Set reconciliation

Ivo Kubjas

Institute of Computer Science, University of Tartu

ivokub@ut.ee

The problem of set reconciliation arises in the systems for concurrent use of data.

Definition 1. The **set reconciliation problem** is defined as finding the union of sets with the smallest communication complexity.

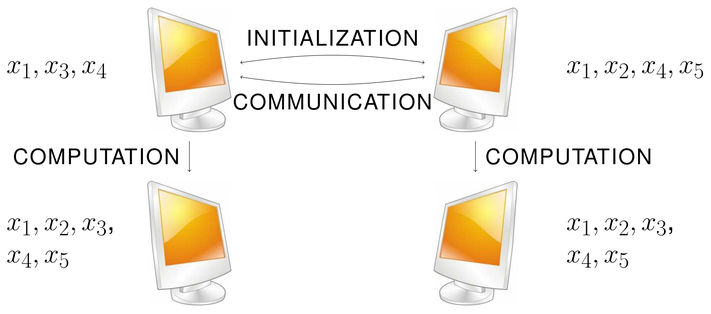


Figure 1: *Set reconciliation process*

- n_i - the number of items in i-th set
- n - the total number of items in union of the sets
- m - the number of items in the symmetric difference of the sets
- k - the number of parties.

Example 1. RSYNC *synchronization software compares the modification dates of the files between the client and server. This requires transmission of $O(n)$ symbols for producing the transmission queue.*

If the protocol stores the state of the file listings, then the clients can only send the differences. After initial communication, the communication complexity is $O(m)$. However, for every client the storage complexity is still $O(n)$, totalling in $O(kn)$.

In the environment where there are either large number of files or large number of servers, this is not desirable.

1. Set reconciliation protocols

1.1 Using characteristic polynomials [3]

Characteristic polynomials can be used for reconciling sets $S_A, S_B \subset \mathbb{F}^*$ what parties A and B are holding.

1: Evaluation points $E \in \mathbb{F}^*$ are known to A and B;
2: party A evaluates polynomial $\chi_{S_A}(Z) = \prod_{x \in S_A} (Z - x)$ at evaluation points E;
3: party B evaluates polynomial $\chi_{S_B}(Z) = \prod_{x \in S_B} (Z - x)$ at evaluation points E;
4: from

$$\frac{\chi_{S_A}(Z)}{\chi_{S_B}(Z)} = \frac{\chi_{S_A \cap S_B}(Z) \cdot \chi_{\Delta_A}(Z)}{\chi_{S_A \cap S_B}(Z) \cdot \chi_{\Delta_B}(Z)} = \frac{\chi_{\Delta_A}(Z)}{\chi_{\Delta_B}(Z)}$$

clients interpolate the rational polynomial $\frac{\chi_{\Delta_A}(Z)}{\chi_{\Delta_B}(Z)}$ and recover $\chi_{\Delta_A}(Z)$ and $\chi_{\Delta_B}(Z)$;
5: the roots of these polynomials correspond to the differences.

Algorithm 1: *Set reconciliation using characteristic polynomials*

The communication complexity of this protocol is $O(\overline{m} \log_2 |\mathbb{F}|)$, where $\overline{m}$ is an upper bound on the size of the symmetric difference. The computational complexity of this protocol is $O(\overline{m}^3)$.

1.2 Using invertible Bloom filters [1]

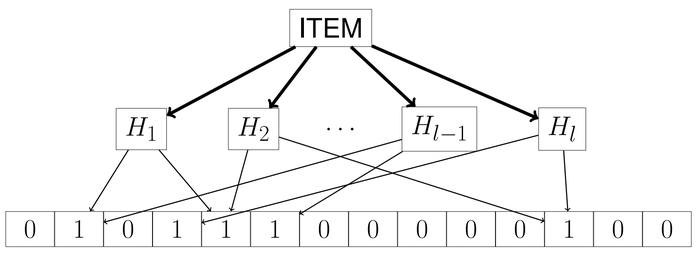


Figure 2: *Bloom filter*

A Bloom filter is a data structure for efficiently testing if an element belongs to a set.

An invertible Bloom filter is a data structure which allows to efficiently remove the elements from the set. In [1], invertible Bloom filters were modified to allow retrieval.

The communication complexity is $O(\overline{m})$. Experiments in [1] showed that using characteristic polynomials requires less overhead. The computational complexity is $O(n)$, which is dominated by hashing the files with several hash functions and computing the data structure. However, this can be done non-interactively while characteristic polynomial synchronization requires interaction on collision.

2. Higher-order networks

2.1 Three-party network

Hosts A, B and C are connected in a network which corresponds to a graph where B has degree 2 and other hosts have degree 1.

1: Hosts A and C evaluate $\chi_{S_A}(Z)$ and $\chi_{S_C}(Z)$ at $\overline{m}$ evaluation points E;
2: the evaluation values $\chi_{S_A}(E)$ and $\chi_{S_C}(E)$ and the sizes of the sets $|S_A|$, $|S_C|$ are sent to host B;
3: host B recovers $S_A \setminus S_B$, $S_B \setminus S_A$, $S_C \setminus S_B$ and $S_B \setminus S_C$;
4: host B recovers $S_A \cup S_B$ and $S_B \cup S_C$;
5: host B recovers $S_A \cup S_B \cup S_C$;
6: host B sends $S \setminus S_A$ to host A and $S \setminus S_C$ to host C;
7: hosts A and C recover $S_A \cup S_B \cup S_C$.

Algorithm 2: *A three-party protocol reconciliation algorithm using characteristic polynomials [2]*

Compared to naive protocol for reconciling items between three parties, this protocol requires sending about $1/3$ less information and doing 2 polynomial interpolations instead of 3.

The protocol can be extended to a larger star graph, where the central host computes the set differences and sends the missing items to the hosts.

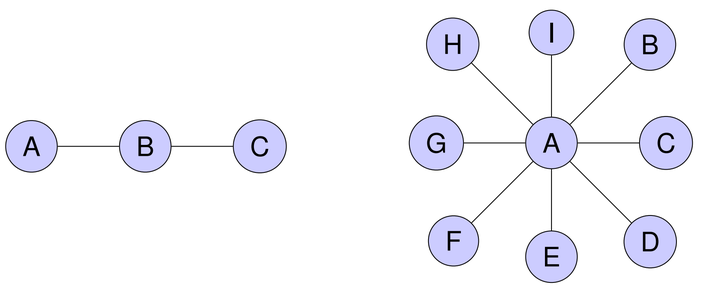
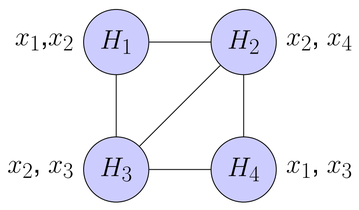


Figure 3: *Required network in Algorithm 2 and a star graph*

2.2 Broadcast network

A broadcast network allows to transmit messages to all parties. In [4], a method using linear coding was proposed to solve the data exchange problem.

We observed the connections between set reconciliation problem and data exchange problem. The method was modified for general networks.



$$L_1 = \begin{bmatrix} 1&0&0&0&0&0&0&0 \\ 0&1&0&0&0&0&0&0 \\ 0&0&1&0&0&0&0&0 \\ 0&0&0&1&0&0&0&0 \\ 0&0&0&0&1&0&0&0 \\ 0&0&0&0&0&1&0&0 \\ 0&0&0&0&0&0&0&0 \\ 0&0&0&0&0&0&0&0 \end{bmatrix}$$

$$A_1 = \begin{bmatrix} * & * & 0 & 0 \\ * & * & 0 & 0 \end{bmatrix} \qquad B_1 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}$$

Figure 4: *Data exchange problem modified for general network with edge between H_1 and H_4 missing. Examples of corresponding matrices for host H_1.*

Let A_i and B_i be $n_i \times n$-dimensional matrices describing the items each party has and L_i be matrices describing the connections between parties. Then $A = \begin{bmatrix} A_1 \\ A_2 \\ \vdots \\ A_k \end{bmatrix}$ and $\mathbb{A}$ is the family of all such matrices where stars in A are replaced with elements from $\mathbb{F}$.

Theorem 1. *If the network defined by connection matrices L_i, $i = 1, \ldots, k$, is 1-solvable then there exists a data exchange protocol with τ transmissions, where*

$$\tau = \min_{A \in \mathbb{A}} \operatorname{rank}(A)$$

for matrices A which are subject to

$$\forall i = 1, \ldots, k : \operatorname{rank}\left(\begin{bmatrix} L_i A \\ B_j \end{bmatrix}\right) = n, \forall j = 1, \ldots, k.$$

Example 2. *Theorem 1 applied to network in Figure 4 results in matrix A with rank $\tau = 3$*

$$A = \begin{bmatrix} 0&0&0&0&0&0&1&0 \\ 0&0&1&0&1&0&0&0 \\ 0&0&0&0&1&0&1&0 \\ 0&0&1&0&0&0&0&0 \end{bmatrix}^T.$$

Then H_2 has to broadcast $x_2 + x_4$, H_3 has to broadcast $x_2 + x_3$ and H_4 has to broadcast $x_1 + x_3$.

2.3 General unicast network

The model describes arbitrary networks where the goal is to decrease the number of reconciliation protocol invocations considering that each host can participate in only one pairwise reconciliation at a time. We test the algorithm based on using maximum weighted matching in the graph.

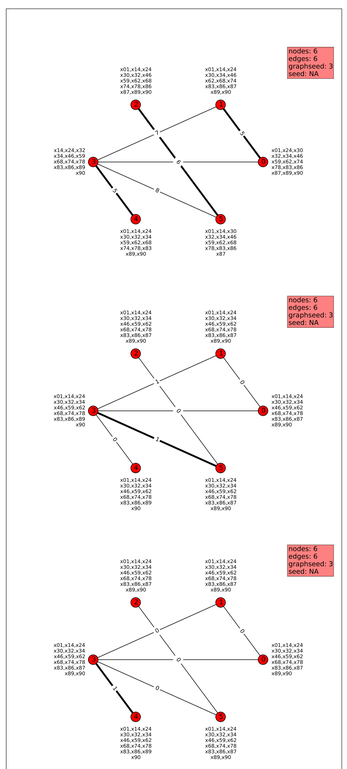


Figure 5: *Maximum matching in the network*

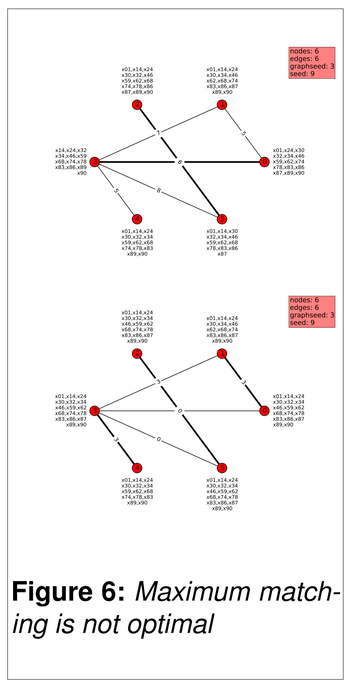


Figure 6: *Maximum matching is not optimal*

Maximum weight matching gave optimal results in $99,8\%$ of the tested graphs. It could be further studied to characterize the networks where it was not most efficient algorithm for choosing the parties which would reconcile their sets.

References

[1] David Eppstein, Michael T. Goodrich, Frank Uyeda, and George Varghese. What's the difference?: Efficient set reconciliation without prior context. *SIGCOMM Comput. Commun. Rev.*, 41(4):218–229, August 2011.

[2] Ivo Kubjas. Set reconciliation using rational polynomials. 2013.

[3] Y. Minsky, A. Trachtenberg, and R. Zippel. Set reconciliation with nearly optimal communication complexity. 2004.

[4] Salim Y. El Rouayheb, Alexander Sprintson, and Parastoo Sadeghi. On coding for cooperative data exchange. *CoRR*, abs/1002.1465, 2010.

norway grants

Eesti Teadusagentuur
Estonian Research Council

European Union
Regional Development Fund
Investing in your future

European Union
European Social Fund
Investing in your future

IT Akadeemia
toetab Skype™