Exploring the Resolution of Delegations in Liquid Democracy with Fractional Delegation

David Holzwarth
Advisor: Prof. Dr. Bryan Ford
Chair of Computer Systems
https://dse.in.tum.de/

08.04.2025– 05.08.2025

Structure

- Introduction
- Problem Statement
- Design
- Implementations
- Evaluation

Liquid Democracy: Introduction & Motivation

Liquid Democracy – Fractional Delegation

- Currently: one person, one delegation
- Fractional delegation: one person, many delegations

- Motivation:
  - Less concentration of voting power
  - Less loss of voting power
  - Empowers voters

- Drawbacks:*
  - Less intuitive
  - Not computationally trivial

Problem statement

- Given:
  - Voters who vote directly
  - Voters who delegate their vote
- Find:
  - Each voter‘s final voting power according to the delegations
  - Power must be conserved*

- Challenge: cyclic delegations & efficient computation

Design

- Definition: Well-formed delegation graph:
  - 1. Delegation graph with 2. no closed delegation cycles

1. $V = S \dot{\cup} D$, meaning that V is the union of the two *disjoint* sets of sinks and delegators.
2. Each edge $e \in E$ is a triple (u, v, w) denoting a delegation from node u to node v of weight w.
3. Each sink $s \in S$ has no outgoing edges.
4. Each delegator $d \in D$ has $n \in \mathbb{N}$ outgoing edges, each with a positive weight, such that the sum of all of its outgoing edge weights equals 1.

We define a **closed delegation cycle** $C \subseteq V$ in a delegation graph $G = (S \dot{\cup} D, E)$ as a cycle in G such that for every node $v \in C$, there exists no path from v to any sink node in S.

Design: Resolving Delegations

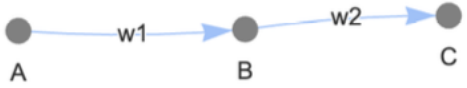
- For all nodes v:

$$p'_v = 1 + \sum_{(u,v,w)\in E} w p'_u$$

$$p_v = \begin{cases} p'_v & \text{if } v \in S \\ 0 & \text{if } v \in D \end{cases}$$

- System of linear equations

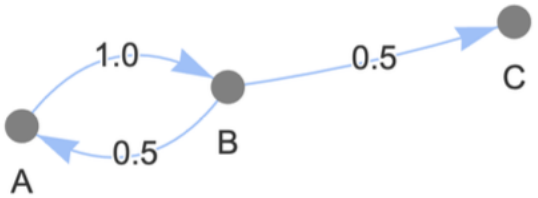
- A well-formed delegation graph:
  - Has exactly one solution,
  - which conserves power



$$\begin{aligned} p_C &= 1 + w_2 p'_B \\ &= 1 + w_2(1 + w_1 p'_A) \\ &= 1 + w_2(1 + w_1 \cdot 1) \end{aligned}$$

Implementations

- Linear Systems Solver
- Linear Programming Solver
- Iterative Solver
  - Provably similar to the other approaches

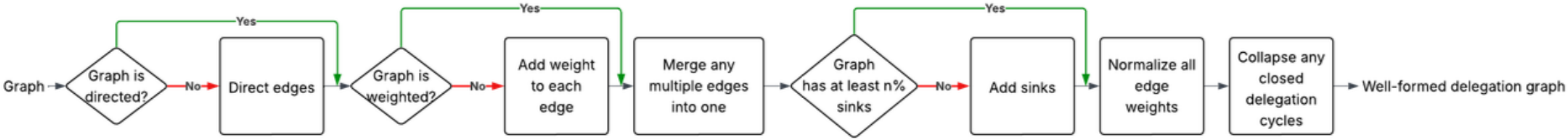


Implementations: Robustness

	Invalid Delegations	Closed Delegation Cycles
Linear Systems Solver	Invalid solution	Error
Linear Programming Solver	Invalid solution	Error
Iterative Solver	Depends	Won’t terminate

Evaluation: Method

- Preprocessing

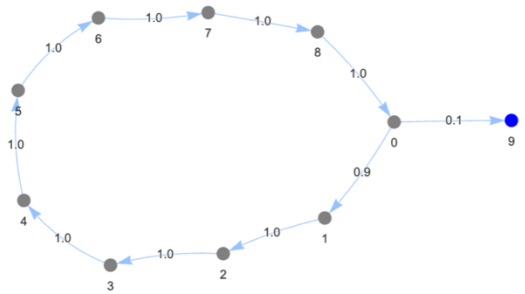


  - *The „Add sinks“ step is only for experimental purposes, to have the option to avoid graphs with no sinks for benchmarks*

- Measurement
  - of solving time only

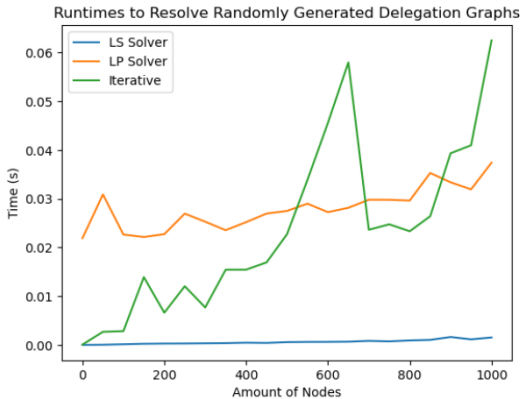
Evaluation: Overview

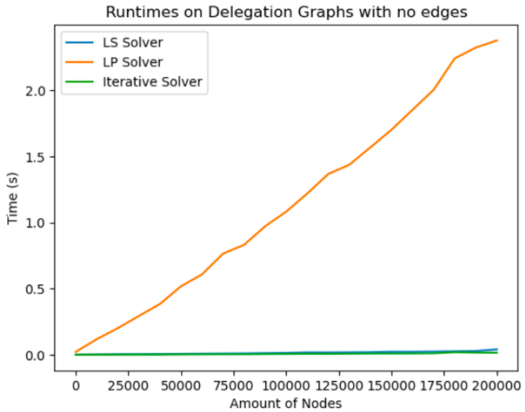
- Synthetic graphs
  - Small graphs
  - Large graphs
  - Dense graphs
  - Cycles which retain a lot of their power
  - No delegations
- Synthetic Social graphs
  - Small world graphs
  - R-Mat graphs
- Real-World Datasets
  - Bitcoin OTC trust network, Epinions, Slashdot Zoo

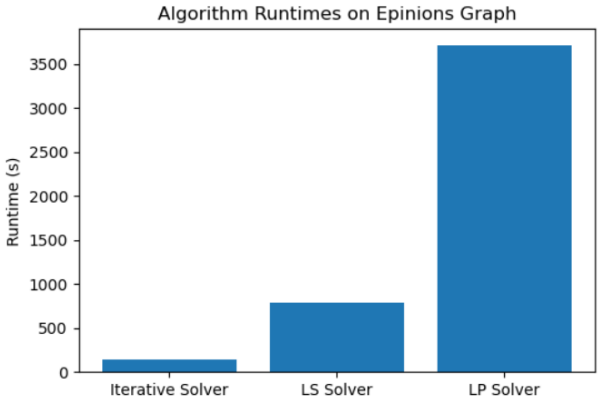


Evaluation: Key Insights

- Small graphs:
  - Linear Systems Solver
- Very sparse, very large graphs:
  - Iterative Solver (not perfectly precise!)
  - Linear Programming Solver‘s runtime grows slower, but it is slower



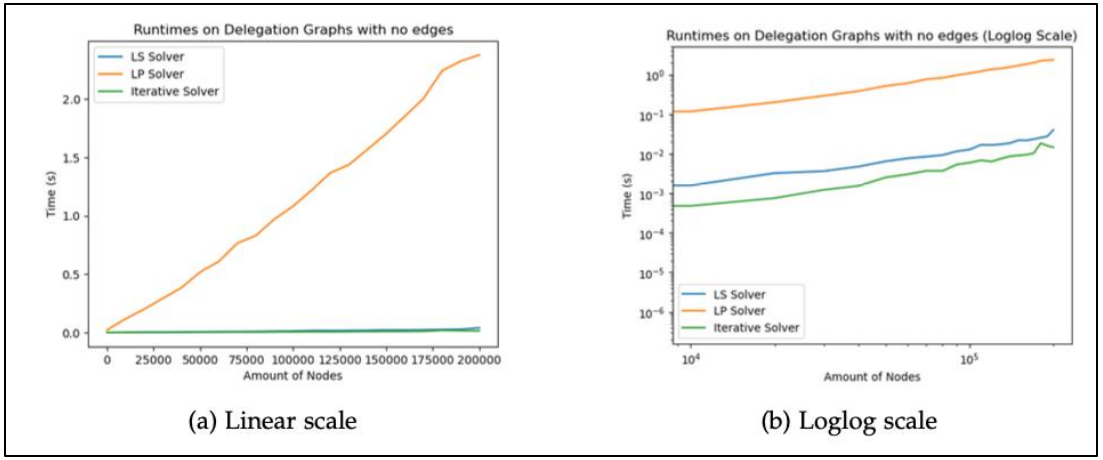




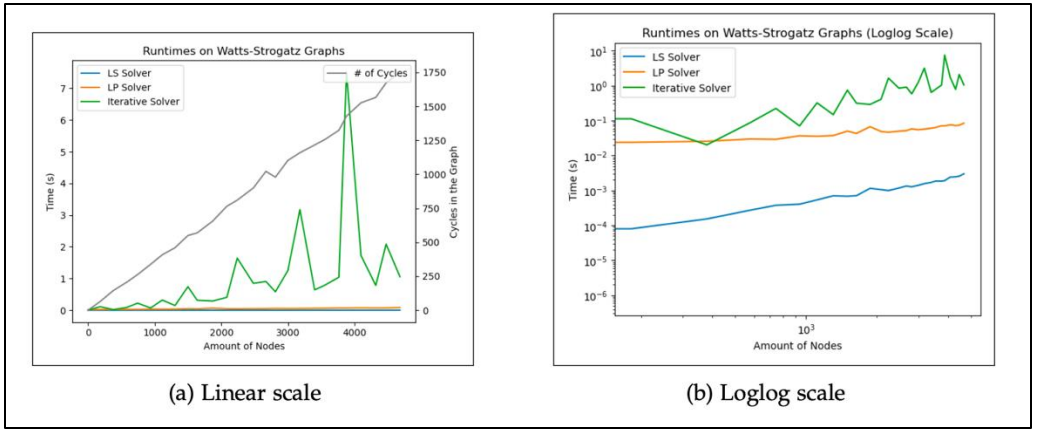
For the Future

- Does fractional delegation actually reduce vote power concentration?
- Benchmarking on real world data
- User study:
  - How easy is it to understand Liquid Democracy with fractional delegation?
  - Do / How do people delegate?

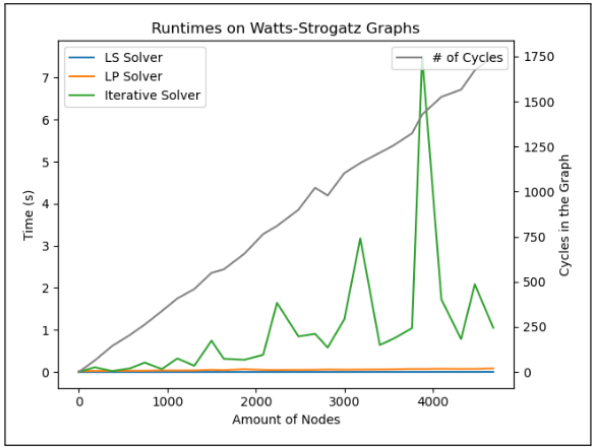
Thank you!



(a) Linear scale (b) Loglog scale

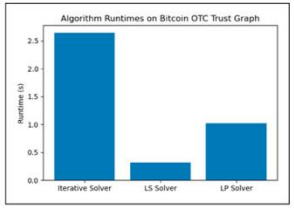


(a) Linear scale (b) Loglog scale

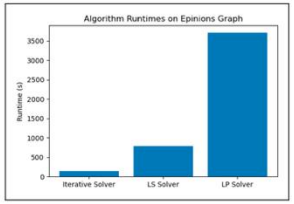


(a) Linear scale

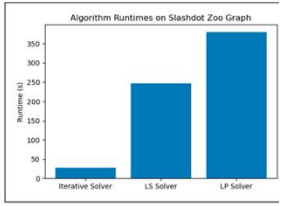
Thank you!



(a) Bitcoin OTC dataset

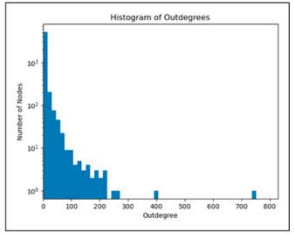


(b) Epinions dataset

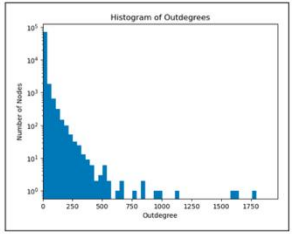


(c) Slashdot Zoo dataset

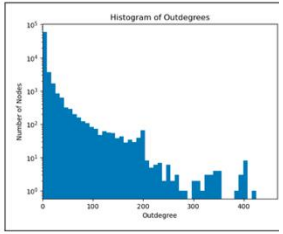
Figure 5.17.: Runtimes



(a) Bitcoin OTC dataset

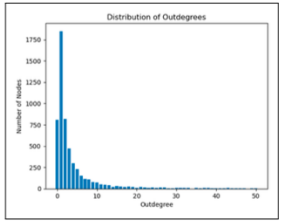


(b) Epinions dataset

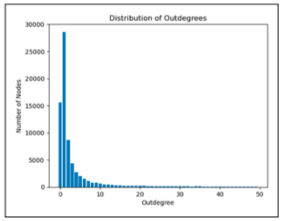


(c) Slashdot Zoo dataset

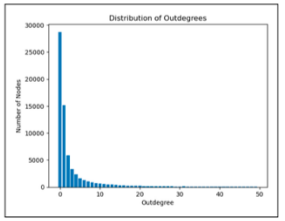
Figure 5.18.: Histogram of outdegrees



(a) Bitcoin OTC dataset

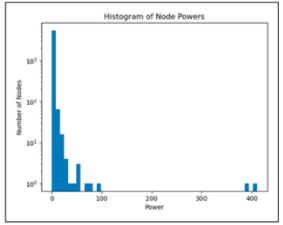


(b) Epinions dataset

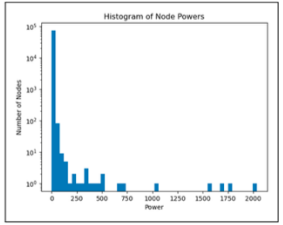


(c) Slashdot Zoo dataset

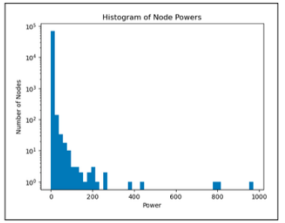
Figure 5.19.: Distribution of outdegrees between 0 and 50



(a) Bitcoin OTC dataset



(b) Epinions dataset



(c) Slashdot Zoo dataset

Figure 5.20.: Distribution of powers