Set Reconciliation

Data Synchronization Problem:

\[ S_A \subseteq U \qquad S_B \subseteq U \]

**Goal:** Alice and Bob learn \( S_A \oplus S_B = (S_A \setminus S_B) \cup (S_B \setminus S_A) \)

- Well-studied problem: \( O(|S_A \oplus S_B|) \) communication cost
- Many applications e.g. data consistency in distributed databases

**Techniques:**
- Ordered Data: Error Correcting Codes
- Unordered Data: Invertible Bloom Lookup Table
Example:

\[ S_A = \{2, 43, 119, 321, 599\} \quad S_B = \{2, 44, 119, 222, 319\} \]

Sets can be reconciliated with communication cost \( O(|S_A \oplus S_B|) \)

Sets are very similar:
- Two exact matches: 2, 119
- Two almost matches: 43 \( \approx \) 44, 321 \( \approx \) 319
- One true difference: 599 \( \neq \) 222

Our Goal: Reconciliation that only considers the true differences with small communication cost
Synchronization of Image Databases

Difficulties:
- Same image, different encodings (bmp, jpeg, ...)
- In general: rounding errors, introduction of noise

Communication Cost Constraint:
Given a communication budget, reconcile as many true differences as possible
Robust Set Reconciliation

**Input:**
- Alice and Bob hold $S_A, S_B \subseteq [\Delta]^d$ on $d$-dim. grid of length $\Delta$
- Communication budget $k$

**Similarity measure:** Earth-Mover-Distance

$\text{EMD}(S_A, S_B) := \text{weight of minimum weight matching between } S_A \text{ and } S_B$

- $\text{EMD}(S_A, S_B)$ is the weight of the minimum weight matching between sets $S_A$ and $S_B$. This value represents the minimum cost of transforming one set into the other through a series of operations that preserve the structure of the sets. The matching is obtained by constructing a bipartite graph where nodes are the elements of $S_A$ and $S_B$, and edges are weighted by the distance between the corresponding elements. The goal is to find the matching with the minimum total weight.
Robust Set Reconciliation

**Input:**
- Alice and Bob hold $S_A, S_B \subseteq [\Delta]^d$ on $d$-dim. grid of length $\Delta$
- Communication budget $k$

**Similarity measure:** Earth-Mover-Distance

$\text{EMD}(S_A, S_B) := \text{weight of minimum weight matching between } S_A \text{ and } S_B$

$\text{EMD}(S_A, S_B) = \text{Sum of the lengths of the arrows}$
Robust Set Reconciliation

Input:
- Alice and Bob hold $S_A, S_B \subseteq [\Delta]^d$ on $d$-dim. grid of length $\Delta$
- Communication budget $k$

Similarity measure: Earth-Mover-Distance

$$\text{EMD}(S_A, S_B) := \text{weight of minimum weight matching between } S_A \text{ and } S_B$$

![Diagram of set reconciliation]

$$\text{EMD}(S_A, S_B) = \text{Sum of the lengths of the arrows}$$

Robust Set Reconciliation: Alice sends message $M$ to Bob with $|M| = \tilde{O}(k)$. Then Bob finds a set $S'_B$ so that $\text{EMD}(S_A, S'_B)$ is minimized
Optimal Solution

Communication budget limited by \( \tilde{O}(k) \):
We cannot expect to reconcile more than \( k \) point-pairs

\( k \)-residual EMD:

\[
\text{EMD}_k(S_A, S_B) := \min_{S_B^k} \text{EMD}(S_A, S_B^k),
\]

where \( S_B^k \) is obtained from \( S_B \) by relocating at most \( k \) points:

Our Goal: Approximation Scheme. Bob finds \( S'_B \) so that

\[
\text{EMD}(S_A, S'_B) \leq C \cdot \text{EMD}_k(S_A, S_B)
\]

"Remove the \( k \) heaviest edges"
Upper Bound: We have designed a one-way protocol with
- Communication Cost $O(kd \log(n\Delta^d) \log \Delta)$ so that
- Bob computes $S'_B$ and

$$\text{EMD}(S_A, S'_B) \leq O(d) \cdot \text{EMD}_k(S_A, S_B).$$

- The runtimes of both Alice and Bob is $O(dn \log \Delta)$.

Lower Bound: Any possibly randomized one-way communication protocol that computes an $O(1)$ approximation has communication cost

$$O(k \log(\Delta^d / k) \log \Delta).$$

→ For typical settings $d = O(1)$, $n = \Delta^{O(1)}$, $k = O(\Delta^{d-\epsilon})$ UB is tight

Experiments:
- Comparison to a baseline method that uses lossy compression
- Image reconciliation
Key Technique 1: Classical (One-way) Reconciliation

Ordered Data:

\[ u \in \mathcal{U}^n \quad \text{and} \quad \nu \in \mathcal{U}^n \]

There is a one-way protocol so that:
- Communication Cost is \( \tilde{O}(k) \),
- If \( d_H(u, \nu) \leq k \) then Bob can learn Alice’s input,
- If \( d_H(u, \nu) > k \) then Bob can report that \( d_H(u, \nu) > k \).

(\( d_H \): Hamming distance)

Technique:
- Forward Error Correction such as a Reed-Solomon code
- Invertible Bloom Lookup Table (near linear time for decoding/decoding)
Quad-trees:

- A layer corresponds to a resolution of the point set
- Alice and Bob construct quad-trees $T_A, T_B$ for their inputs $S_A, S_B$
- A layer of the difference tree ($T_A - T_B$) indicates “surplus” and “deficit cells”

**Correction given layer $L$ of Alice’s tree:**
Subtract this layer from own layer $L$ and do corrections as follows: Move points from surplus cells to center of deficit cells
Note: Additional error introduced since exact position is unknown
Key Technique 3: Random Shift

Let $M = (m_i)$ be a min-cost perfect matching between $S_A$ and $S_B$

**Interesting Layer:** Consider layer in difference tree ($T_A - T_B$) that reflects the $k$ heaviest edges of $M$ (Hamming distance $= \Theta(k)$)

**Technical Difficulty: False Positives**

→ Perform a random shift of the grid
Summary: Algorithm

Alice:
1. **Random Shift**: Alice shifts all points by u.a.r. chosen $\gamma$
2. **Build Quad-tree**
3. **Invertible Bloom Lookup Table**: For every layer $L$ of the quad-tree, build an IBLT that allows Bob to recover Alice’s layer $L$ if Bob’s layers $L$ differs by at most $ck$ (for a constant $c$)
4. **Send Message**: Alice sends $\gamma$ and the IBLT’s to Bob

Bob:
1. **Random Shift**
2. **Build Quad-tree**
3. **Decode IBLTs**: Bob decodes the IBLTs and determines the highest layer $L'$ so that Hamming distance is at most $ck$
4. **Move points**: Move points from surplus cells to deficit cells (center)
5. **Reverse Random Shift**

**Redundancy factor $c$**: Account for moving points to center of cells
Summary: Algorithm

- One-way two-party communication protocol for $O(d)$-approximation
- Algorithm cannot compute EMD nor residual EMD
- Computing EMD in one-way two-party communication model is a hard problem: constant approximation has communication cost polynomial in $\Delta$
One dimensional Experiment

- Alice’s point set: 1D data set with $n = 10^6$ points
- Inject $k = 100$ true differences by randomly picking $k$ points and moving them to an arbitrary location
- For all other nodes inject noise in $[-1, 1]$
- Baseline Method based on lossy Haar Wavelet Compression
Reconciliation of Image Database

Data Set:
- Alice has 10,000 high quality JPEG images
- Bob has a copy of this set which is modified as follows:
  - All images are recompressed with 95%-quality JPEG compression
  - $k$ images are replaced by different ones

Adaption of the Algorithm:
- Images are mapped to 6-dimensional feature space
- Algorithm adapted to two-way communication

<table>
<thead>
<tr>
<th></th>
<th>2%</th>
<th>4%</th>
<th>6%</th>
<th>8%</th>
<th>10%</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>0%</td>
<td>56%</td>
<td>92%</td>
<td>100%</td>
<td>100%</td>
</tr>
<tr>
<td>10</td>
<td>2%</td>
<td>34%</td>
<td>84%</td>
<td>100%</td>
<td>100%</td>
</tr>
<tr>
<td>$k$</td>
<td>0%</td>
<td>28%</td>
<td>80%</td>
<td>100%</td>
<td>100%</td>
</tr>
<tr>
<td>20</td>
<td>0%</td>
<td>19%</td>
<td>67%</td>
<td>98%</td>
<td>99%</td>
</tr>
<tr>
<td>25</td>
<td>0%</td>
<td>5%</td>
<td>66%</td>
<td>87%</td>
<td>99%</td>
</tr>
</tbody>
</table>

Table: Recovery rate for image reconciliation
Conclusion

Summary:
- Robust set reconciliation method that works well in practice
- Lower Bound illustrating that communication budget is almost tight

Open Questions:
- Can $O(d)$-approximation be improved? (e.g. $(1 + \epsilon)$-approx.)
- Improvement via multiple communication rounds?
Conclusion

Summary:
- Robust set reconciliation method that works well in practice
- Lower Bound illustrating that communication budget is almost tight

Open Questions:
- Can $O(d)$-approximation be improved? (e.g. $(1 + \epsilon)$-approx.)
- Improvement via multiple communication rounds?

Thank you for your attention.