Proof: Minimizing the Number of Transactions to Settle Debts is NP-Complete

Ammar Ahmad and Anand Kumar
Under Prof. Arijit Mondal

Problem Definition

Consider a set $P = \{P_1, P_2, \ldots, P_n\}$ of n participants. Each participant may either owe or be owed money by others. We define the debts as a directed weighted graph $G = (P, E)$, where:

- P represents the set of vertices (participants),
- $E \subseteq P \times P$ represents directed edges, where each edge $e_{ij} \in E$ has a weight $d_{ij} \geq 0$, indicating that P_i owes d_{ij} units of money to P_j [1].

Each participant P_i has a net balance b_i defined by:

$$b_i = \sum_{j=1}^{n} d_{ji}.$$

where d_{ji} represents incoming debt and d_{ij} represents outgoing debt. Our goal is to find a set of transactions that balances all debts (i.e., all net balances become zero) with the fewest number of transactions.

Decision Version of the Debt Minimization Problem: Given a set of net balances $B = \{b_1, b_2, \ldots, b_n\}$ such that $\sum_{i=1}^{n} b_i = 0$ and an integer k, determine if it is possible to settle all debts using at most k transactions.

Introduction

We aim to prove that the decision version of the Debt Minimization Problem is NP-complete by:

1. Showing that Debt Minimization Problem is in NP, meaning that a given solution can be verified in polynomial time.
2. Reducing a known NP-complete problem, the Maximum Zero-Sum Partition Problem (MZSP), to Debt Minimization Problem, demonstrating that Debt Minimization Problem is at least as hard as MZSP [2].

Step 1: Debt Minimization Problem is in NP

To demonstrate that Debt Minimization Problem is in NP, we need to show that, given a proposed solution (a set of transactions T), we can verify in polynomial time whether:

(i) The transactions in T settle all debts, resulting in a zero net balance for each participant.

(ii) The number of transactions is at most k.

Verification Procedure: Given a proposed solution $T = \{t_1, t_2, \ldots, t_m\}$, where each transaction t_i is a triple (P_x, P_y, δ) indicating that participant P_x transfers an amount δ to participant P_y, we verify the solution in the following steps:

1. **Initial Balance Calculation:** Let $b = (b_1, b_2, \ldots, b_n)$ represent the vector of initial net balances for each participant P_i, where:

$$b_i = \sum_{j=1}^{n} d_{ji}.$$

   This vector b is derived from the input debts d_{ij} and represents the initial state of all debts.

2. **Transaction Simulation and Balance Update:** For each transaction $t_i = (P_x, P_y, \delta) \in T$:

   - Update the net balances of P_x and P_y as follows:

$$b_x \leftarrow b_x - \delta \quad \text{and} \quad b_y \leftarrow b_y + \delta.$$

   - Ensure that $\delta \leq |b_x|$ at each step to avoid any participant P_x paying more than their net debt.

   Each transaction in T is processed in $O(1)$ time, and updating all transactions requires $O(m)$ time, where $m = |T|$.

3. **Final Balance Check:** After all transactions in T are applied, verify that the updated balance vector b satisfies:

$$b_i = 0 \quad \text{for all } i \in \{1, 2, \ldots, n\}.$$

   This ensures that all debts are settled. Verifying $b_i = 0$ for each participant requires $O(n)$ time.

4. **Transaction Count Verification:** Finally, check if the number of transactions $|T| = m$ satisfies $m \leq k$. This check is a single comparison operation, performed in $O(1)$ time.

Since each of these verification steps (initial balance calculation, transaction simulation, final balance check, and transaction count verification) can be completed in polynomial time relative to the number of participants and transactions, the overall verification procedure is polynomial in the size of the input. Therefore, Debt Minimization Problem is in NP.

Step 2: Reduction from Maximum Zero-Sum Partition Problem

To prove NP-hardness, we reduce the **Maximum Zero-Sum Partition Problem** (MZSP) to Debt Minimization Problem. The MZSP is defined as follows:

Maximum Zero-Sum Partition (MZSP)

Let $S = \{s_1, s_2, \ldots, s_n\}$ be a set of integers where $\sum_{i=1}^{n} s_i = 0$. The Maximum Zero-Sum Partition problem seeks a partition of S into disjoint subsets $S_1, S_2, \ldots, S_m \subseteq S$ such that:

$$\sum_{s \in S_j} s = 0 \quad \forall j = 1, 2, \ldots, m.$$

The objective is to maximize the number of such zero-sum subsets m. In other words, we aim to find the largest integer m for which there exists a collection of disjoint subsets of S, each having a total sum of zero.

The decision problem version of MZSP asks:

> *Given a set of integers S and an integer k, can S be partitioned into at least k zero-sum subsets?*

The MZSP is known to be NP-complete [3]. We will reduce MZSP to Debt Minimization Problem, demonstrating that solving Debt Minimization Problem allows us to solve MZSP, thereby proving that Debt Minimization Problem is NP-hard.

Reduction Construction

Given an instance of MZSP with a set S, construct an instance of Debt Minimization Problem as follows:

- Map each integer $s_i \in S$ to a participant P_i in Debt Minimization Problem.
- Define the net balance b_i of each participant P_i as s_i.
- Set the target number of transactions $k = m - p$, where m is the number of integers in S and p is the maximum number of zero-sum subsets in an optimal solution of MZSP.

Reduction Analysis

If the MZSP instance has a solution with p zero-sum subsets, then we can partition the participants in Debt Minimization Problem into groups corresponding to these subsets. Each subset can be settled with exactly $|S_i| - 1$ transactions. Therefore, the total number of transactions required is:

$$k = \sum_{i=1}^{p} (|S_i| - 1) = m - p$$

Thus, a solution to MZSP with p zero-sum subsets corresponds to a solution of Debt Minimization Problem with at most $k = m - p$ transactions, and vice versa. This reduction is polynomial in the size of the input, establishing that Debt Minimization Problem is NP-hard.

Conclusion

We have shown that Debt Minimization Problem is in NP and is NP-hard by reduction from the Maximum Zero-Sum Partition Problem. Therefore, Debt Minimization Problem is NP-complete.

References

[1] Verhoeff, T. (2003). *Settling Multiple Debts Efficiently: An Invitation to Computing Science*. Faculty of Mathematics and Computing Science, Eindhoven University of Technology.

[2] Garey, M. R., and Johnson, D. S. (1979). *Computers and Intractability: A Guide to the Theory of NP-Completeness*. W.H. Freeman and Company.

[3] Fertin, G., Fontaine, O., Jean, G., and Vialette, S. *The Maximum Zero-Sum Partition Problem.* Journal of Discrete Algorithms, 2020.