Taming Dynamic and Selfish Peers

Stefan Schmid
Distributed Computing Group
ETH Zurich, Switzerland
schmiste@tik.ee.ethz.ch

Talk based on our papers at IPTPS 2005 and 2006
Outline of this Talk

• Current research of our group at ETH
  – Based on our papers at IPTPS 2005 and IPTPS 2006
  – Still many interesting open questions!

• Two challenges related to P2P topologies

  **CHALLENGE 1: Dynamic Peers**
  - dynamics of P2P systems,
  - i.e., joins and leaves of peers ("churn")
  - our approach to maintain desirable properties in spite of churn

  **CHALLENGE 2: Selfish Peers**
  - impact of selfish behavior on P2P topologies
  - How bad are topologies formed by selfish peers?
  - Stability of topologies formed by selfish peers?
CHALLENGE 1:

Dynamic Peers
Motivation (1)

• P2P systems are
  – composed of unreliable desktop machines
  – under control of individual users

⇒ Peers may join and leave the network at any time and concurrently ("churn")!

• However:
  – many systems maintain their properties only in static environments!
Motivation (2)

How to maintain desirable properties such as

- Connectivity,
- Network diameter,
- Peer degree?
A First Approach

- Fault-tolerant hypercube?
- What if number of peers is not $2^i$?
- How to prevent degeneration?
- Where to store data?

Idea: Simulate the hypercube!
Simulated Hypercube System

Simulation: Node consists of several peers!

Basic components:

- Route peers to sparse areas
  Token distribution algorithm!

- Adapt dimension
  Information aggregation algorithm!
The Adversary

- Model worst-case faults with an adversary $ADV(J,L,\lambda)$

- $ADV(J,L,\lambda)$ has complete visibility of the entire state of the system

- May add at most $J$ and remove at most $L$ peers in any time period of length $\lambda$

- Note: Adversary is not Byzantine!
Results

• In spite of $\text{ADV}(O(\log n), O(\log n), 1)$:
  
  – always at least one peer per node (no data lost!),
  
  – peer degree $O(\log n)$ (asymptotically optimal!),
  
  – network diameter $O(\log n)$. 
Discussion

- Simulated topology: Taming dynamic peers by redundancy!

- Simulated topology: A simple blueprint for many P2P topologies!
  - Requires token distribution and information aggregation on the topology!

- A lot of future work!
  - A first step only: dynamics of P2P systems offer many research challenges!
  - E.g.: Other dynamics models, self-stabilization after larger changes, etc.!
CHALLENGE 2:

Selfish Peers
Challenge 1 -> Challenge 2

- Simulated hypercube topology is fine...
- ... if peers act according to protocol!
- However, in practice, peers can perform selfishly!
Motivation (1)

Power of Peer-to-Peer Computing = Accumulation of Resources of Individual Peers

- CPU Cycles
- Memory
- Bandwidth
- ...

Collaboration is of peers is vital!

However, many free riders in practice!
Motivation (2)

- Free riding
  - Downloading without uploading
  - Using storage of other peers without contributing own disk space
  - Etc.

- Our research: selfish neighbor selection in unstructured P2P systems

- Goals of selfish peer:
  1. Maintain links only to a few neighbors (small out-degree)
  2. Small latencies to all other peers in the system (fast lookups)

What is the impact on the P2P topologies?
Problem Statement (1)

• $n$ peers \{$\pi_0, \ldots, \pi_{n-1}$\}

• distributed in a metric space
  – Metric space defines distances between peers
  – triangle inequality, etc.
  – E.g., Euclidean plane
Problem Statement (2)

• Each peer can choose...
  – to which
  – and how many
  – ... other peers its connects

• Yields a **directed graph** $G$
Problem Statement (3)

- Goal of a selfish peer:

  (1) Maintain a small number of neighbors only (\textit{out-degree})

  (2) Small \textit{stretches} to all other peers in the system

  - Fast lookups!
  - Shortest distance using edges of peers in \( G \ldots \)
  - \ldots divided by shortest direct distance

- Only little \textit{memory} used
- Small \textit{maintenance} overhead
Problem Statement (4)

- Cost of a peer:
  - Number of neighbors (out-degree) times a parameter $\alpha$
  - plus stretches to all other peers
  - $\alpha$ captures the trade-off between link and stretch cost

$$\text{cost}_i = \alpha \ \text{outdeg}_i + \sum_{i \neq j} \text{stretch}_G(\pi_i, \pi_j)$$

- Goal of a peer: Minimize its cost!
Game-theoretic Tools (1)

- **Social Cost**
  - Sum of costs of all individual peers:
  - \( \Rightarrow \) Criterion to evaluate the overall efficiency of a P2P topology!
  
  \[
  \text{Cost} = \sum_i \text{cost}_i = \sum_i (\alpha \ \text{outdeg}_i + \sum_{i \neq j} \text{stretch}_G(\pi_i, \pi_j))
  \]

- **Social Optimum OPT**
  - Topology with minimal social cost of a given problem instance
  - \( \Rightarrow \) “topology formed by collaborating peers”!

- **Nash equilibrium**
  - “Result” of selfish behavior \( \Rightarrow \) “topology formed by selfish peers”
  - Topology in which no peer can reduce its costs by changing its neighbor set
  - In the following, let NASH be social cost of worst equilibrium
Game-theoretic Tools (2)

• How to compute the impact of selfish behavior?

• **Price of Anarchy**
  – Captures the impact of selfish behavior by comparison with optimal solution
  – Formally: social costs of worst Nash equilibrium divided by optimal social cost

\[ \text{PoA} = \max_i \{ \text{NASH}(I) / \text{OPT}(I) \} \]
Results: Price of Anarchy

Theorem: The price of anarchy is
\[ \text{PoA} \in \Theta(\min\{\alpha, n\}) \]

=> PoA can grow linearly in the total number of peers

=> PoA can grow linearly in the relative importance of degree costs \( \alpha \)

- This is already true in a 1-dimensional Euclidean space:
  - Is Nash equilibrium, at has large social costs compared to doubly linked list

\[
\begin{array}{cccccccc}
\pi_1 & \pi_2 & \pi_3 & \pi_4 & \pi_5 & \ldots & \pi_{i-1} & \pi_i & \pi_{i+1} & \ldots & \pi_n \\
\frac{1}{2} & \alpha & \frac{1}{2} \alpha^2 & \alpha^3 & \frac{1}{2} \alpha^4 & \ldots & \frac{1}{2} \alpha^{i-2} & \alpha^{i-1} & \frac{1}{2} \alpha^i & \ldots & \frac{1}{2} \alpha^{n-1}
\end{array}
\]
Results: Stability

How long thus it take until no peer has an incentive to change its neighbors anymore?

Theorem:
Even in the absence of churn, peer mobility or other sources of dynamism, the system may never stabilize (i.e., P2P system never reaches a pure Nash equilibrium)!
Discussion

- **Unstructured topologies** created by selfish peers

- **Efficiency of topology deteriorates** linearly in the relative importance of links compared to stretch costs, and in the number of peers

- **Instable** even in static environments

- **Discussion**
  - Relevance in practice?
  - If yes: **How to tame the selfish peers?**
  - Mechanism design?
Taming Dynamic and Selfish Peers

Thank you for your attention!

Questions? Comments? Feedback?

Further reading:


Email: schmiste@tik.ee.ethz.ch

Website: http://dcg.ethz.ch/members/stefan.html

Stefan Schmid @ Dagstuhl 2006