mar 1998b.

V. Gavi and K. M. Passino. Stability analysis of social foraging swarms. *IEEE Trans. Systems Man and Cybernetics-Part B*, 34(1):539–557, 2004.

Chia-Feng Juang. A hybrid of genetic algorithm and particle swarm optimization for recurrent network design. *IEEE Trans. Systems Man and Cybernetics-Part B*, 34(2), 2004.

J. Kennedy. Particle behaviours. In *Proc. 7th Annual Conference on Evolutionary Programming*, pages 581–589, Mar 1998.

H. K. Khalil. *Nonlinear Systems*. Macmillan Publishing Company, 866 Third Avenue, New York, 1992.

E. Ozcan and C. K. Mohan. Surfing the waves. In *Proc. IEEE Congress on Evolutionary Computation*, 1999.

K. E. Parsopouls and M. N. Vrahatis. Particle swarm optimization method in multiobjective problems. In *Proc. ACM Symposium on Applied Computing (SAC) on Evolutionary Computation*, Madrid, Spain, May 2002.

K. M. Passino. Biomimicry of bacterial foraging for distributed optimozation and control. *IEEE Control Systems Magazine*, 22(3):52–67, 2002.

Y. Shi and R. C. Eberhart. Empirical study of particle swarm optimization. In *Proc. IEEE Congress on Evolutionary Computation*, pages 1945–1950, 1999.

P. N. Suganthan. Particle swarm optimizer with neighbourhood operator. In *Proc. IEEE Congress on Evolutionary Computation*, pages 1958–1962, Piscataway, NJ, USA, 1999.

I. C. Trelea. The pariticle swarm optimization algorithm: convergence analysis and parameter selection. *Information Processing Letters*, 85:317–325, Mar 2003.

F. van den Bergh. *An Analysis of Particle Swarm Optimizers*. PhD thesis, University of Pretoria, Dapartment of computer science, 2002.

M. Vidyasagar. *Nonlinear Systems Analysis*. Prentice-Hall, Englewood Cliffs, New Jersey, 1993.

H. Yoshida, K. Kawata, Y. Fukuyama, S. Takayama, and Y. Naknishi. A particle swarm optimization for reactive power and voltage control considering voltage security assessment. *IEEE Trans. Power Systems*, 15(4):1232–1239, Nov 2000.

Author Index