Keeping Peers Honest In EigenTrust

Robert McGrew

Joint work with Zoë Abrams and Serge Plotkin

Roadmap

- Incentive problems in P2P systems
- Trust systems and manipulation
- Requirements for a truthful trust system
- A negative result
- A positive result
  - two truthful trust systems based on EigenTrust

P2P Systems Incentive Problems

- P2P systems face two kinds of problems
- Malicious peers
  - RIAA on Kazaa
  - A malicious collective answering queries with fake files
- Free-riders
  - Most other users on Kazaa
  - Individual users selfishly maximizing their own utility by downloading without sharing

P2P Trust Systems

- Attempt to solve malicious peer and free-rider problems
- Peers make recommendations based on successful downloads received
- Recommendations are used to calculate a trust score
- Trusted peers are chosen to serve files
- Trusted peers are rewarded with better quality of service

Keeping Them Honest

- Trust systems resolve the malicious peers problem and the free-rider problem
- Are we done?

Keeping Them Honest

- Trust systems resolve the malicious peers problem and the free-rider problem
- Are we done?
- Unfortunately, honest peers that previously had no reason to cheat now will lie to improve their trust
- Recommendations won’t reflect downloads
- Our contribution: we seek a *truthful* trust system

EigenTrust (Kamvar, Schlosser, Garcia-Molina 03)

- Recommendations form a directed graph
- Calculates PageRank on recommendation graph
- Recommendations from trusted nodes are worth more
- Malicious peers who recommend each other still won’t receive high trust scores

Malicious and Selfish Agents

- Malicious agents attempt to spread bad files
  - EigenTrust alienates these peers in simulations
  - Our new trust system should retain this property
- Selfish agents seek to maximize own utility
  - EigenTrust makes selfish agents wish to maximize their trust by sharing many files
  - Selfish agents now also lie about downloads
  - Trust system should make selfish agents truthful

EigenTrust Model

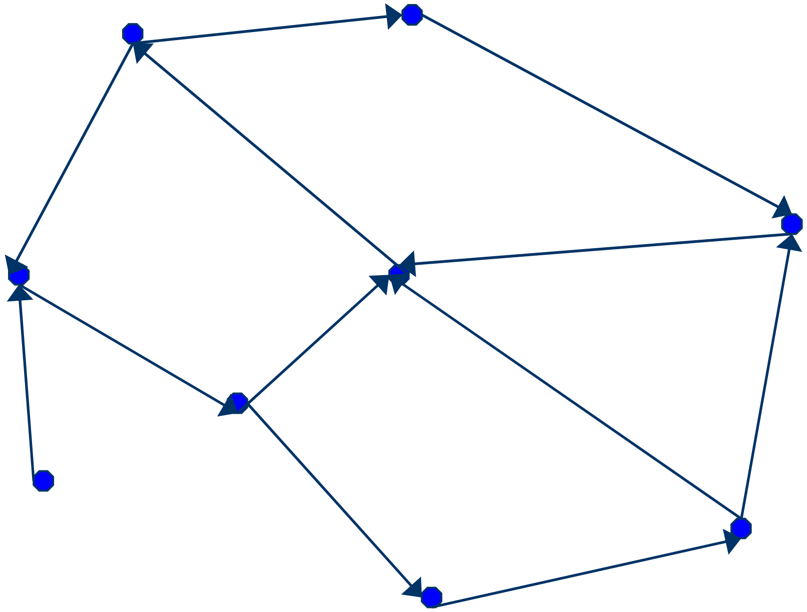
- Time is divided into rounds
- During the round
  - each agent makes queries q 5 Q_i
  - servers are chosen based on their trust scores
- At end of round
  - agents report downloads d_i to center
  - center calculates trust score $T_i(d_i, d_{-i})$ for next round

EigenTrust Algorithm

- Trust is defined as the stable distribution over a “random surfer” Markov chain
  - Each recommendation by a peer is a probability distribution over nodes it has downloaded from
  - Follow links or teleport *a la* PageRank

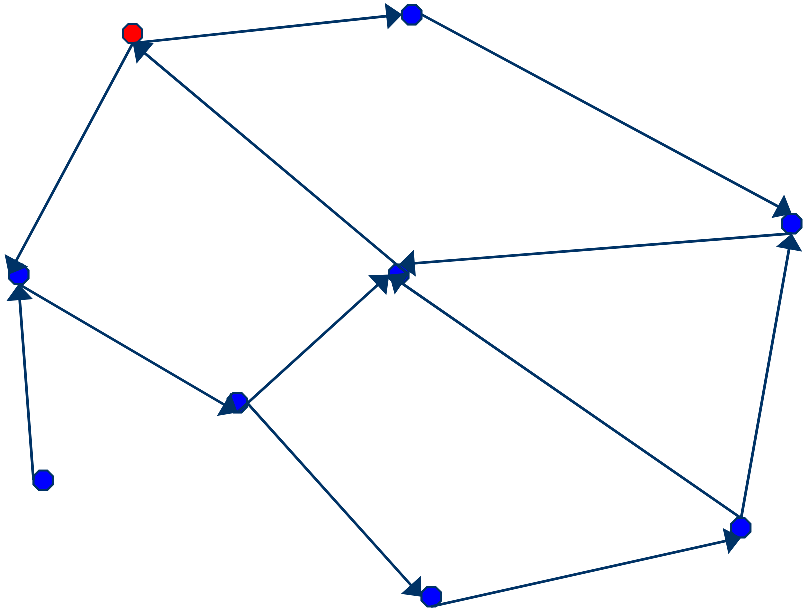
EigenTrust Algorithm

- Consider the Markov chain defined as random walk over the download graph



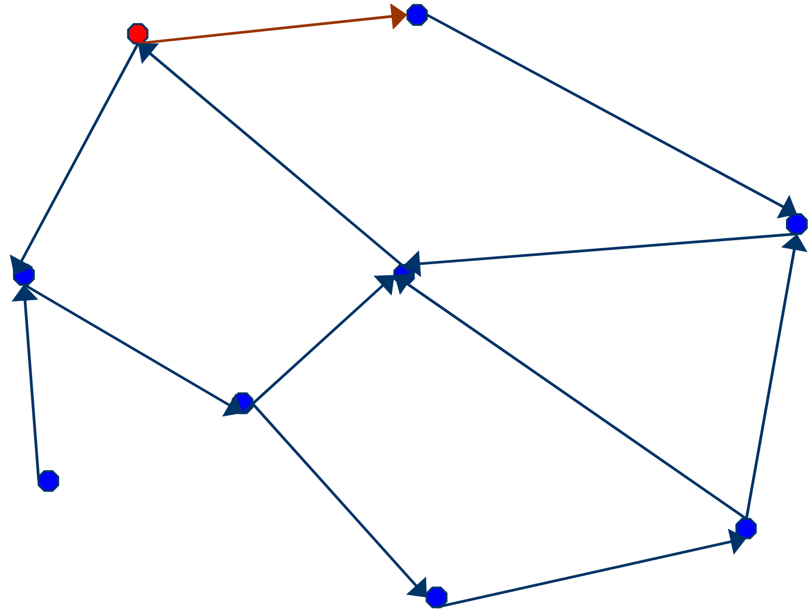
EigenTrust Algorithm

- First, choose a peer at random.



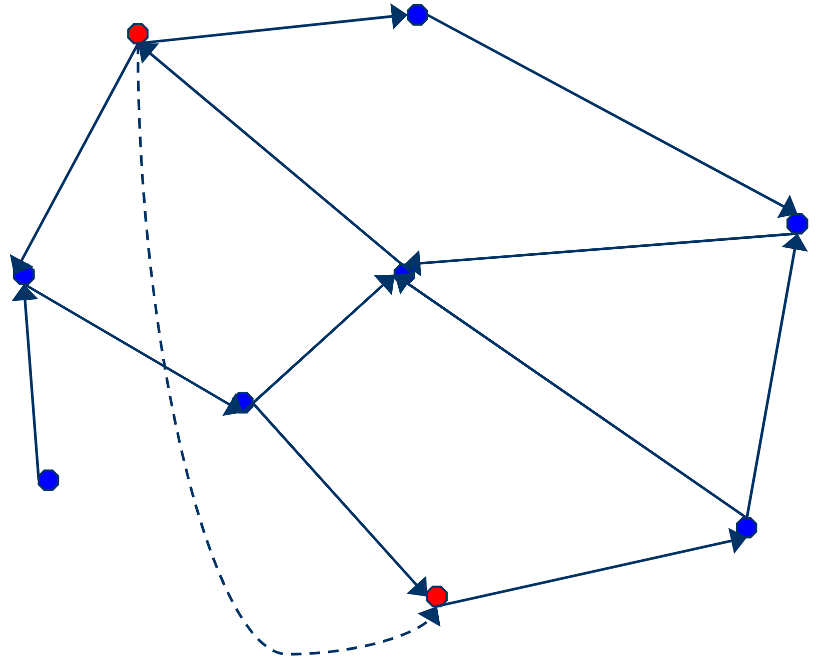
EigenTrust Algorithm

- With 1- ε probability, follow a recommendation from that peer to another peer.



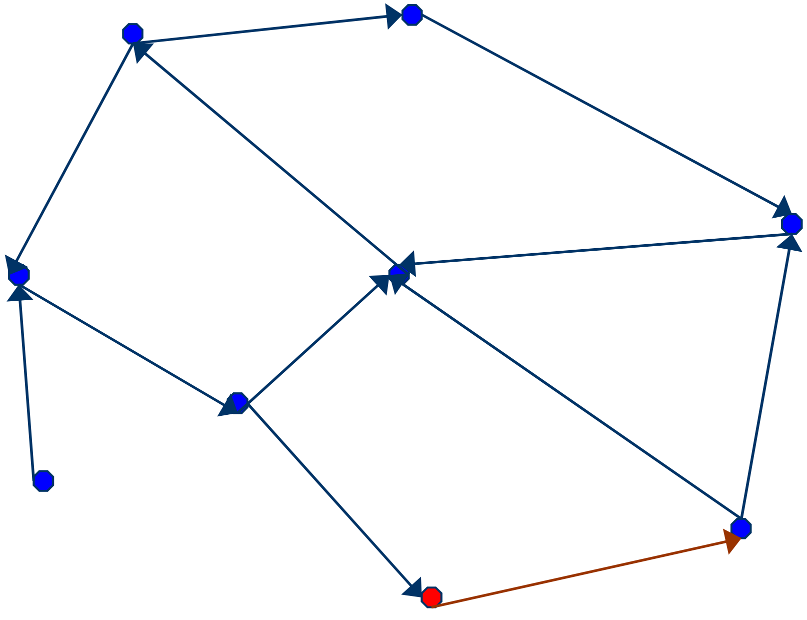
EigenTrust Algorithm

- Or with ε probability, teleport to a random peer.



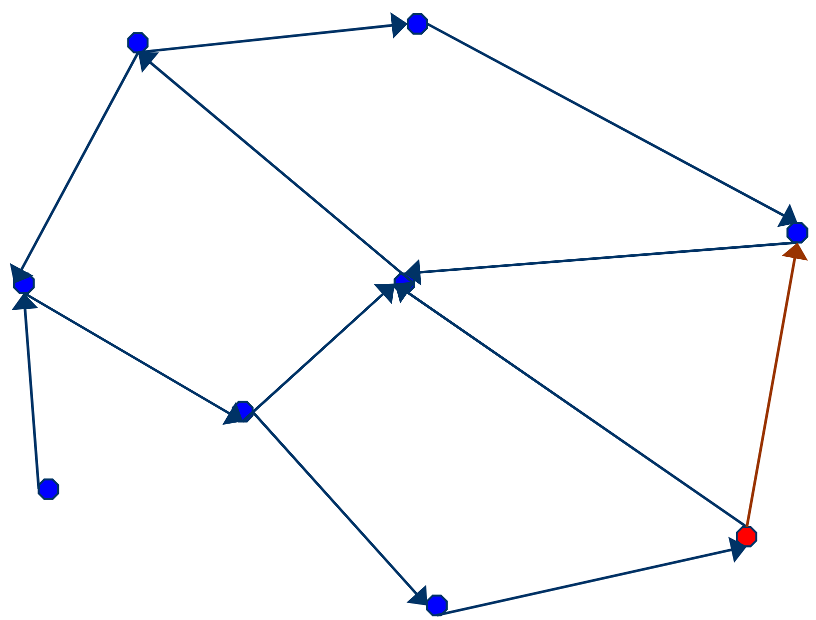
EigenTrust Algorithm

- Repeat



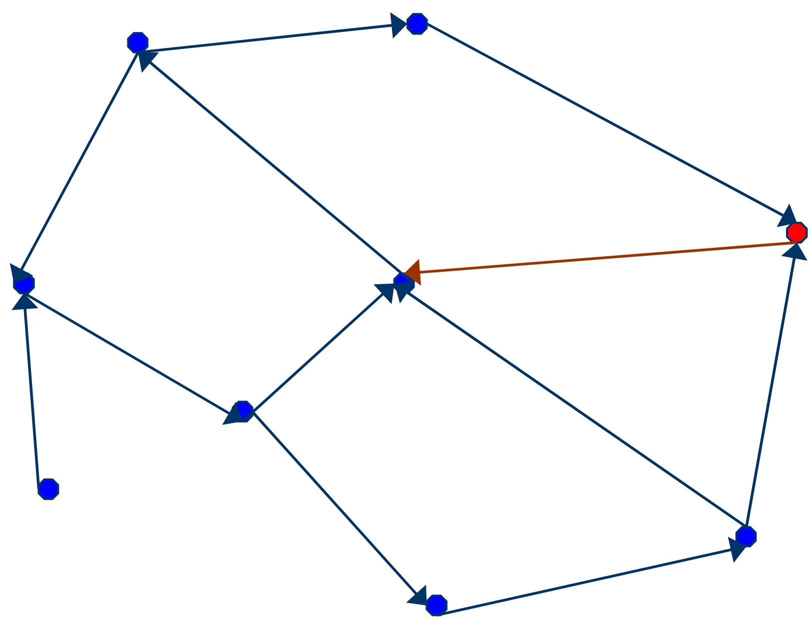
EigenTrust Algorithm

- Repeat



EigenTrust Algorithm

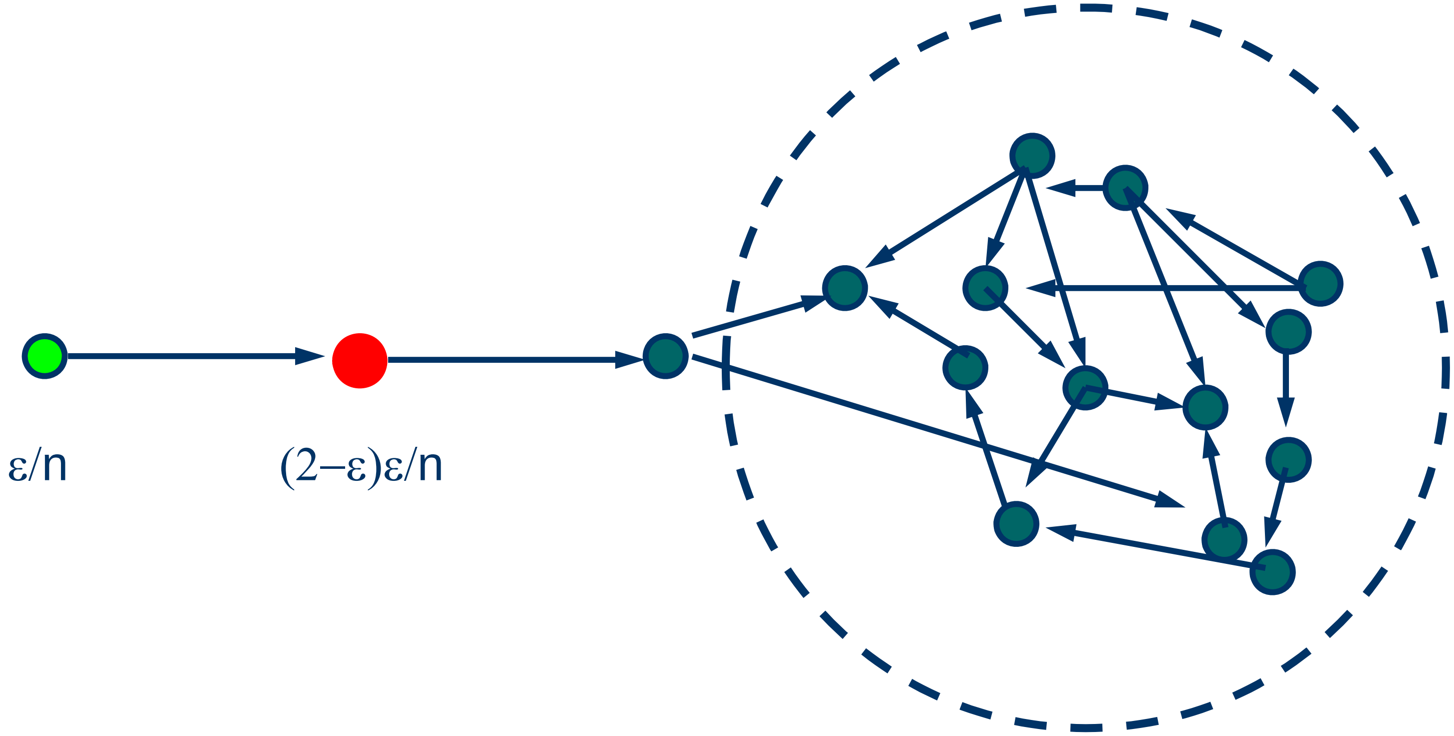
- Repeat



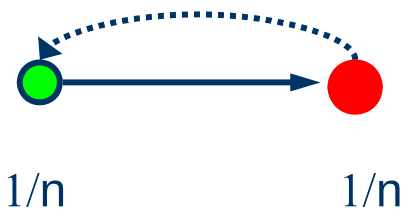
EigenTrust Algorithm

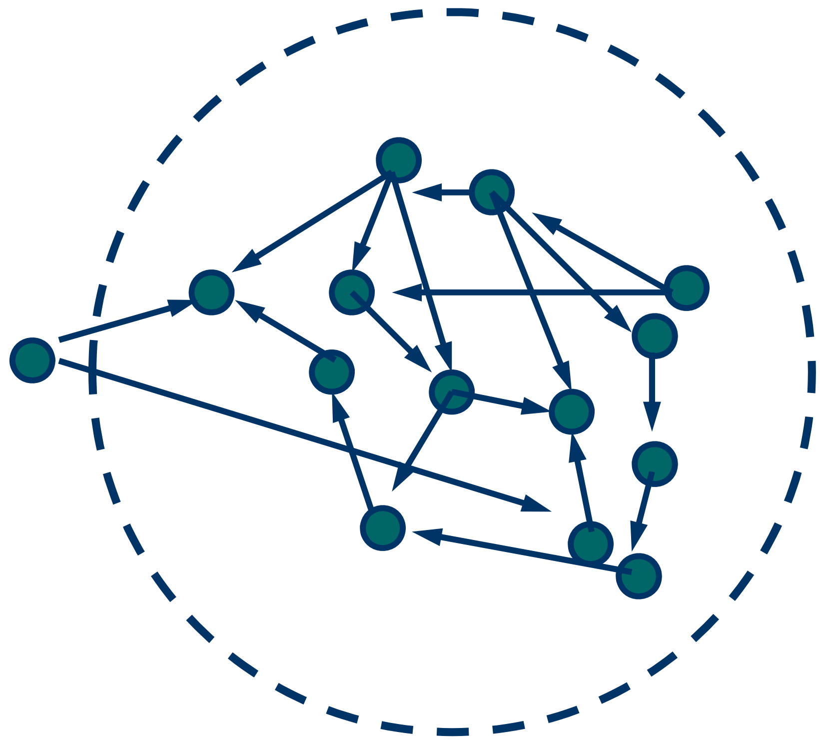
- Trust scores are the stable distribution over this Markov chain
- Can be calculated as the principal right eigenvector of the transition matrix

Manipulation Example

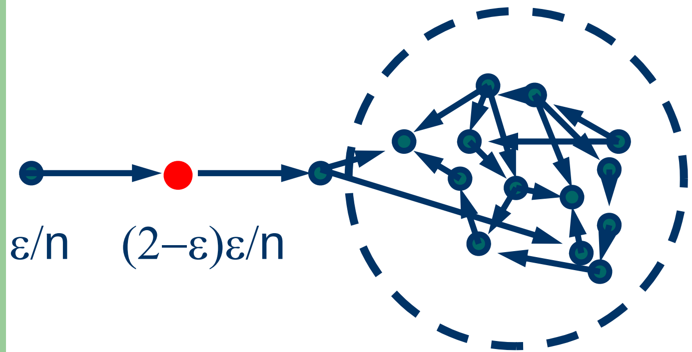


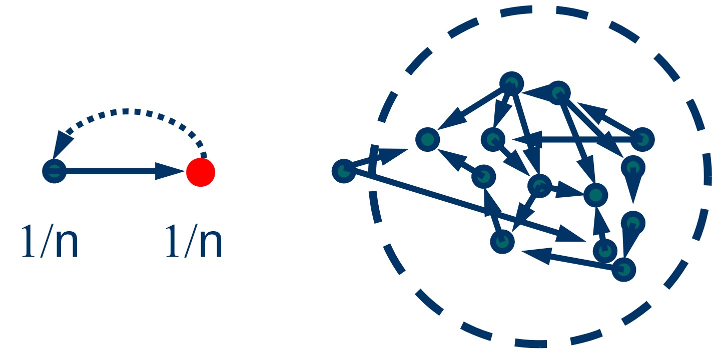
Manipulation Example





Manipulation example





- The middle node can increase its trust from (2-ε)ε/n to 1/n by manipulating its recommendations
- This is nearly a factor of three, and independent of the number of peers
- This example is likely to be common in practice

One-round Strategic Model

- Players: N
- Actions: report of downloads d_i
- Payoff for i 5 N: $T_i(d_i, d_{-i})$
- Dominant Strategy Equilibrium
  - For all d_{-i}, d_i^*, and d_i, $T_i(d_i^*, d_{-i})$ ¿ $T_i(d_i, d_{-i})$
  - Thus $T_i(d_i, d_{-i})$ must be equal over all d_i
  - No report of downloads can give greater utility than any other.

Indifference

- Peers' private values are not preferences!
  - Preferences are commonly known
  - Everyone wants high trust
  - This means we cannot use standard mechanism design tricks
- We will have to make peers indifferent between their recommendations

Strategic Goals

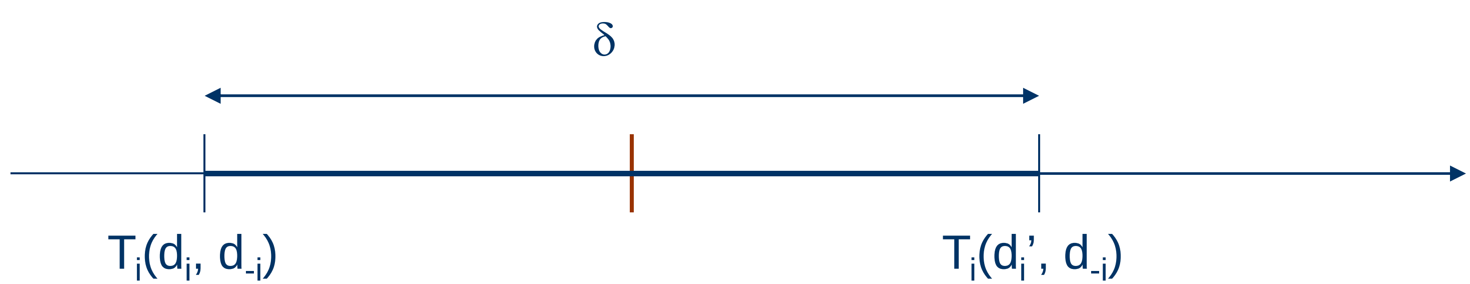
- Myopic non-manipulability
  - A peer cannot affect its score in round r+1 by manipulating its recommendations in round r
- Strong non-manipulability
  - A peer cannot affect its score in *any* future round by manipulating its recommendations in round r

Goals

- We seek to achieve a trust system T' with the following three properties:
  1. T' is non-manipulable (myopic or strong)
  2. T' approximates EigenTrust
     - We use variational distance $\sum_{i5\ N} |t_i - t_i'|$
     - This error is the maximum amount of trust malicious peers can gain over EigenTrust
  3. T' generates trust that sums to 1
     - We use T' to parcel out the quality of service in network.

A Negative Result

- Let *D* be a set of allowed downloads
- Let d_i, $d_i' \in D_i$ be download reports s.t.
  - $T_i(d_i', d_{-i}) \geq T_i(d_i, d_{-i}) + \delta$
- Then no T' can have error less than $\delta/2$



Best $T_i(d_{-i}) = ½\, T_i(d_i, d_{-i}) + ½\, T_i(d_i', d_{-i})$

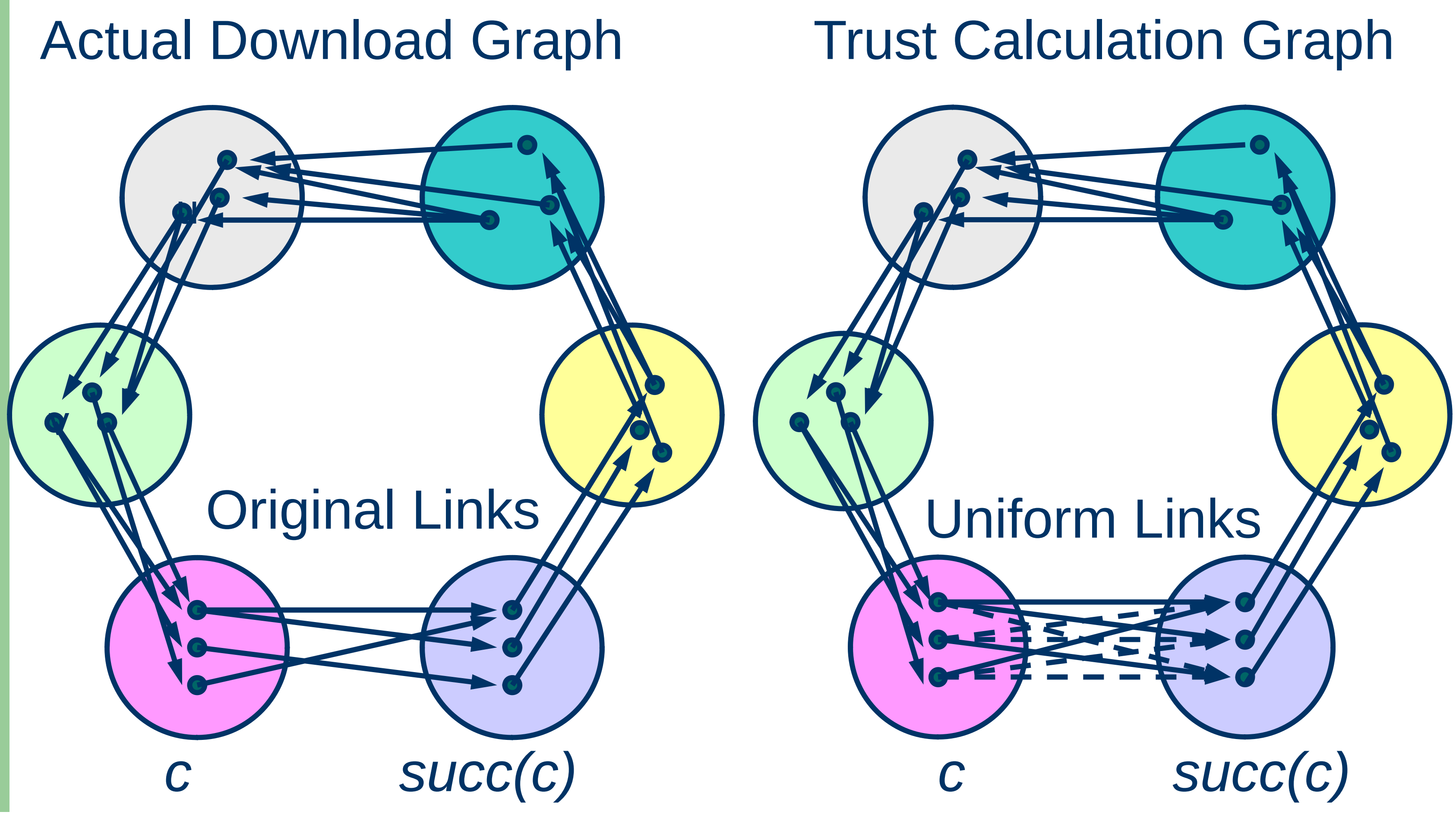
A Negative Result

- Let *D* be a set of legal downloads
- Let d_i, d_i' 5 D_i be download reports s.t.
  - $T_i(d_i', d_{-i})$ ¿ $T_i(d_i, d_{-i}) + \delta$
- Then no T' can have error less than $\delta/2$
- Thus, we must restrict the topology of the network so that no manipulation is too profitable

Achieving Myopic Non-manipulability

- Cyclic partitioning
  - Partition the nodes into m colors of equal size
  - Arrange the colors randomly into a cycle
  - Peers are only allowed to download from peers of their successor color
- Modified eigenvector calculation
  - When calculating the trust of peers of color c, set all peers of color c to have uniform links to succ(c).

Cyclic Partitioning



Cyclic Partitioning Properties

- T' sums to 1
  - Each color sums to 1/m
- T' is myopically non-manipulable
  - Each peer's recommendations are not used in calculating its trust
- T' approximates EigenTrust
  - $\sum_{i5\ N} |t_i - t_i'|$ ‰ $2(1-\varepsilon)^m$
  - need only O(log(1/α)) colors to bound error by α
  - proven by coupled Markov chain argument

Trade-offs

- As the number of colors increases
  - the approximation improves
  - but the trust scores become less useful
  - if m = n, then each node has no choice in downloading, and the trust scores are useless
- Thus, there is a trade-off between fidelity to EigenTrust and the usefulness of the trust scores
  - With a logarithmic number of colors, we have the best of both worlds

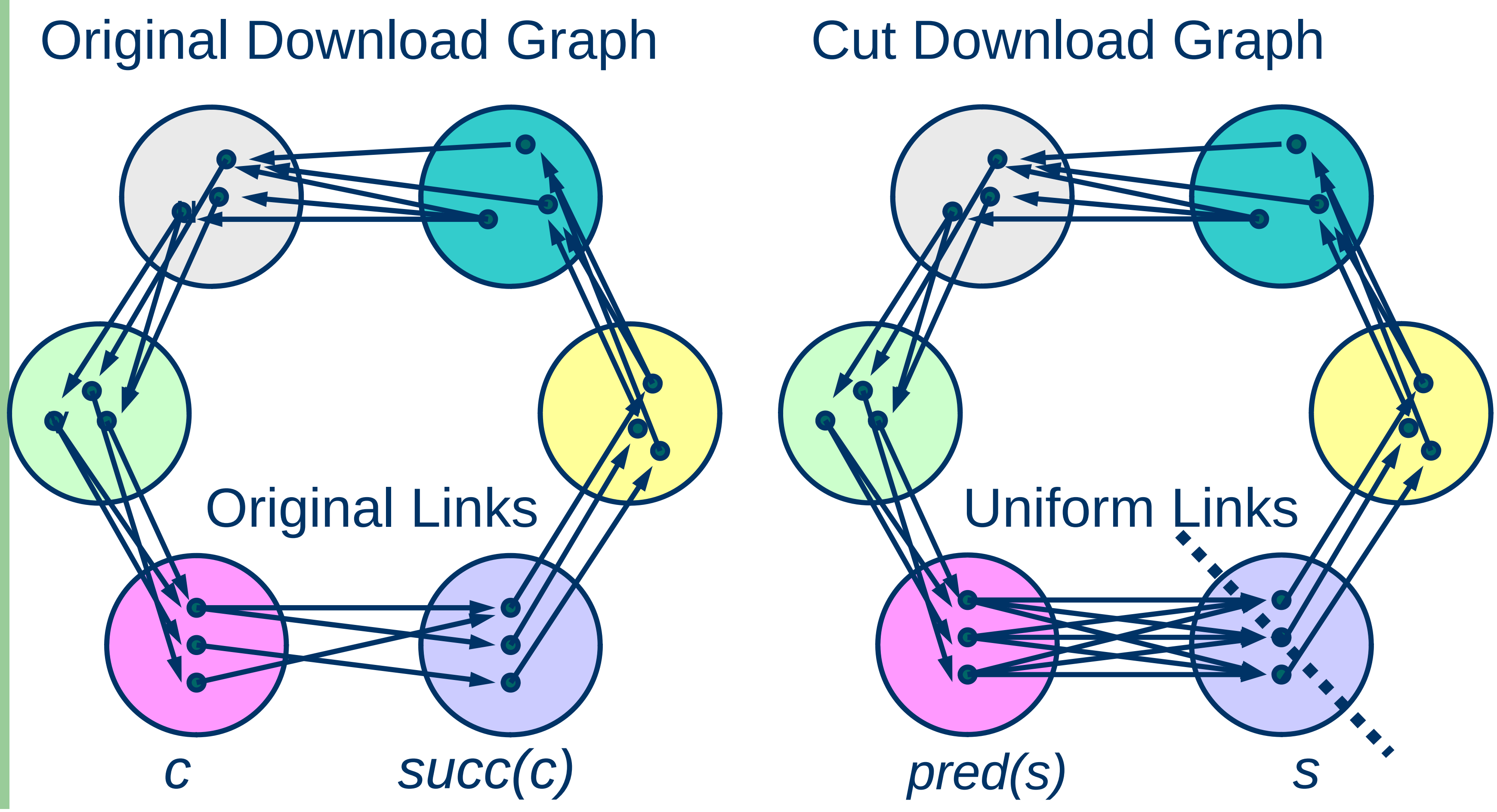
Achieving Strong Non-manipulability

- We wish to prevent peers from manipulating their scores in every future round
- Thus, we require that the trust score of a peer *i* is independent of the recommendations of each peer *j* whose trust score *i* can affect
- Influence is a directed acyclic graph

Cut Partitioning

- Choose a start color s
- Set nodes of the start color to have uniform trust
  - Equivalent to setting pred(s) to have uniform outgoing links
- Propagate trust forward around the cycle to pred(s)

Cut Partitioning



Cut Partitioning Properties

- T' generates a trust score that sums to 1
  - Trust is calculated as a probability distribution
- T' is strongly non-manipulable
  - Influence is a directed acyclic graph
- T' approximates EigenTrust
  - $\sum_{i \in N} |t_i - t_i'| \le 2/\varepsilon m$
  - only linear decrease in error as m increases
  - strong non-manipulability is harder to achieve with EigenTrust

Conclusions

- We have exhibited two truthful trust systems under different strategic models
- Our model and methods are applicable to a variety of trust systems, not just EigenTrust
- Future work:
  - Quantify trade-off between approximation and usefulness of trust
  - Specify the remaining policies of a truthful P2P system based on EigenTrust
  - Extend to other trust systems