Dynamics, Non-Cooperation, and Other Algorithmic Challenges in Peer-to-Peer Computing

Stefan Schmid

Distributed Computing Group

ETH

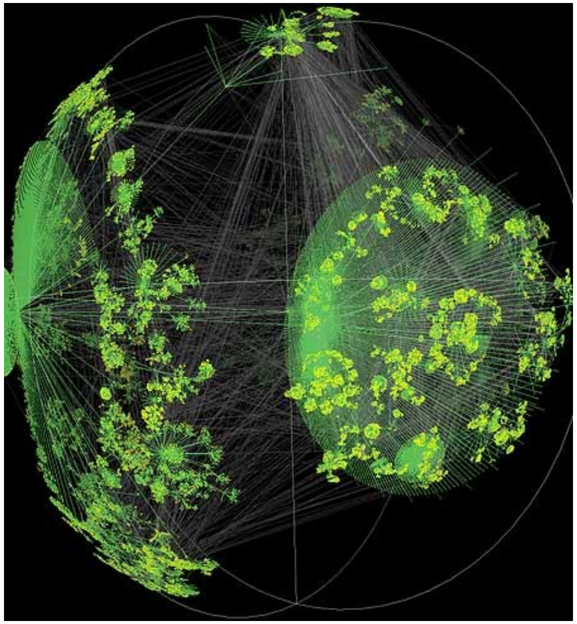
Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Visit at

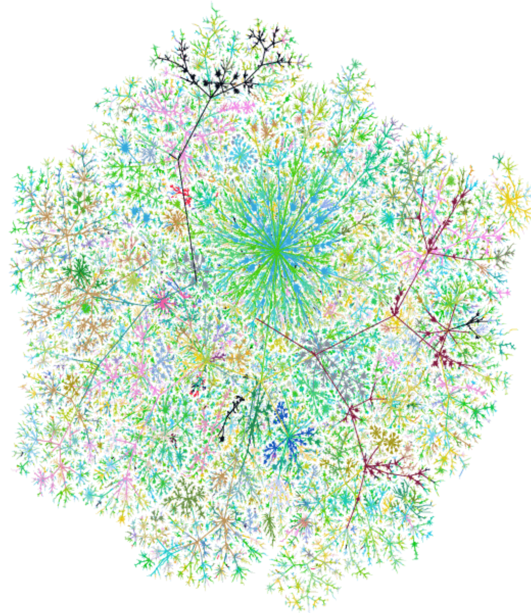
Los Alamos National Laboratories

November 2007

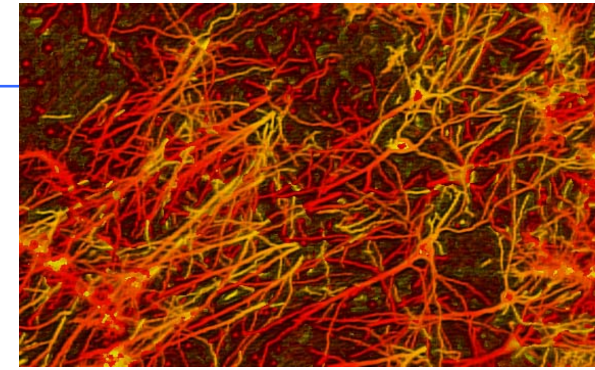
Networks



Internet Graph

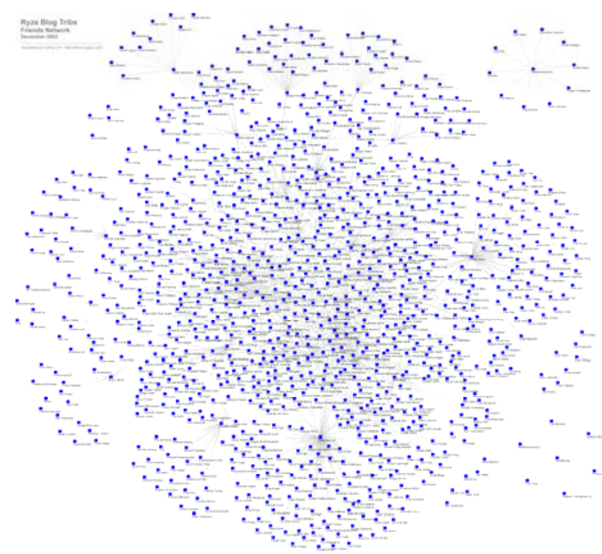


Web Graph



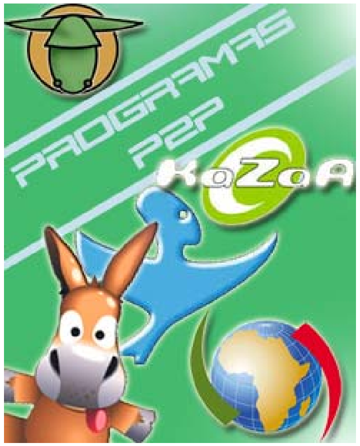
Neuron Networks

Public Transportation Networks



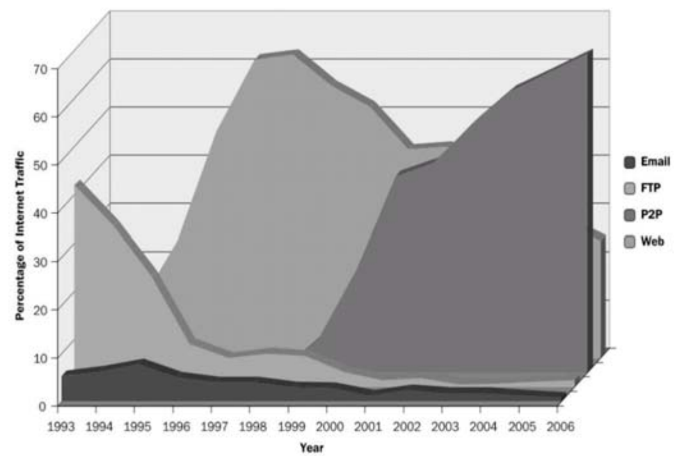
Social Graphs

This Talk: Peer-to-Peer Networks



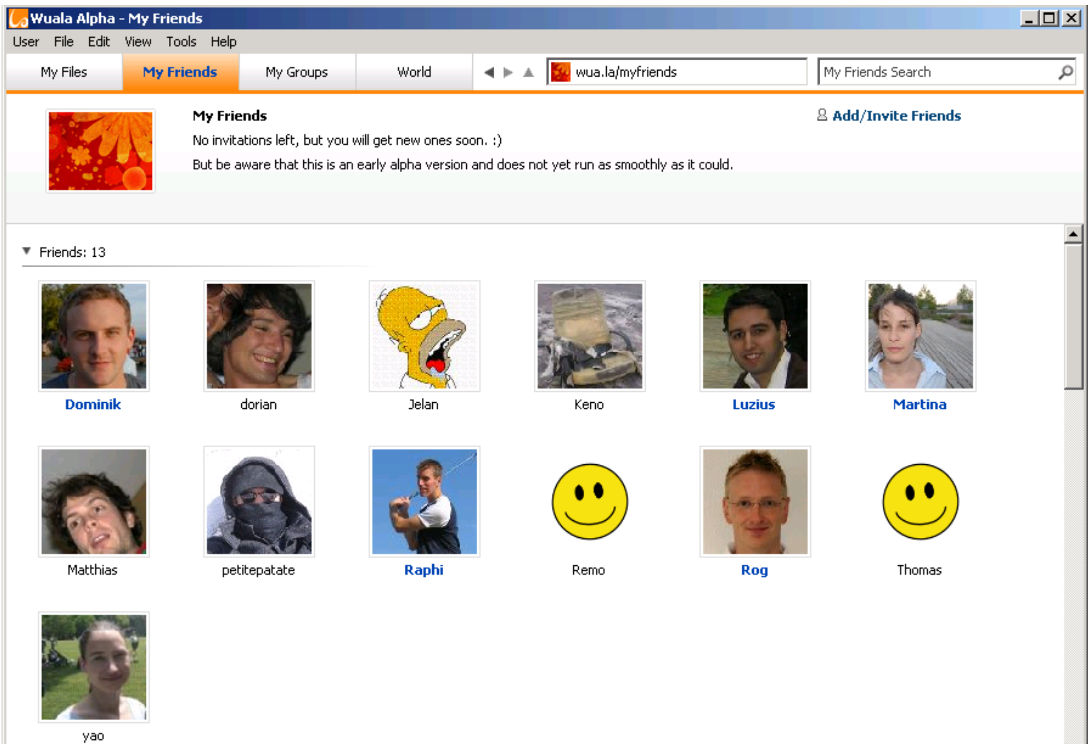
- Popular Examples:
  - BitTorrent, eMule, Kazaa, ...
  - Zattoo, Joost, ...
  - Skype, ...
  - etc.

- Important: Accounts for much Internet traffic today!
  (source: *cachelogic.com*)



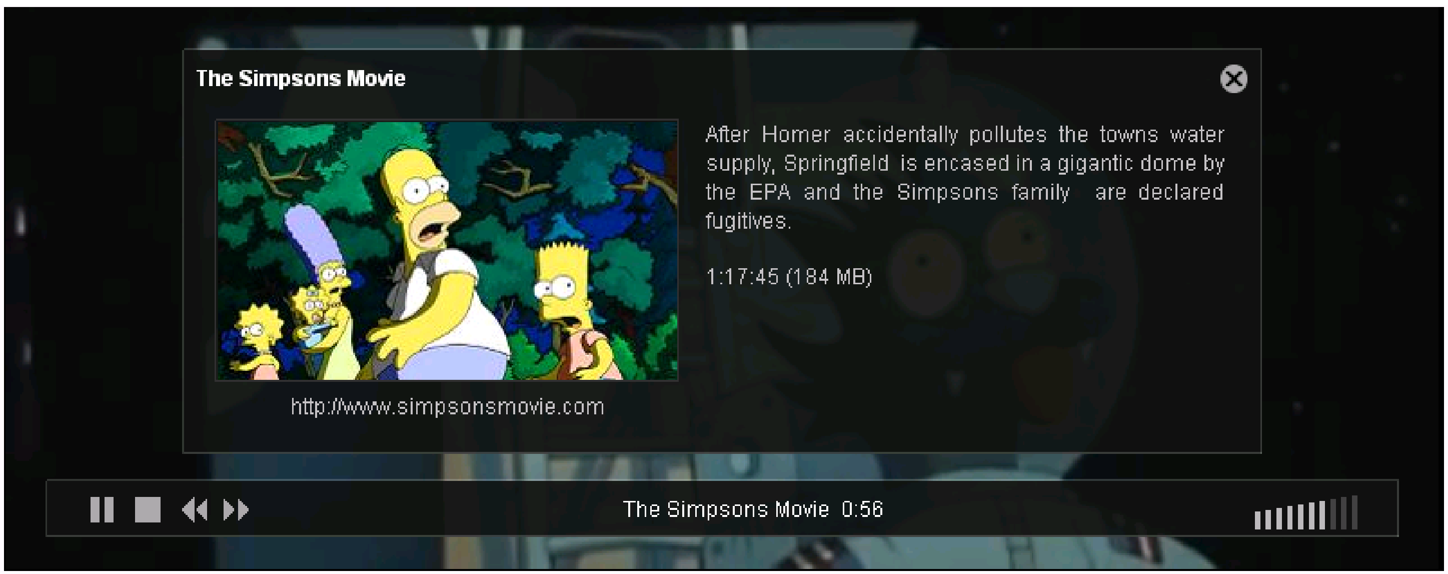
What For?

- Many applications!
- File sharing, file backup, social networking: e.g. Wuala



What For?

- On demand and live streaming, e.g., Pulsar
  - Users / peers help to distribute contents further
  - Cheap infrastructure at content provider is ok!



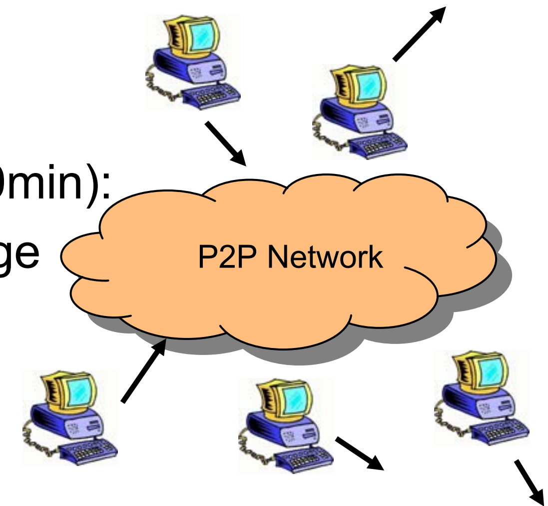
What For?

- Peer-to-peer games, e.g., xPilot
  - Scalability (multicast updates, distributed storage, ...)
  - Cheaters? Synchronization?

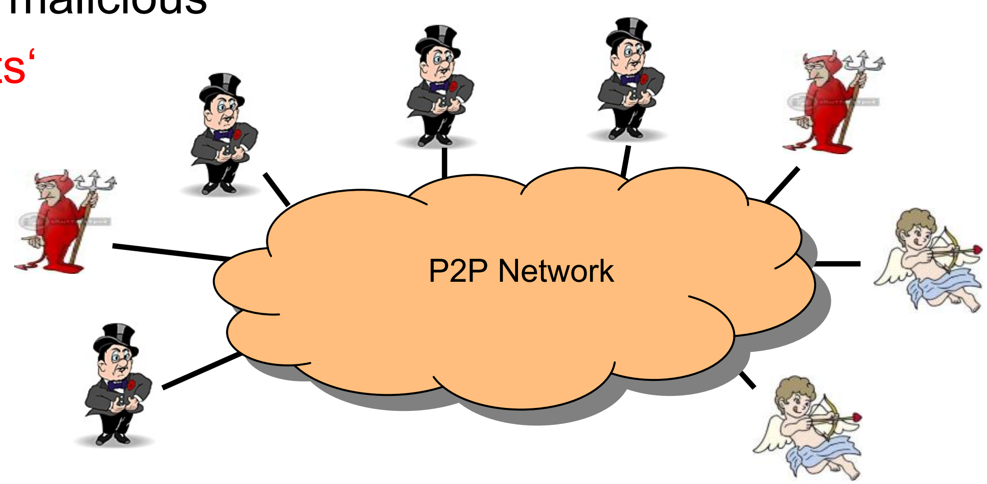
- Among many more...

Why Are P2P Networks Interesting for Research?

- Challenging properties...
- Peer-to-peer networks are highly dynamic
  - Frequent membership changes
  - If a peer only connects for downloading a file (say 60min): Network of 1 mio. peers implies a membership change every 3 ms on average!
  - Peers join and leave all the time and concurrently



- Participants are humans
  - Peers are under control of individual decision making
  - Participants may be selfish or malicious
  - Paradigm relies on participants' contribution of content, bandwidth, disk space, etc.!

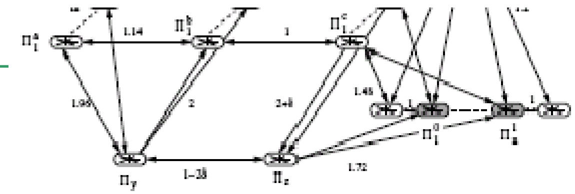


So…

How to provide full functionality despite dynamic, selfish and heterogeneous participants?

Our Research

- Often requires algorithms and theory...



$[K] \times [M]$ of elements $e_1, \ldots, e_k \in [K]$ w
nctions $\mathfrak{h}_1, \ldots, \mathfrak{h}_t$. It suffices to show that t
f the functions $\mathfrak{h}_i$ are λ-good for S is less
less than $(1-\delta)t/2$ functions $\mathfrak{h}_i$ satisfy $\mathfrak{h}$

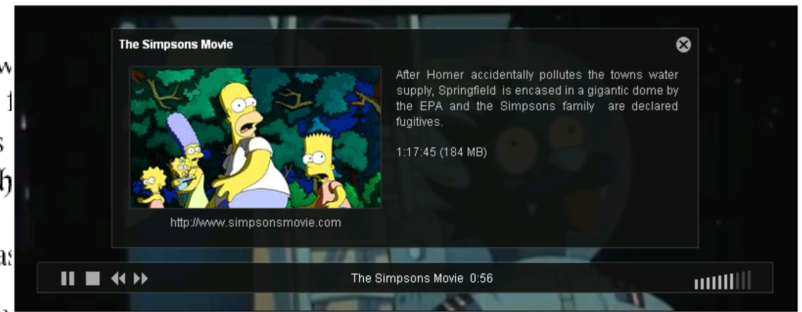
ow that the the probability of a random has

$$\mathbb{P}[\mathfrak{h} \text{ not } \lambda\text{-good for } S] < 2e^{-\lambda^2/2}.$$

hash functions $\mathfrak{h}_i$, $i \in \{1, \ldots, t\}$ that are λ-good for S.
). Using a Chernoff bound, for $\lambda \geq 2$ we thus get

$$(1-\delta)t(1-2e^{-\lambda^2/2})\Big] < e^{-\delta^2 t(1-2e^{-\lambda^2/2})/2} \leq e^{-\delta^2 t}$$

is becomes less than $1/(2\ell 2^{KM})$. It is also not hard to see
function



ology resulting
re connected as
ate inter-cluster
$\hat{\pi}_z$ connects to
e satisfying as-
0 and $x_n = 1$.

for all j that the
$(\Pi_j^a \cup \Pi_j^b \cup \Pi_j^c)$
ecting to such a

6.2 Satisfiable Instances

In this section, we show that if I has a satisfying assignment A_I, then there exists a Nash equilibrium in $\mathcal{M}_I^k$. For this purpose, we explicitly construct a set of strategies s, which we prove to constitute a Nash equilibrium. As for notation, we define $A_I(x_i)$ to be the assignment of x_i in A_I, i.e.,

$$A_I(x_i) := \begin{cases} 1, & x_i \text{ is set to 1 in } A_I \\ 0, & x_i \text{ is set to 0 in } A_I. \end{cases} \quad (1)$$

Furthermore, we define in every cluster Π_g a single *leader peer*, which we denote by $\hat{\pi}_g$. The role of this leader-peer is to construct all inter-cluster links going from this cluster to peers located in other clusters. The strategy of the remaining *non-leader peers* $\tilde{\pi}_g \in \Pi_g \setminus \{\hat{\pi}_g\}$ is to connect to the unique leader peer within their cluster. Formally, the strategy s_g for a non-leader peer $\tilde{\pi}_g \in \Pi_g \setminus \{\hat{\pi}_g\}$ is $s_g := \{\hat{\pi}_g\}$. For each leader-peer, we define the set of strategies s as follows:

$$s_y := \Pi_y \cup \{\hat{\pi}_z\} \cup \bigcup_{C_j \in \mathcal{C}} \{\hat{\pi}_j^a, \hat{\pi}_j^b\}$$

$$s_z := \Pi_z \cup \{\hat{\pi}_y\} \cup \bigcup_{x_i \in \mathcal{X}} \{\hat{\pi}_i^{A_I(x_i)}\}$$

$$s_c := \Pi_c \cup \bigcup_{x_i \in \mathcal{X}} \{\hat{\pi}_i^0 \cup \hat{\pi}_i^1\} \cup \bigcup_{C_j \in \mathcal{C}} \{\hat{\pi}_j^a \cup \hat{\pi}_j^b \cup \hat{\pi}_j^c\}$$

$$s_j^a := \Pi_j^a \cup \{\hat{\pi}_c, \pi_y, \hat{\pi}_j^b\} \qquad , \forall C_j \in \mathcal{C}$$

Outline of Talk

- Co… churn (IPTPS 200…, IWQoS 2006)
- BitThief: Today's s… be exploited by selfish participants (HotNets 2006)
- Game-theoret… analysis of selfi… …or (IPTPS 2006, PODC 2006)

very briefly…

very briefly…

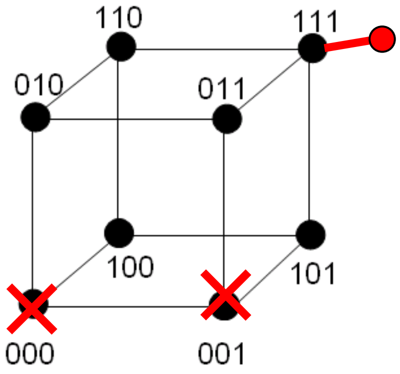
more in detail…

Outline of Talk

- **Coping with churn (IPTPS 2005, IWQoS 2006)**
- BitThief: Today‘s system can be exploited by selfish participants (HotNets 2006)
- Game-theoretic analysis of selfish behavior (IPTPS 2006, PODC 2006)

High Dynamics on Hypercube?

- Motivation: Why is dynamics a problem?
- Frequent membership changes are called churn
- How to maintain low network diameter and low node degree in spite of dynamics? How to prevent data loss?
- Popular topology: Hypercube
  - Logarithmic diameter, logarithmic node degree

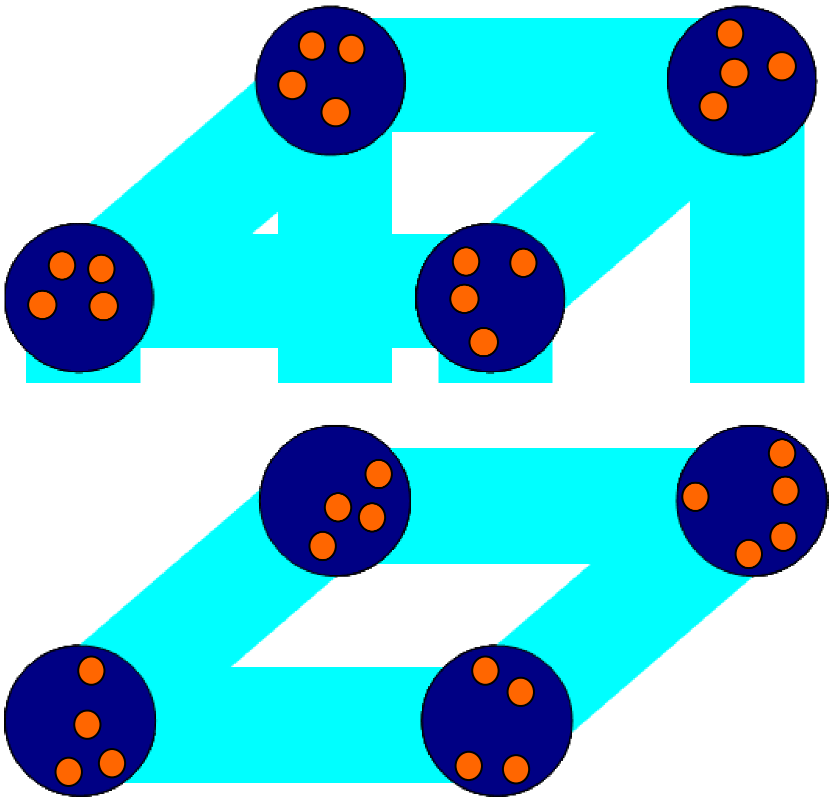


Resilient Solution

- Simulating the hypercube!
  - Several peers „simulate“ one node

- Maintenance algorithm:
  - Distribute peers evenly among IDs (nodes) (-> token distribution problem)
  - Distributed estimation of total number of peers and adapt dimension of hypercube when necessary

- Thus, at least one peer per ID (node) at any time!



Analysis

Even if an adversary adds and removes a **logarithmic** number of peers per communication round in a worst-case manner, the network diameter is always **logarithmic** and **no data is lost**.

- Also works for other topologies, e.g., pancake graph!

Outline of Talk

- Coping with churn (IPTPS 2005, IWQoS 2006)
- BitThief: Today's system can be exploited by selfish participants (HotNets 2006)
- Game-theoretic analysis of selfish behavior (IPTPS 2006, PODC 2006)

Outline of Talk

- Coping with churn (IPTPS 2005, IWQoS 2006)
- **BitThief: Today‘s system can be exploited by selfish participants (HotNets 2006)**
- Game-theoretic analysis of selfish behavior (IPTPS 2006, PODC 2006)

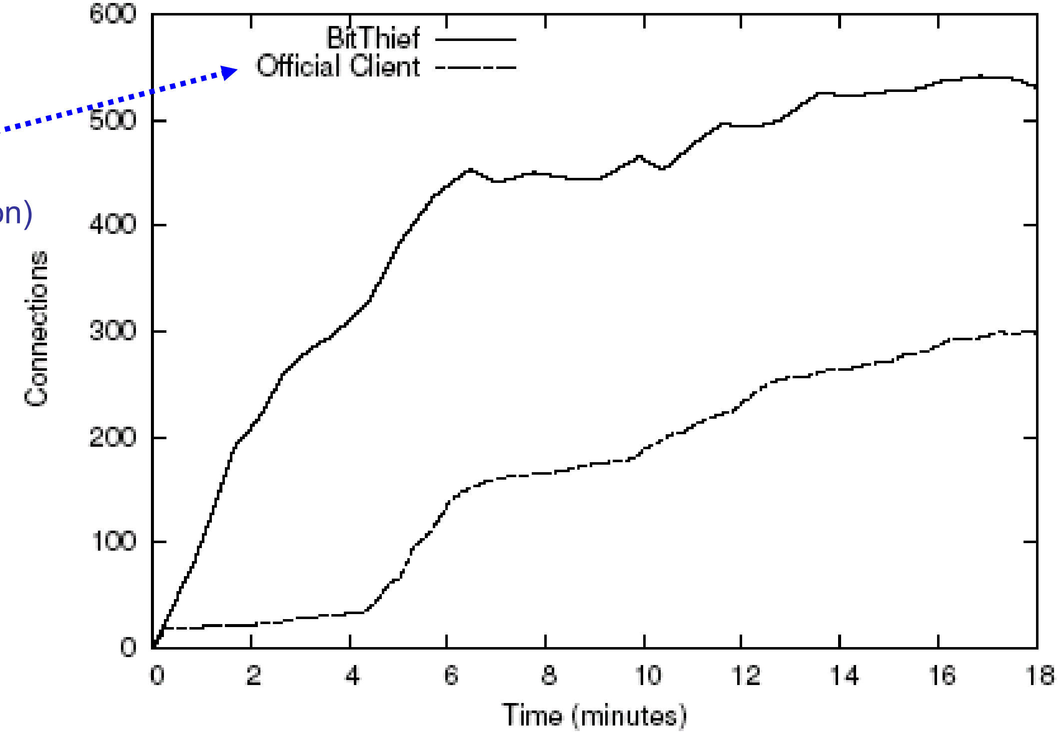
BitThief

- Case Study: Free riding in BitTorrent
- BitThief: Free-riding BitTorrent client
  - written in Java
  - Downloads entire files efficiently without uploading any data
  - Despite BitTorrent's Tit-for-Tat incentive mechanism!

BitThief’s Exploits (1)

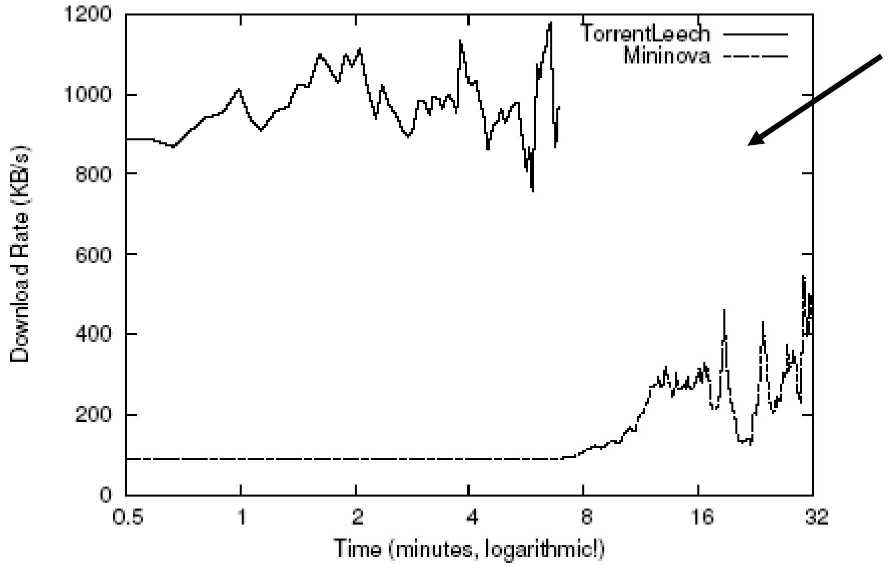
- Exploit 1: Exploit unchoking mechanism
  - New peer has nothing to offer -> BitTorrent peers have unchoking slots
  - Exploit: Open as many TCP connections as possible!

V4.20.2 from bittorrent.com (written in Python)



BitThief's Exploits (2)

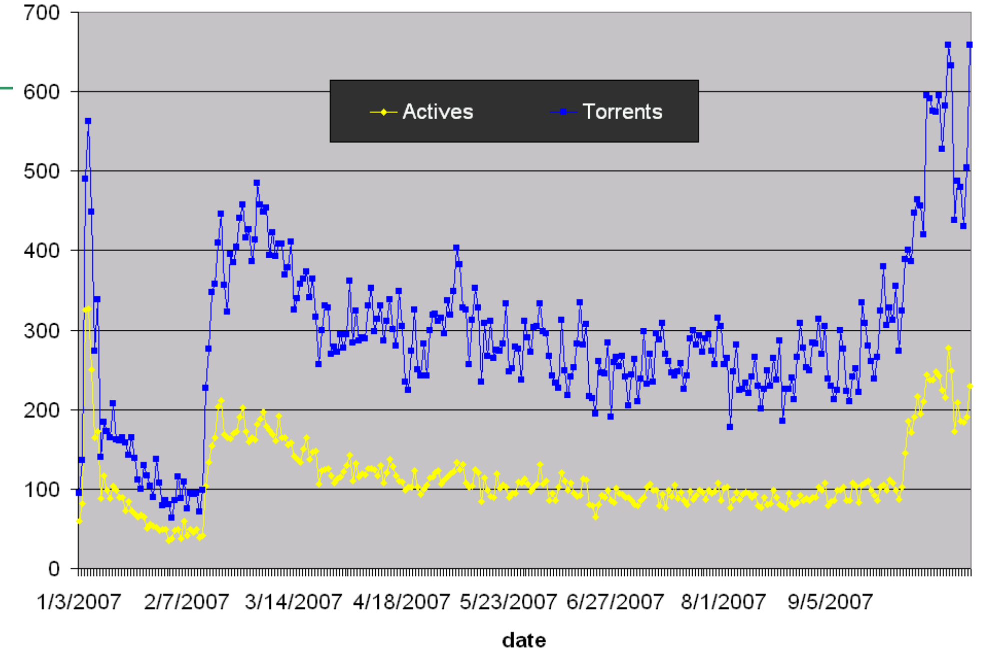
- Exploit 2: Sharing Communities
  - Communities require user registration and ban uncooperative peers
  - Many seeders! (= peers which only upload)
  - Exploit: Fake tracker announcements, i.e., report large amounts of uploaded data



4 x faster!
(BitThief had a faked sharing ratio of 1.4; in both networks, BitThief connected to roughly 300 peers)

Some Reactions

- Selfishness in p2p computing seems to be an important topic – inside and outside academic world: blogs, emails, up to 100 paper downloads per day! (>3000 in January 2007)
- Recommendation on Mininova FAQ (!)
- But still some concerns...



"Anyhow, bitthief is a client which I've been waiting for so long, I mean.. bitcomet bent the rules but never really broke any of them.. that much Bitthief is an interesting client in that it openly says "fuck you, and fuck your swarm" to the torrent community. I wonder how fast this will get banned at every tracker alive. As others have said, this makes bittyrant look like a sunday school boy."

-----Original Message-----
From: Warren Henning [mailto:warren.henning@gmail.com]
Sent: Friday, January 12, 2007 3:03 PM
To: lochert@tik.ee.ethz.ch; schmiste@tik.ee.ethz.ch; wattenhofer@tik.ee.ethz.ch
Subject: Stop distributing BitThief, you jerks!

BitTorrent is a beautiful thing and you are intentionally fucking it up by distributing software that is apparently specifically designed to attack the entire basis of the function of BitTorrent, software that serves no legitimate purpose.

Luckily it apparently requires having a JRE installed right now, and the knuckle-dragging numbskulls you've worked so hard to cater to are probably too lazy to install that.

You people piss me off.

Warren Henning

Effects of Selfishness?

- Question remains:

- Is selfishness really a pro[illegible] p2p networks?
  - Tools to estimate i[illegible] selfishness: game theory!

Tackled next!

Outline of Talk

- Coping with churn (IPTPS 2005, IWQoS 2006)
- BitThief: Today's system can be exploited by selfish participants (HotNets 2006)
- Game-theoretic analysis of selfish behavior (IPTPS 2006, PODC 2006)

Outline of Talk

- Coping with churn (IPTPS 2005, IWQoS 2006)
- BitThief: Today's system can be exploited by selfish participants (HotNets 2006)
- **Game-theoretic analysis of selfish behavior (IPTPS 2006, PODC 2006)**

Selfishness in P2P Networks

- How to study the impact of non-cooperation / selfish behavior?
- Example: Impact of selfish neighbor selection in unstructured P2P systems
- Goals of selfish peer:
  - It wants to have small latencies, quick look-ups
  - It wants to have small set of neighbors (maintenance overhead)

What is the impact on the P2P topologies?

→ Efficiency

→ Stability

Model – The “Locality Game”

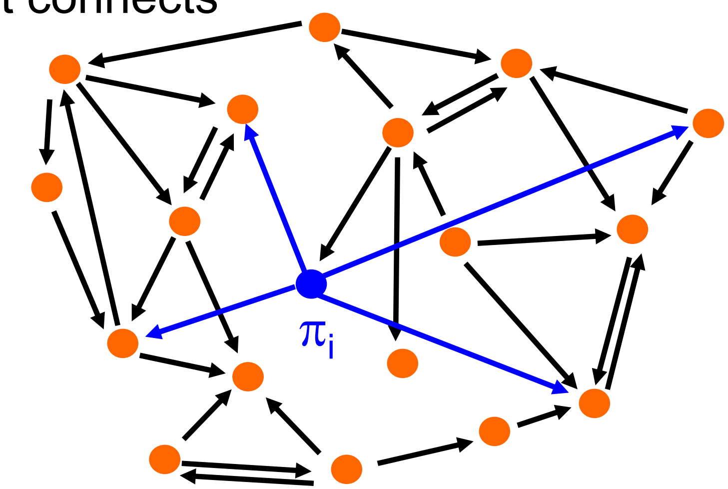
- Model inspired by network creation game [Fabrikant et al, PODC‘03]

  - Sparked much future research, e.g., study of bilateral links (both players pay for link) rather than unilateral by Corbo & Parkes at PODC‘05

- *n* peers $\{\pi_0, \ldots, \pi_{n-1}\}$ distributed in a metric space
  - defines distances (→ latencies) between peers
  - triangle inequality holds
  - Examples: Euclidean space, doubling or growth-bounded metrics, 1D line,…

- Each peer can choose to which other peer(s) it connects

- Yields a directed graph…



Model – The "Locality Game"

- Goal of a selfish peer:

  - Only little memory used
  - Small maintenance overhead

  (1) Maintain a small number of neighbors only (out-degree)

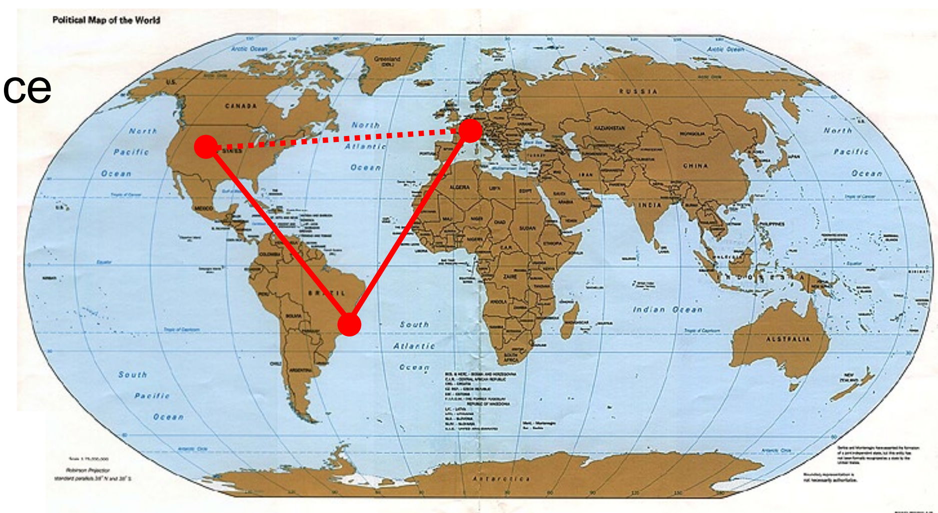
  (2) Small stretches to all other peers in the system

Fast lookups!

– Shortest path using links in G…

– … divided by shortest direct distance

LOCALITY!



Classic P2P trade-off!

Model – The "Locality Game"

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

$$cost_i = \alpha \cdot outdeg_i + \sum_{i \neq j} stretch_G(\pi_i, \pi_j)$$

- Goal of a peer: Minimize its cost!

- α is cost per link
- >0, otherwise solution is a complete graph

Model – Social Cost

- Social Cost is the sum of costs of individual peers
- System designer wants small social costs (-> efficient system)
- Social Optimum (OPT)
  - Topology with minimal social cost of a given problem instance
  - "topology formed by collaborating peers"!

- What topologies do *selfish peers* form?

→ Concepts of Nash equilibrium and Price of Anarchy

Model – Price of Anarchy

- Nash equilibrium
  - "Result" of selfish behavior → "topology formed by selfish peers"
  - Network where no peer can reduce its costs by changing its neighbor set given that neighbor sets of the other peers remain the same

- Price of Anarchy
  - Captures the impact of selfish behavior by comparison with optimal solution: ratio of social costs

$$PoA := \max_{I} \frac{NASH(I)}{OPT(I)}$$

What is the Price of Anarchy of our "*Locality Game*"?

Is there actually a Nash equilibrium…?

Related Work

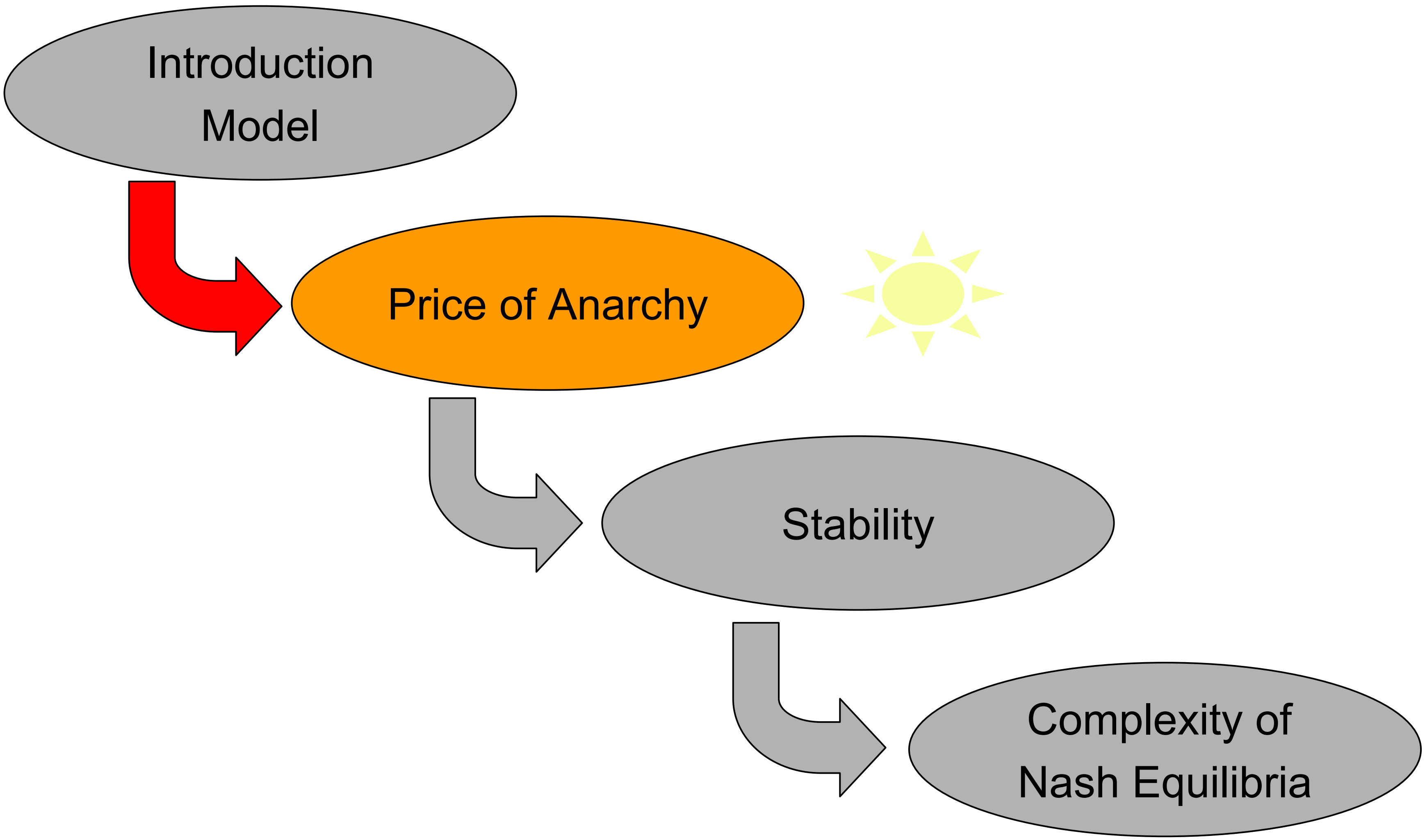
- The "*Locality Game*" is inspired by the "*Network Creation Game*"

- Differences:
  - In the Locality Game, nodes are located in a metric space

  → Definition of stretch is based on metric-distance, not on hops!
  - The Locality Game considers directed links

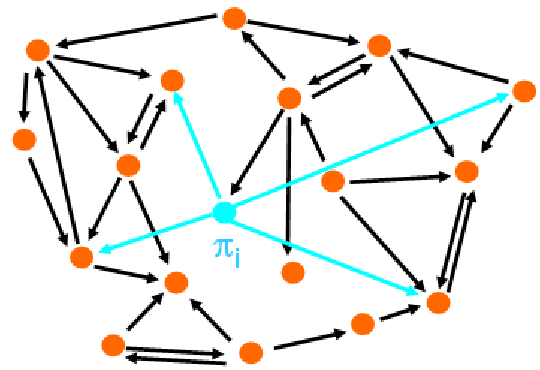
  → Yields new optimization function

Overview


Introduction
Model
Price of Anarchy
Stability
Complexity of
Nash Equilibria

Analysis: Lower Bound for Social Optimum?

$$cost_i = \alpha \cdot outdeg_i + \sum_{i \neq j} stretch_G(\pi_i, \pi_j)$$



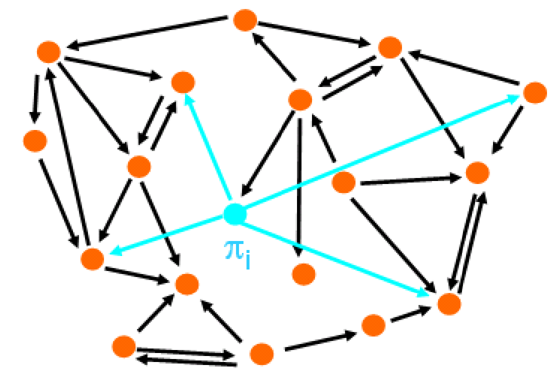
- Compute upper bound for PoA => need lower bound for social opt and an upper bound on Nash equilibrium cost

- OPT > ?
  - Sum of all the peers' individual costs must be at least?
  - Total link costs > ? (Hint: directed connectivity)
  - Total stretch costs > ?

Your turn! ☺

Analysis: Social Optimum

- For connectivity, at least *n* links are necessary
  → OPT $\geq \alpha$ n
- Each peer has at least stretch 1 to all other peers
  → OPT $\geq$ n · (n-1) · 1 = $\Omega(n^2)$



OPT $\in \Omega(\alpha$ n + n$^2)$

- Now: Upper Bound for NE? In any Nash equilibrium, no stretch exceeds α+1: total stretch cost at most O(α n^2)
  → otherwise it's worth connecting to the corresponding peer (stretch becomes 1, edge costs α)
- Total link cost also at most O(α n^2)

Really…?

NASH $\in$ O(αn^2)

Can be bad for large α

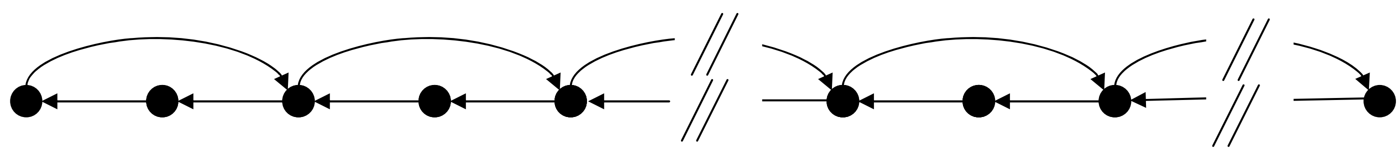
Price of Anarchy $\in$ O(min{α,n})

Analysis: Price of Anarchy (Lower Bound)

- Price of anarchy is tight, i.e., it also holds that

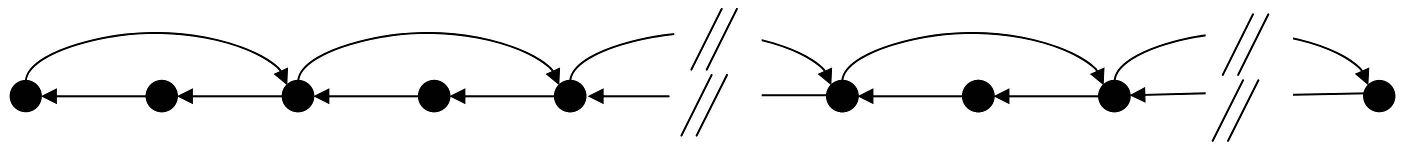
The Price of Anarchy is PoA $\in \Omega(\min\{\alpha, n\})$

- This is already true in a 1-dimensional Euclidean space:



Peer:	π_1	π_2	π_3	π_4	π_5	…	π_{i-1}	π_i	π_{i+1}	…	π_n
Position:	$\frac{1}{2}$	α	$\frac{1}{2}\alpha^2$	α^3	$\frac{1}{2}\alpha^4$	…	$\frac{1}{2}\alpha^{i-2}$	α^{i-1}	$\frac{1}{2}\alpha^i$	…	$\frac{1}{2}\alpha^{n-1}$

Analysis: Price of Anarchy (Lower Bound)



Peer:	π_1	π_2	π_3	π_4	π_5	…	π_{i-1}	π_i	π_{i+1}	…	π_n
Position:	½	α	½ α^2	α^3	½ α^4	…	½ α^{i-2}	α^{i-1}	½ α^i	…	½ α^{n-1}

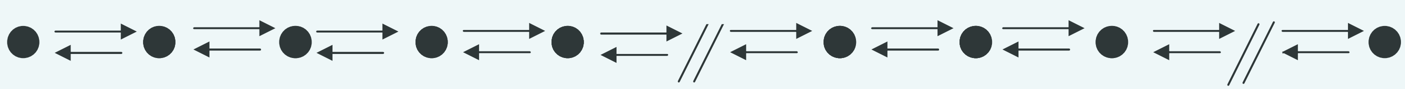
To prove:

(1) "is a selfish topology" = instance forms a Nash equilibrium

(2) "has large costs compared to OPT"

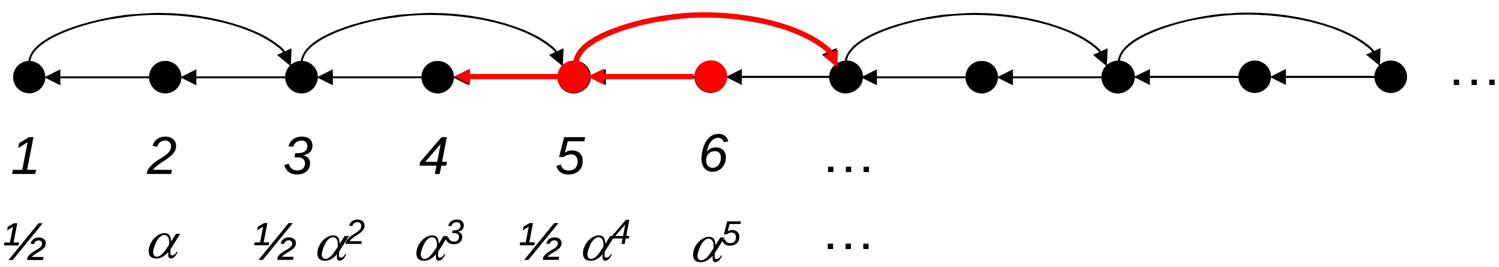
= the social cost of this Nash equilibrium is $\Theta(\alpha n^2)$

Note: Social optimum is at most $O(\alpha n + n^2)$:



$O(n)$ links of cost α, and all stretches = 1

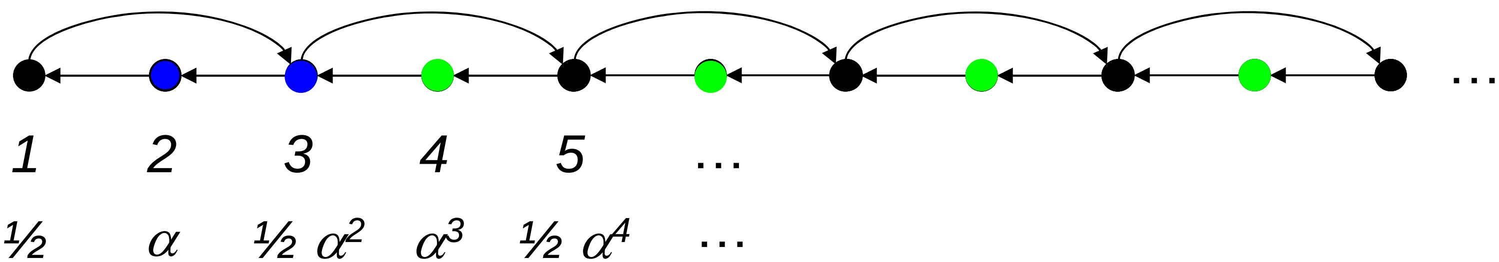
Analysis: Topology is Nash Equilibrium



- Proof Sketch: Nash?
  - Even peers:
    - For connectivity, at least one link to a peer on the left is needed (cannot change neighbors without increasing costs!)
    - With this link, all peers on the left can be reached with an optimal stretch 1
    - Links to the right cannot reduce the stretch costs to other peers by more than α
  - Odd peers:
    - For connectivity, at least one link to a peer on the left is needed
    - With this link, all peers on the left can be reached with an optimal stretch 1
    - Moreover, it can be shown that all alternative or additional links to the right entail larger costs

Analysis: Topology has Large Costs

- Idea why social cost are $\Theta(\alpha\, n^2)$: $\Theta(n^2)$ stretches of size $\Theta(\alpha)$



- The stretches from all odd peers *i* to a even peers *j>i* have stretch > $\alpha/2$

- And also the stretches between even peer *i* and even peer *j>i* *are* > $\alpha/2$

Analysis: Price of Anarchy (Lower Bound)

- Price of anarchy is tight, i.e., it holds that

The Price of Anarchy is PoA $\in \Theta(\min\{\alpha, n\})$

- This is already true in a 1-dimensional Euclidean space
- Discussion:

Need no incentive mechanism

→ For small α, the Price of Anarchy is small!

Need an incentive mechanism

→ For large α, the Price of Anarchy grows with n!

Example: Network with many small queries / files ->
latency matters, α large, selfishness can deterioate performance!

What about stability…?

- We have seen:

Unstructured p2p topologies may deteriorate due to selfishness!

- What about other effects of selfishness…?
- … selfishness can cause even more harm…!

Even in the absence of churn, mobility or other sources of dynamism, the system may never stabilize (i.e., P2P system may never reach a Nash equilibrium)!

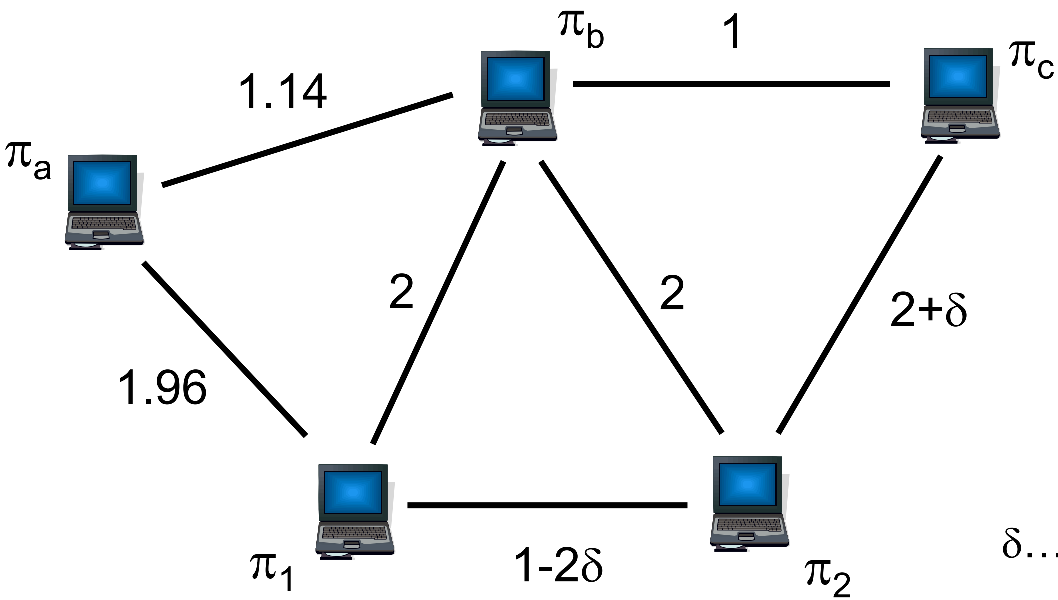
Overview

- Introduction
  Model
- Price of Anarchy
- Stability
- Complexity of Nash Equilibria

What about stability…?

- Consider the following simple toy-example
- Let α=0.6   (for illustration only!)

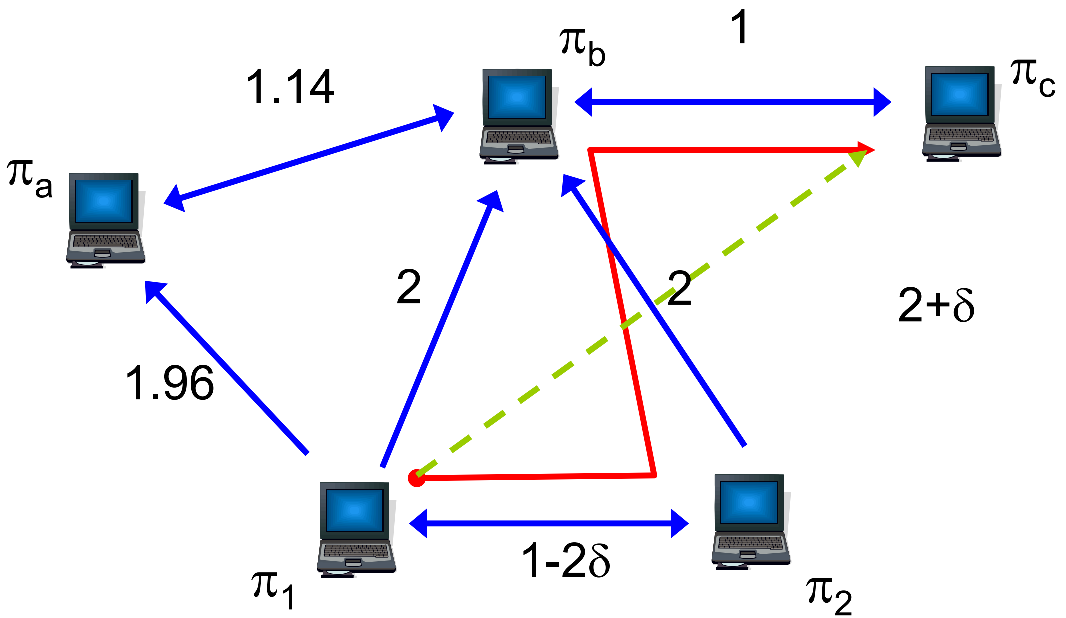
- 5 peers in Euclidean plane as shown below (other distances implicit)
- What topology do they form…?



δ…arbitrary small number

What about stability…?

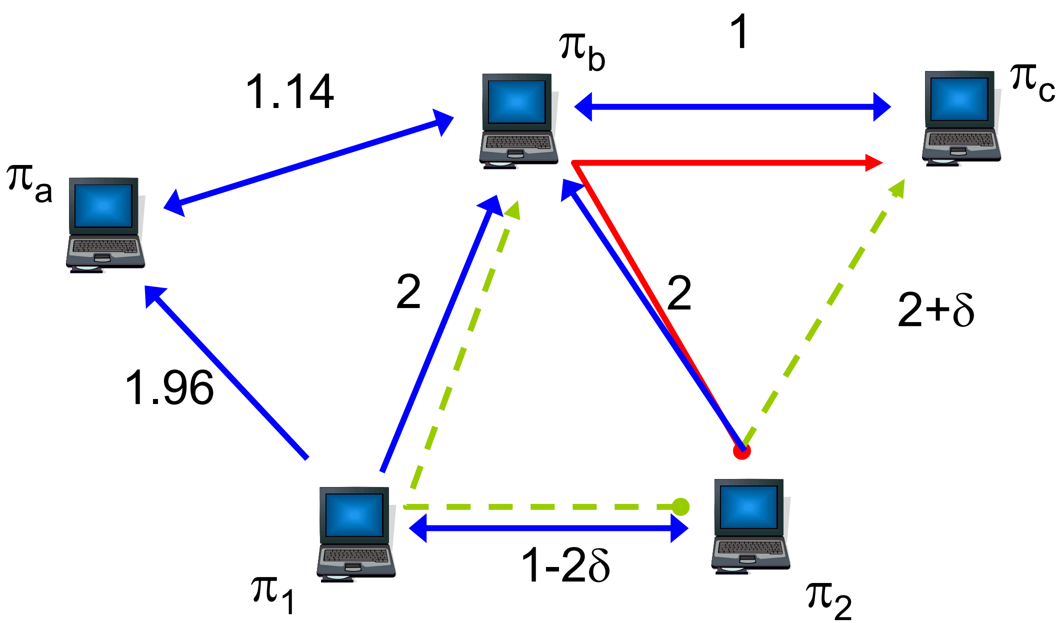
- Example sequence:
  - Bidirectional links shown must exist in any NE, and peers at the bottom must have directed links to the upper peers somehow: considered now! (ignoring other links)



$$\underbrace{\frac{1-2\delta+2+1}{d(\pi_1,\pi_c)}}_{\text{stretch}(\pi_1,\pi_c)}+\underbrace{\frac{1-2\delta+2}{2}}_{\text{stretch}(\pi_1,\pi_b)}>\alpha+1+\underbrace{\frac{3}{d(\pi_1,\pi_c)}}_{\text{stretch}(\pi_1,\pi_c)}$$

What about stability…?

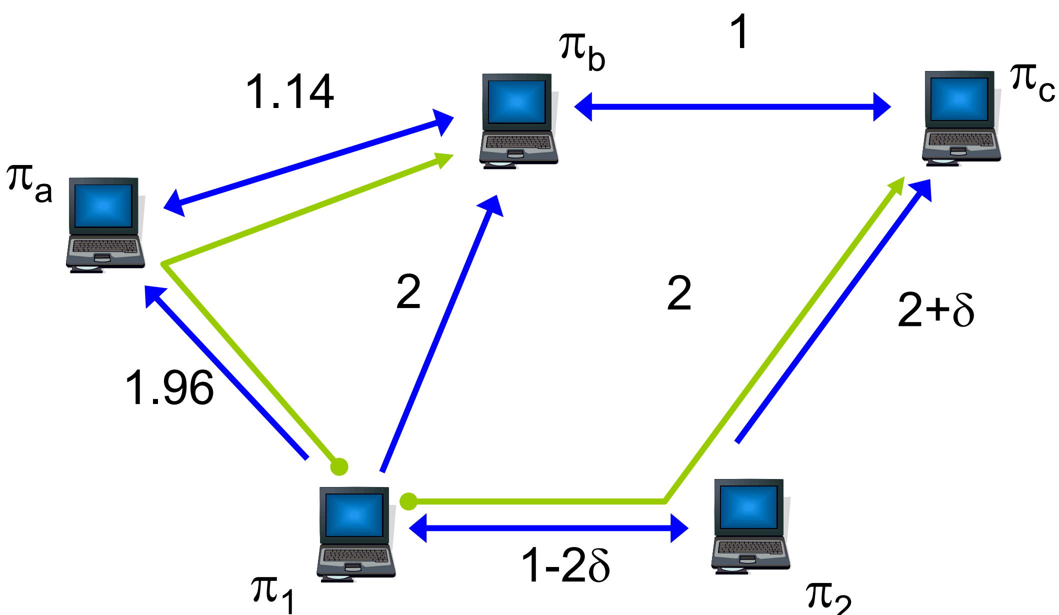
- Example sequence:



$$1 + \underbrace{\frac{1+2}{2+\delta}}_{\text{stretch}(\pi_2,\pi_c)} > \underbrace{\frac{1-2\delta+2}{2}}_{\text{stretch}(\pi_2,\pi_b)} + 1$$

What about stability…?

- Example sequence:

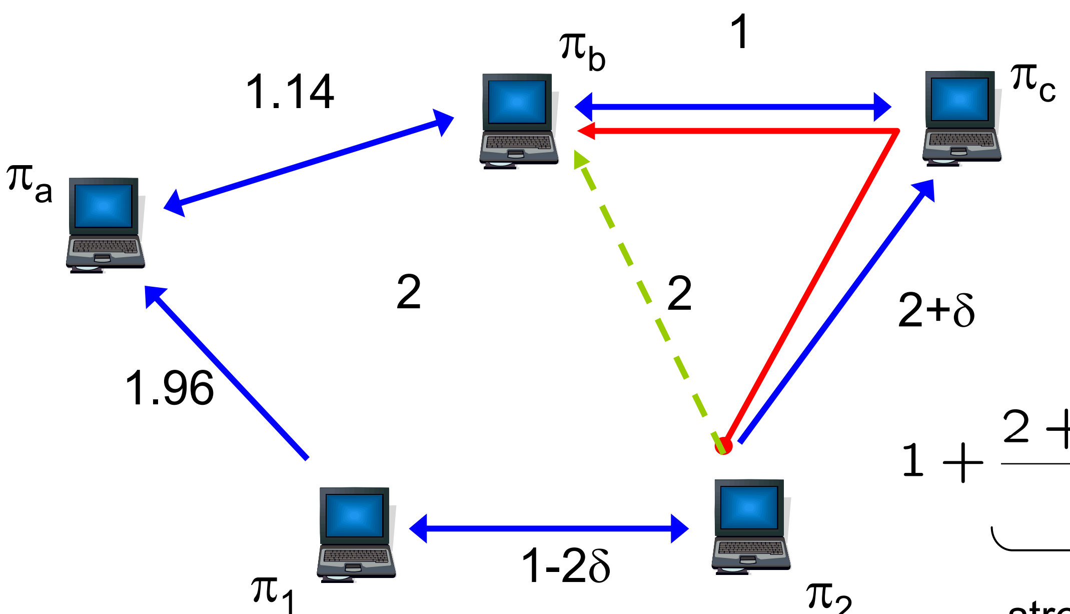


$$\alpha + 1 > \underbrace{\frac{1.96 + 1.14}{2}}_{\text{stretch}(\pi_1,\pi_b)}$$

What about stability…?

- Example sequence:

Again initial situation
→ Changes repeat forever!

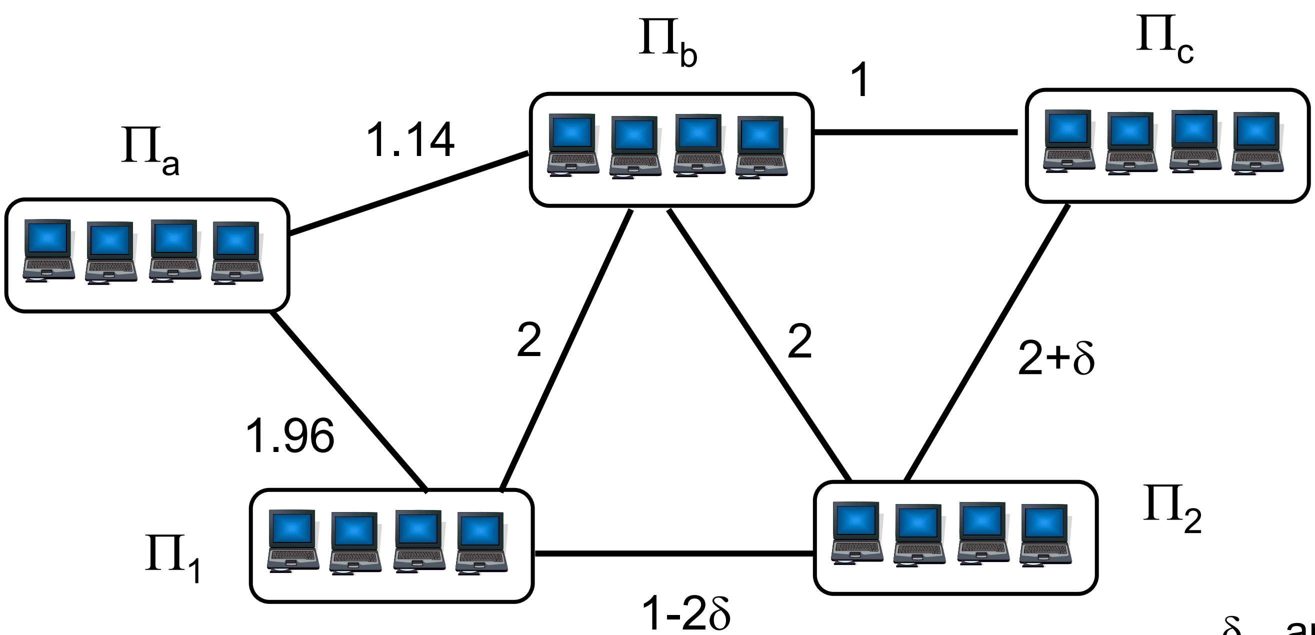


$$1 + \underbrace{\frac{2+\delta+1}{2}}_{\text{stretch}(\pi_2,\pi_b)} > 1 + \underbrace{\frac{3}{2+\delta}}_{\text{stretch}(\pi_2,\pi_c)}$$

Generally, it can be shown that for all α , there are networks, that do not have a Nash equilibrium → that may not stabilize!

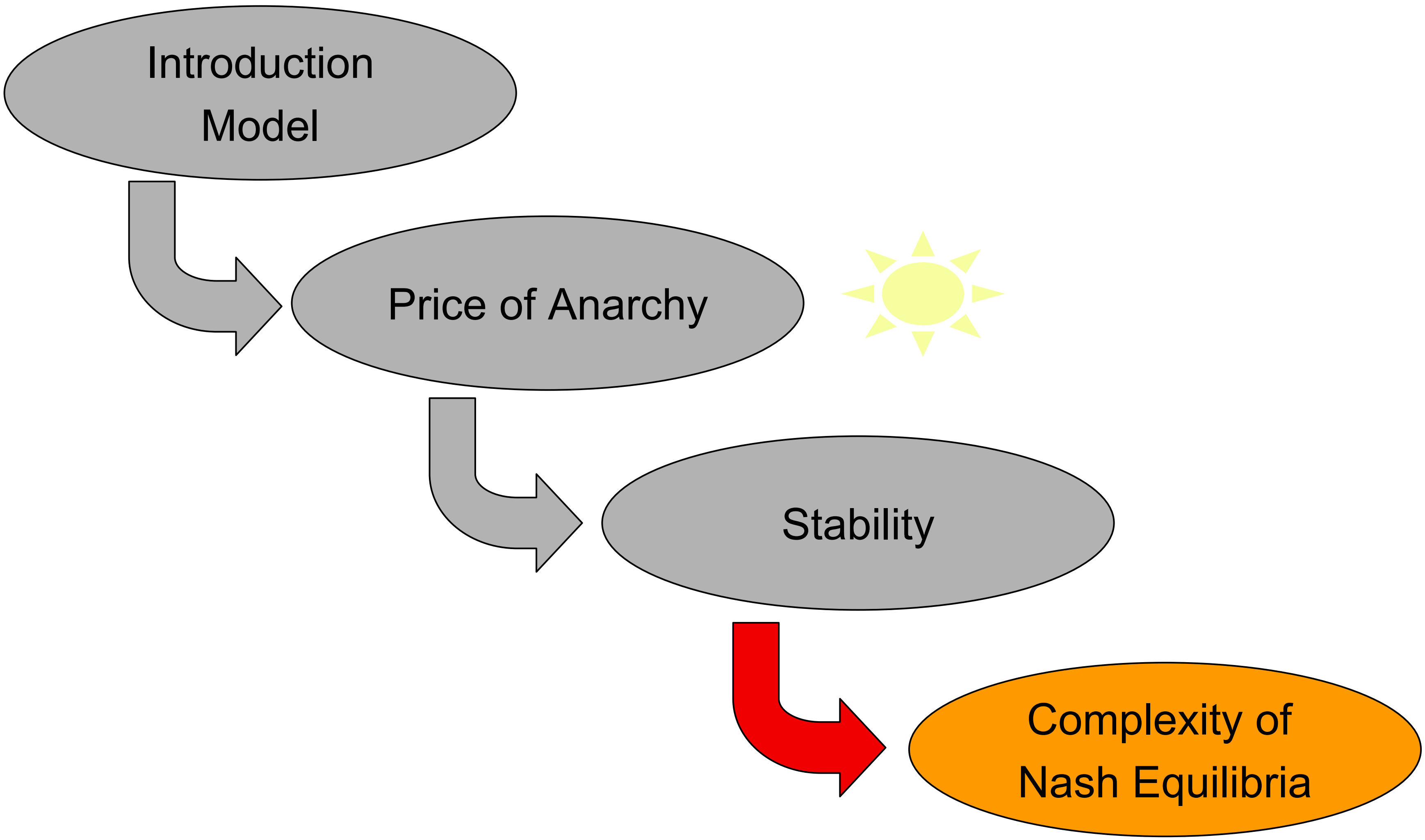
Stability for general α?

- So far, only a result for α=0.6
- With a trick, we can generalize it to all magnitudes of α
- Idea, replace one peer by a cluster of peers
- Each cluster has k peers → The network is instable for α=0.6k
- Trick: between clusters, at most one link is formed (larger α -> larger groups); this link then changes continuously as in the case of k=1.



δ…arbitrary small number

Overview


Introduction
Model
Price of Anarchy
Stability
Complexity of
Nash Equilibria

Complexity issues…

- Selfishness can cause instability!

  (even in the absence of churn, mobility, dynamism….)

- Can we (at least) determine whether a given P2P network is stable?

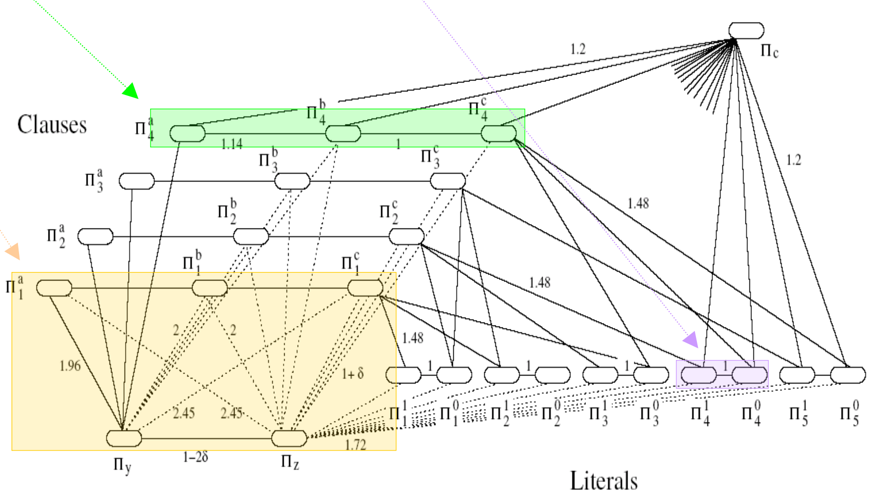
  (assuming that there is no churn, etc…)

→ What is the complexity of stability…???

Determining whether a P2P network has a (pure) Nash equilibrium is NP-hard!

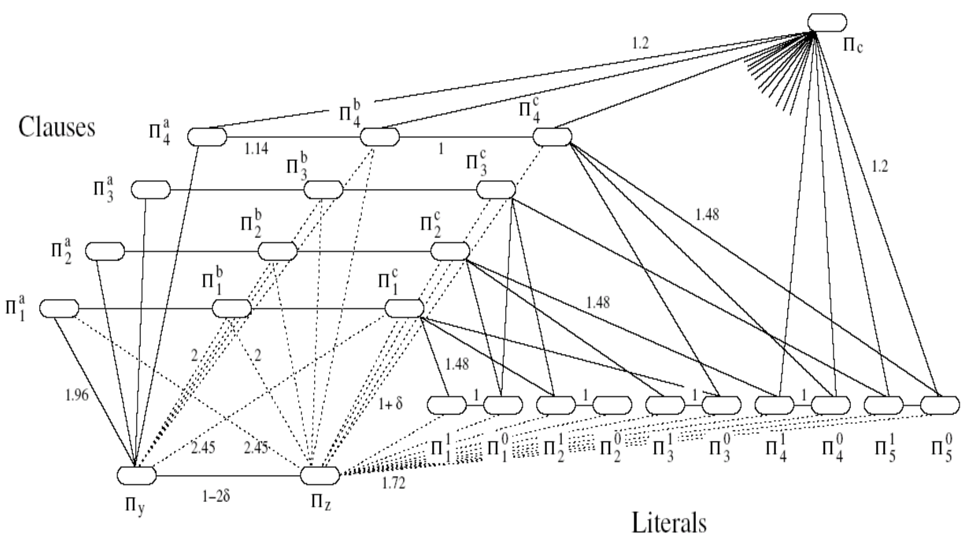
Complexity of Nash Equilibrium

- Idea: Reduction from 3-SAT in CNF form (each clause has 3 literals)
  - Proof idea: Polynomial time reduction: SAT formula -> distribution of nodes in metric space
  - If each clause is satisfiable -> there exists a Nash equilibrium
  - Otherwise, it does not.
  - As reduction is fast, determining the complexity must also be NP-hard, like 3-SAT!
  - (Remark: Special 3-SAT, each variable in at most 3 clauses, still NP hard.)
- Arrange nodes as below
  - For each clause, our old instable network! (cliques -> for all magnitudes of α!)
  - Distances not shown are given by shortest path metric
  - Not Euclidean metric anymore, but triangle inequality etc. ok!
  - Two clusters at bottom, three clusters per clause, plus a cluster for each literal (positive and negative variable)
  - Clause cluster node on the right has short distance to those literal clusters appearing in the clause!

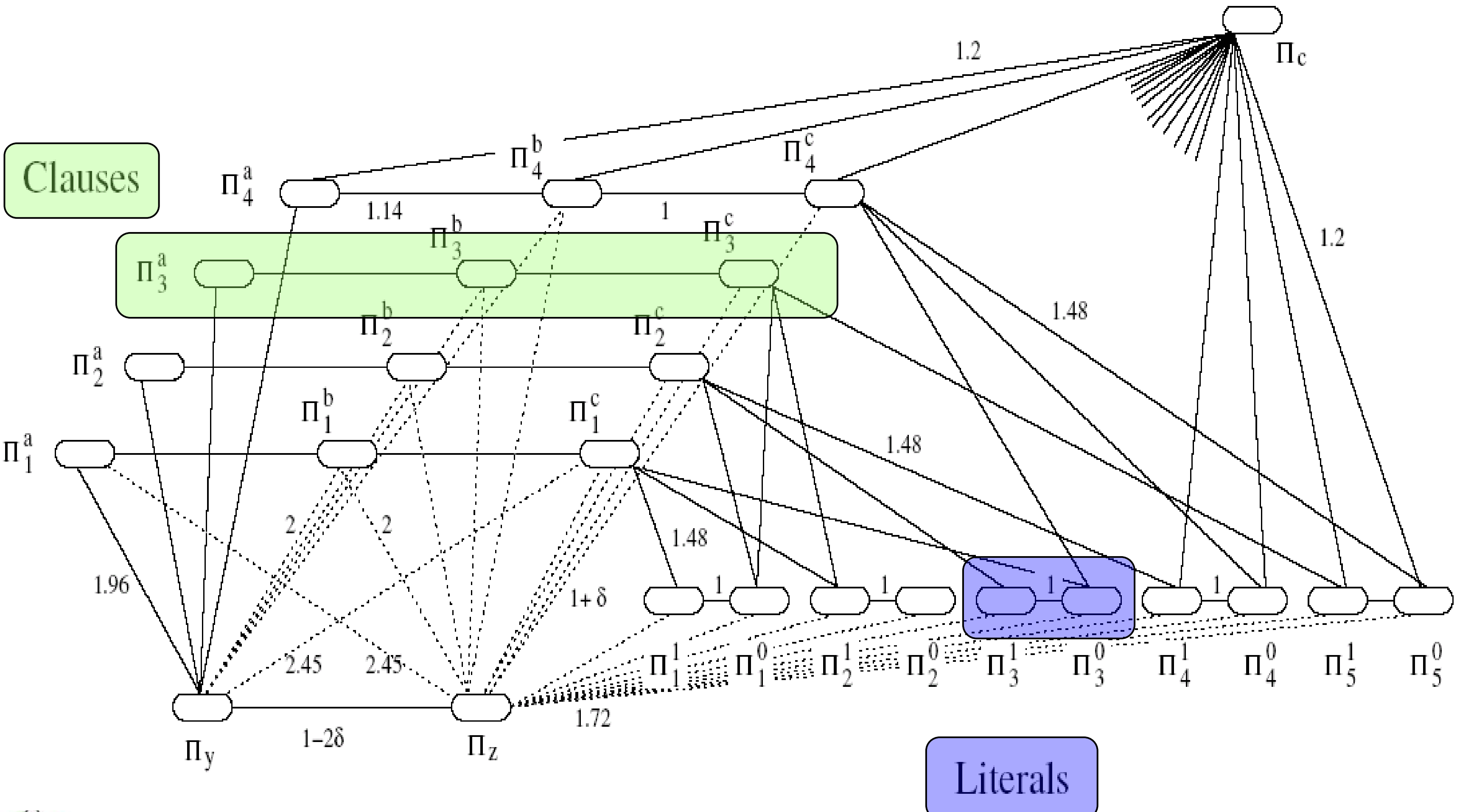


Complexity of Nash Equilibrium

- Main idea: The literal clusters help to stabilize!
  - short distance from Π^c (by construction), and <u>maybe</u> from Π_z
- The clue: Π_z can only connect to one literal per variable! (“assigment”)
  - Gives the satisfiable assignment making all clauses stable.
- If a clause has only unsatisfied literals, the paths become too large and the corresponding clause becomes instable!
  - Otherwise the network is stable, i.e., there exists a Nash equilibrium.

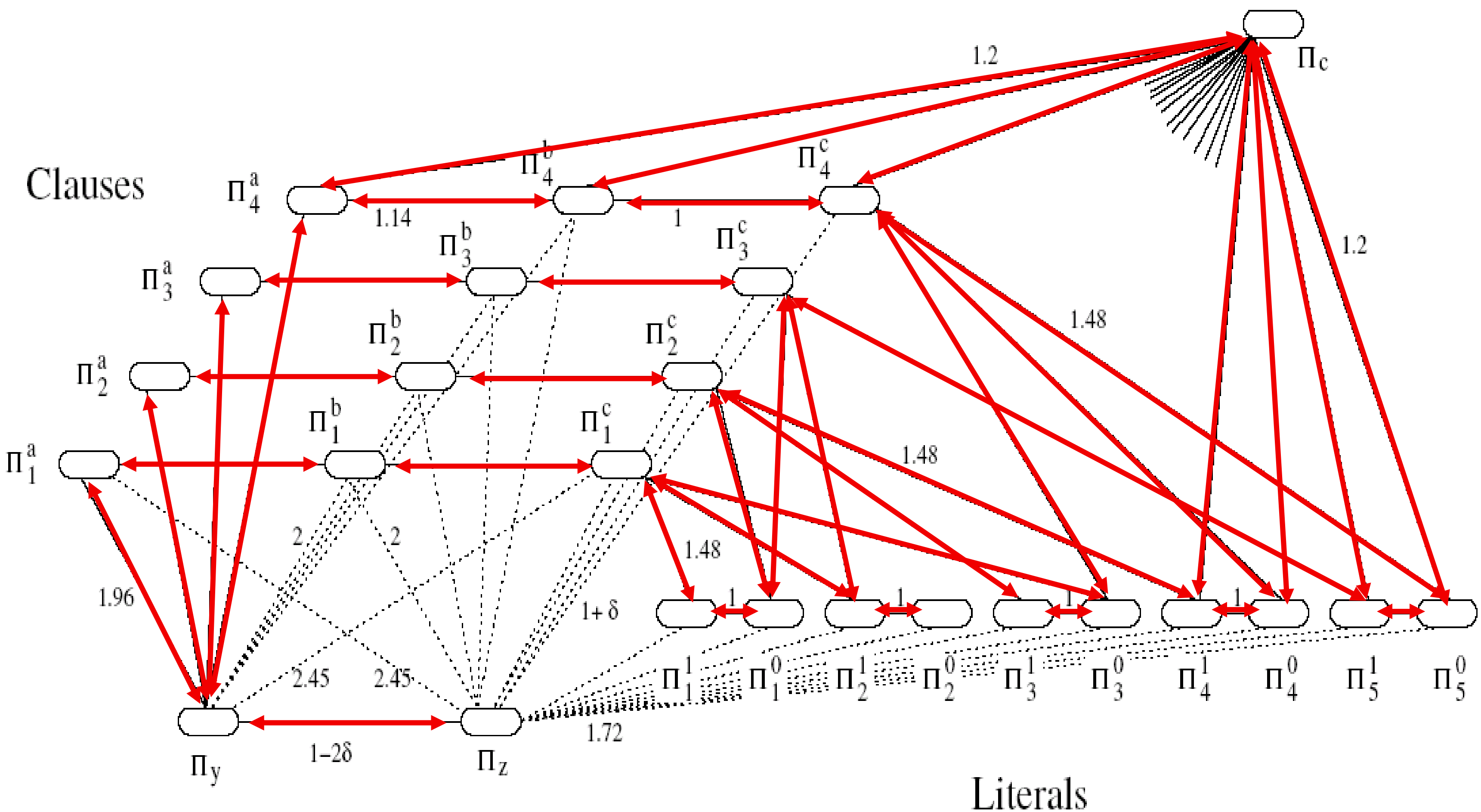


Complexity of Nash Equilibrium



Complexity of Nash Equilibrium

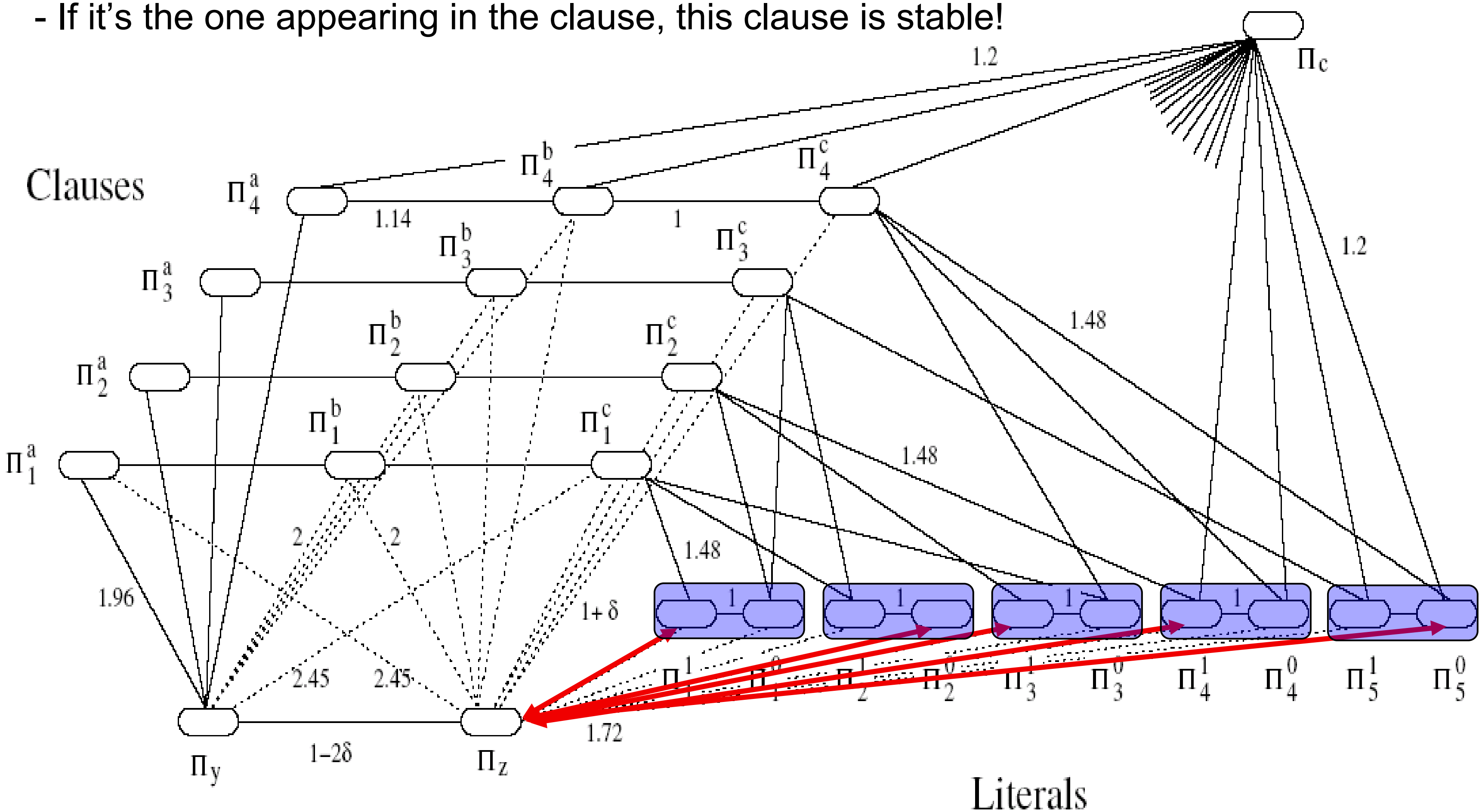
- It can be shown: In any Nash equilibrium, these links must exist…



Complexity of Nash Equilibrium

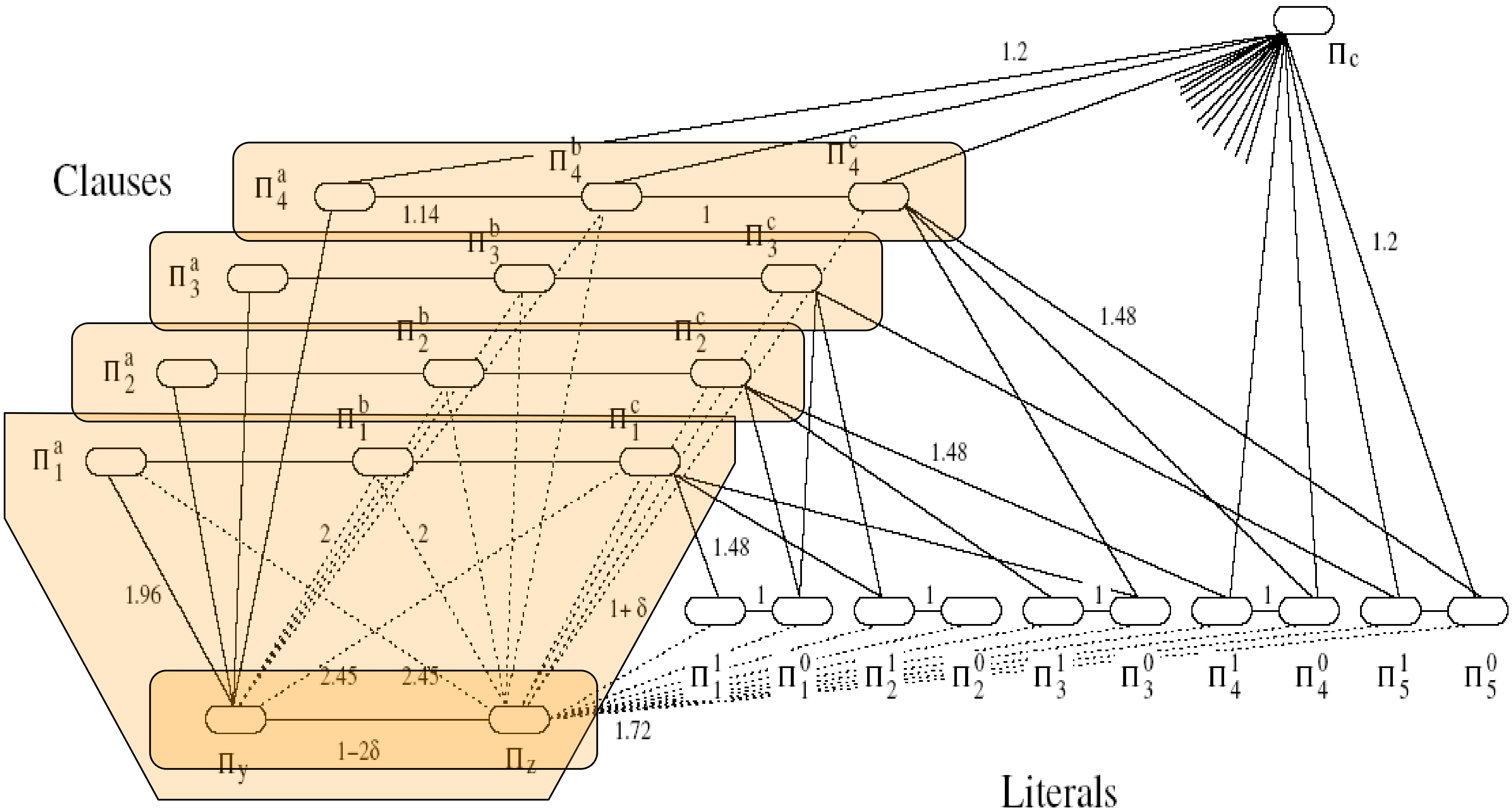
Special 3-SAT: Each variable in at most 3 clauses!

- Additionally, Π_z has exactly one link to one literal of each variable!
  - Defines the "assignment" of the variables for the formula.
  - If it's the one appearing in the clause, this clause is stable!



Complexity of Nash Equilibrium

- Such a subgraph (Π_y, Π_z, Clause) does not converge by itself…

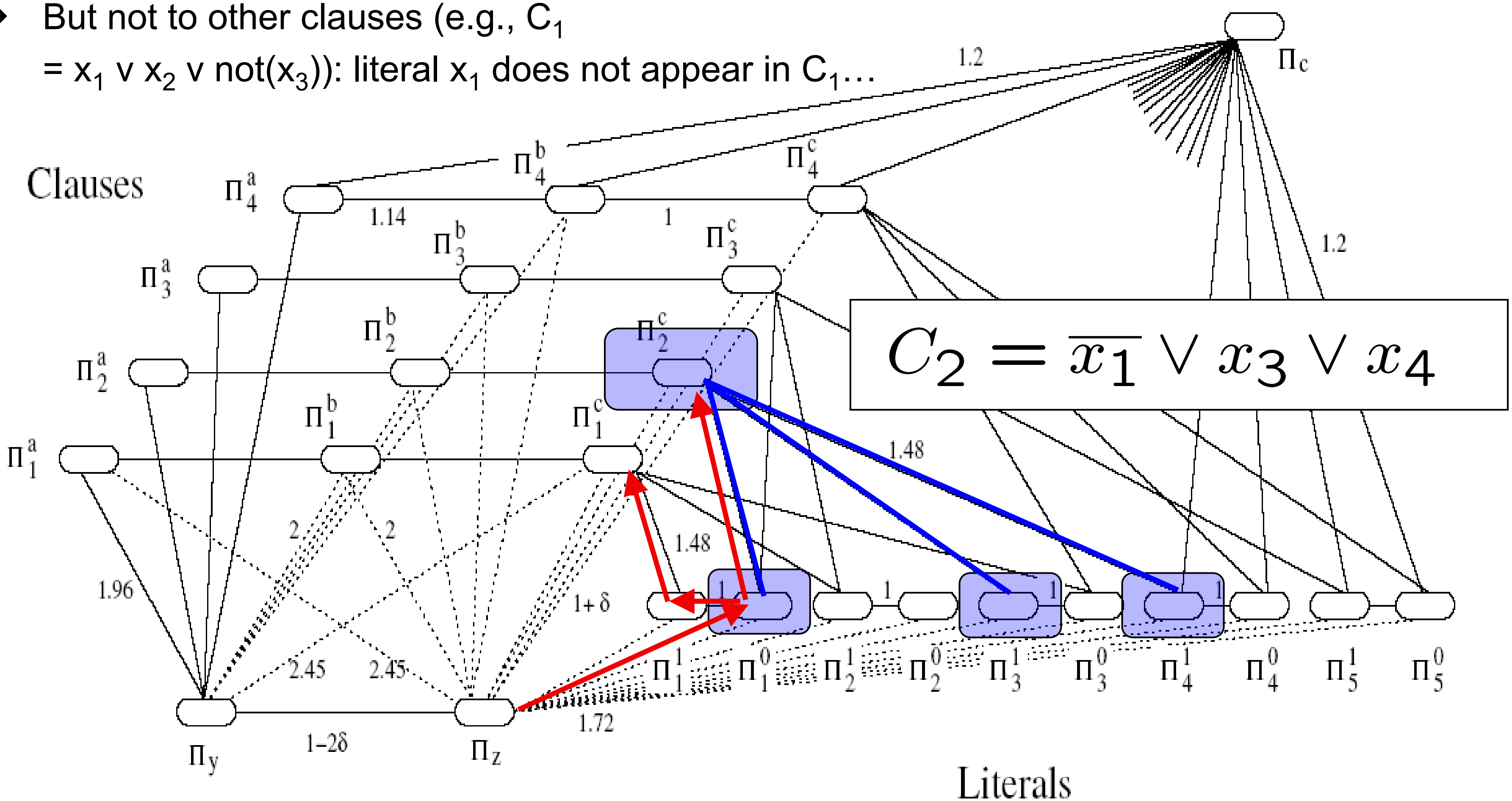


Complexity of Nash Equilibrium

- In NE, each node-set Π^c is connected to those literals that are in the clause (not to other!)

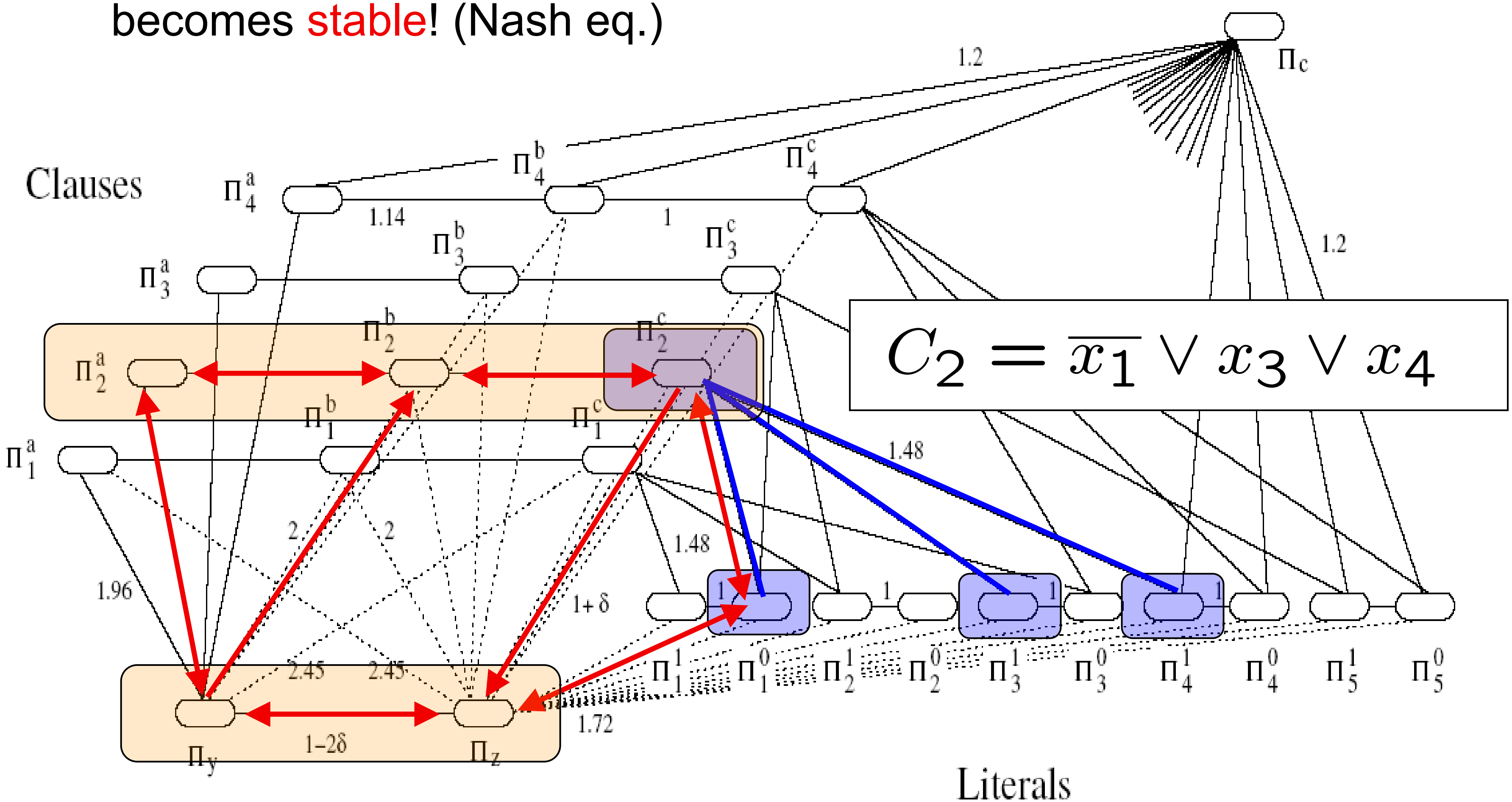
→ if Π_z has link to not(x1),
there is a "short-cut" to such clause-nodes, and C_2 is stable

→ But not to other clauses (e.g., $C_1 = x_1 \vee x_2 \vee not(x_3)$): literal x_1 does not appear in C_1…



Complexity of Nash Equilibrium

- A clause to which Π_z has a "short-cut" via a literal in this clause becomes stable! (Nash eq.)

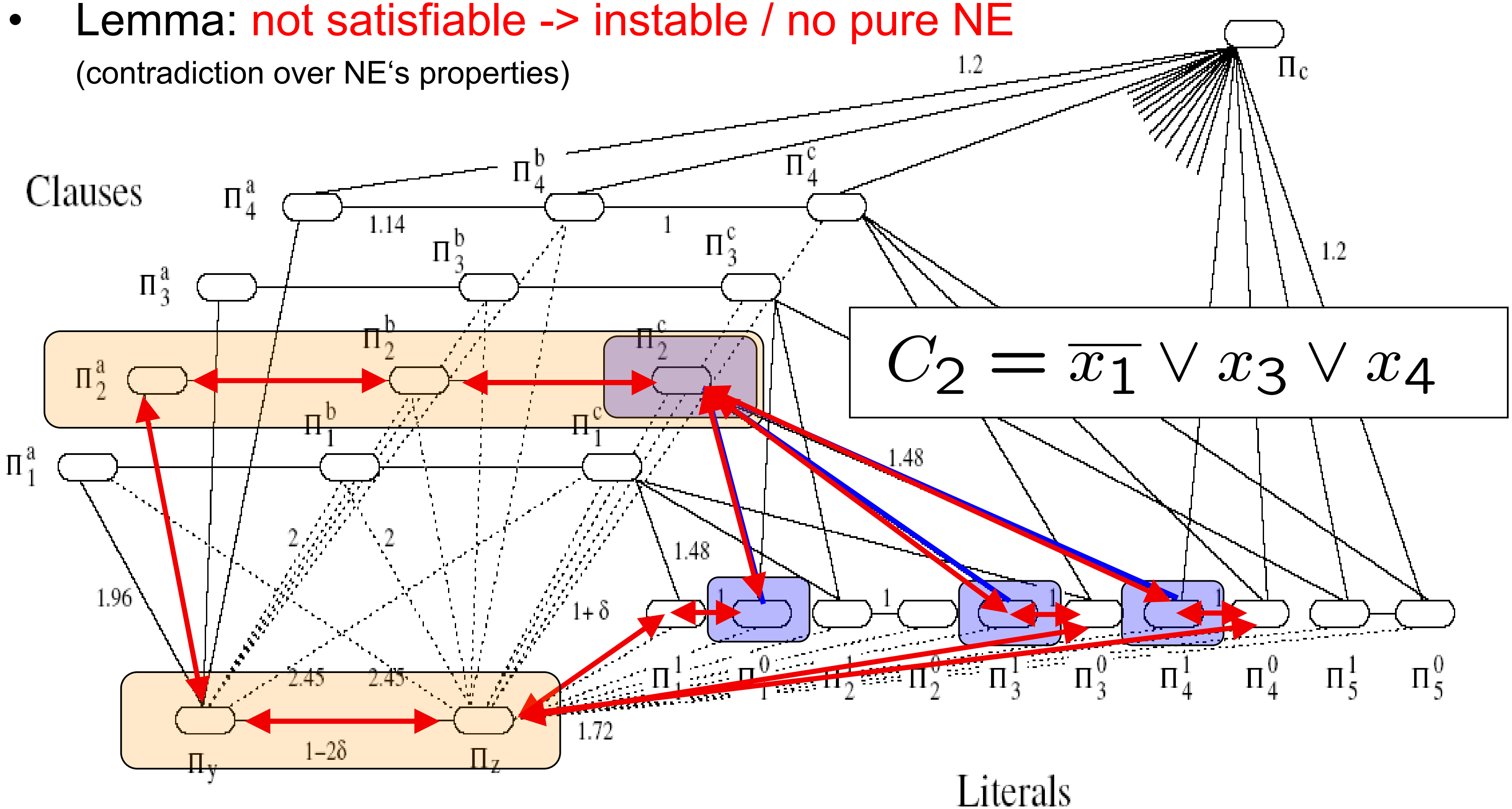


$$C_2 = \overline{x_1} \vee x_3 \vee x_4$$

Complexity of Nash Equilibrium

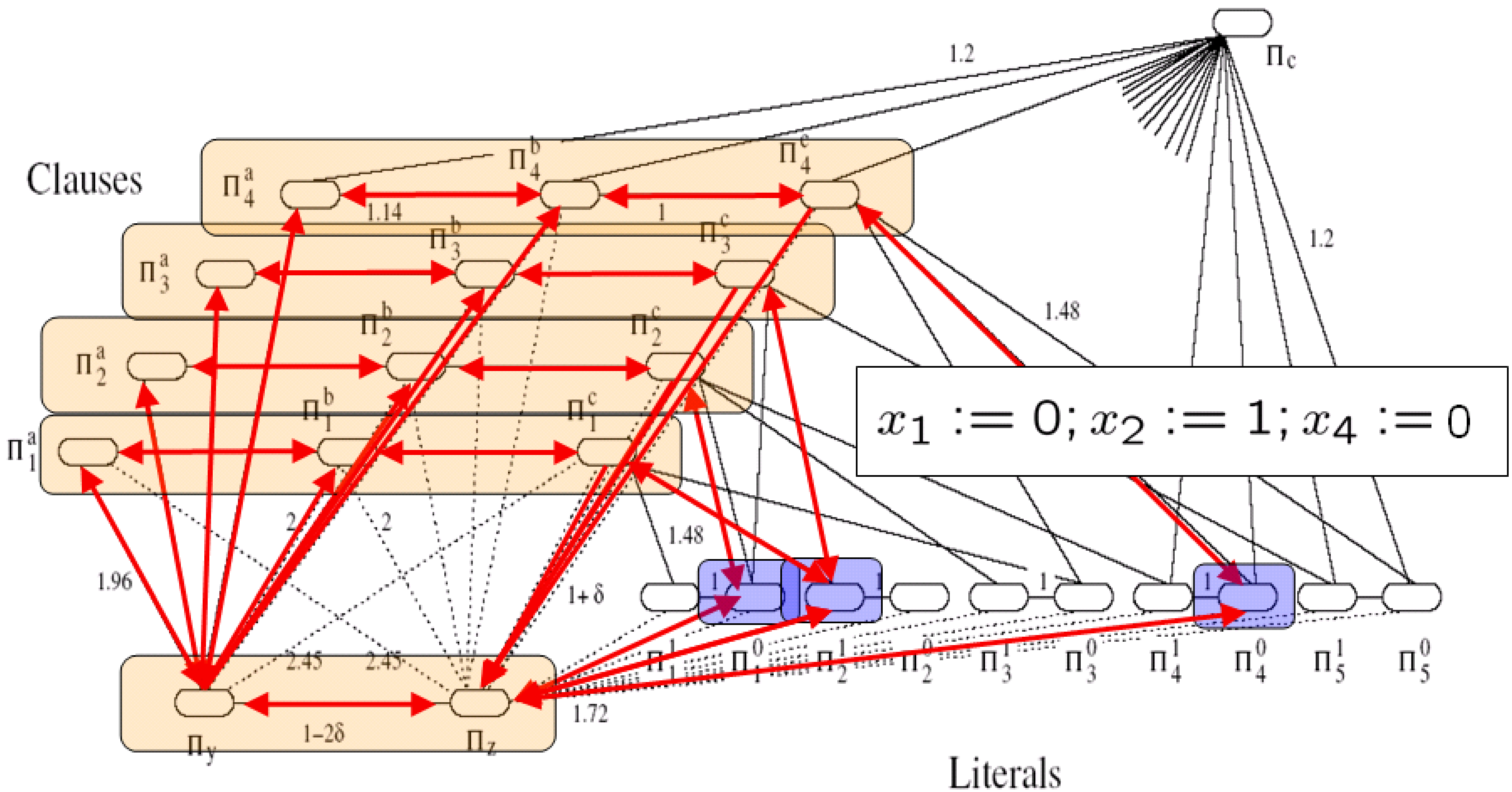
- If there is no such "short-cut" to a clause, the clause remains instable!
- Lemma: not satisfiable -> instable / no pure NE

(contradiction over NE's properties)



Complexity of Nash Equilibrium

- Example: satisfiable assignment -> all clauses stable -> pure NE



The Topologies formed by Selfish Peers

- 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?

Price of Anarchy $\in \Theta(\min\{\alpha,n\})$

Determining whether a P2P network has a (pure) Nash equilibrium is NP-hard!

Even in the absence of churn, mobility or other sources of dynamism, the system may never stabilize

Future Directions – Open Problems

- Nash equilibrium assumes full knowledge about topology!
  → this is of course unrealistic
  → incorporate aspects of local knowledge into model
- Current model does not consider routing or congestion aspects!
  → also, why should every node be connected to every other node?
  (i.e., infinite costs if not? Not appropriate in Gnutella or so!)
- Mechanism design: How to guarantee stability/efficiency..?
- More practical: what is the parameter α in real P2P networks?
- Lots more:
  - Algorithms to compute social opt of locality game?
  - Quality of mixed Nash equilibria?
  - Is it also hard to determine complexity for Euclidean metrics?
  - Computation of other equilibria
  - Comparisons to unilateral and bilateral games, and explanations?

Conclusion

- Peer-to-peer computing continues posing exciting research questions!

- **Dynamics:**
  - Measurements in practice? BitTorrent vs Skype vs Joost?
  - What are good models? Worst-case churn or Poisson model? Max-min algebra?
  - Relaxed requirements? Simulated topology may break, but eventually self-stabilize?
  - Other forms of dynamics besides node churn? Dynamic bandwidth?

- **Non-cooperation:**
  - Game-theoretic assumptions often unrealistic, e.g., complete knowledge of system's state (e.g., Nash equilibrium, or knowledge of all shortest paths)
  - Algorithmic mechanism design: How to cope with different forms of selfishness? Incentives to establish „good links“?
  - Social questions: Why are so many anonymous participants still sharing their resources?

Other Aspects of P2P Computing and Projects

Theory ⟷ Practice

- **Distributed Computation of the Mode**
  under submission
- **Event Detection and Efficient Aggregation**
  under submission
- **Selfish Throughput Maximization in Dynamic Networks**
  WICON 2006, HiPC 2006
- **Structured vs Unstructured P2P Systems**
  HiPC 2007
- Etc.

- **Attacks and Security in P2P Systems**
  SRDS 2006
- **P2P Live & On-demand Streaming**
  DISC 2007
- **Wuala File Sharing & Social Networking**
  Caleido Inc.
- Etc.

Thank you.

Thank you for your interest.

All presented papers can be found at:
http://dcg.ethz.ch/members/stefan.html