NMRG 2006

Econometric Promise Theory

part 2

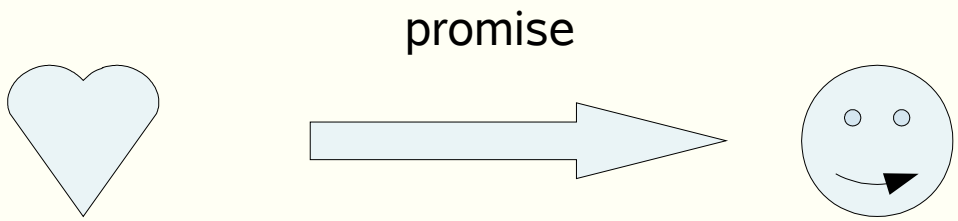
Mark Burgess

oslo university college

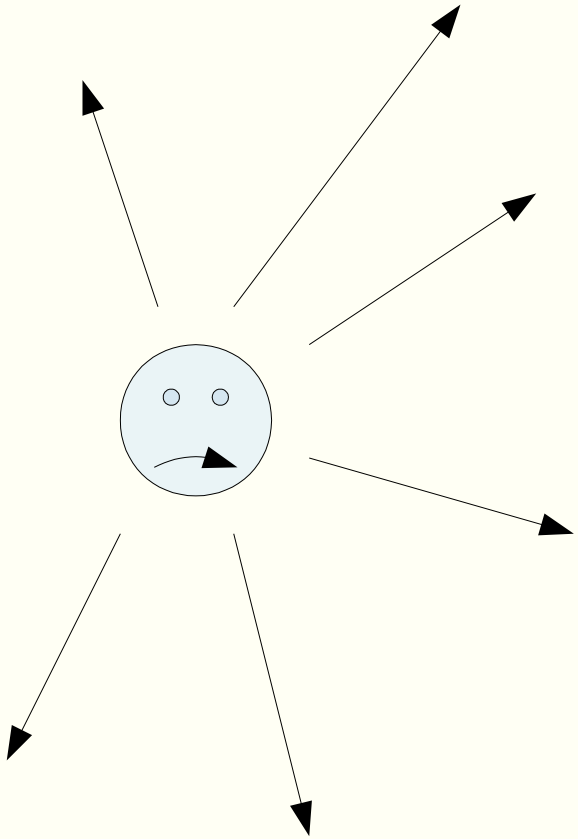
Living with uncertainty

Promises have value

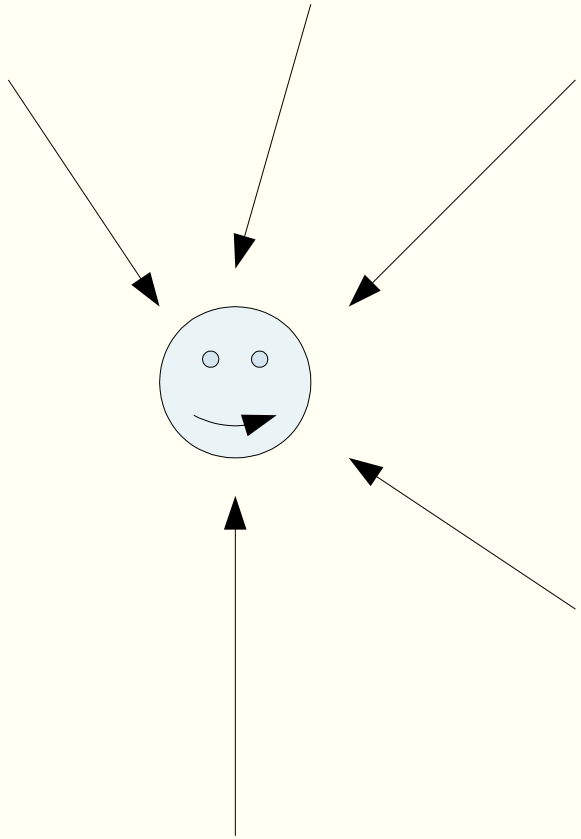
- In terms of what is received
- In terms of what it costs to implement
- Involves an exchange of trust
  - Can a promise be exploited?



Exploited or spoiled?



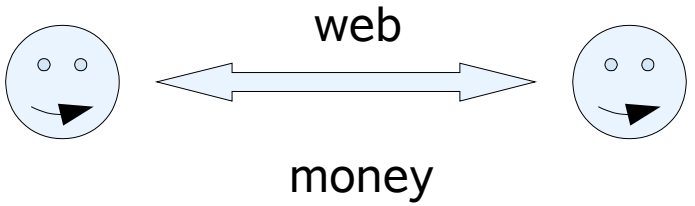
Cost



Reward

Value is important in autonomy

- The nodes can do whatever they please
- Why should they make/keep their promises?
- Give away value – want something in return?
  - What is the currency of exchange?
  - e.g. promise me web-service, promise you money
  - e.g. promise to forward your packets (both ways)
  - Reliability



Cooperation - bargaining

- Bargaining or trade of valuable promises is a basis for understanding the probability of cooperative behaviour.

- **Cooperative dilemma**: do we or don't we?
  - *Autonomy: why should I?*
  - *"You scratch my back, I'll scratch yours"*

- **Cooperation**: obey policy and keep promises

- **Defection**: fail to obey policy

Game theory

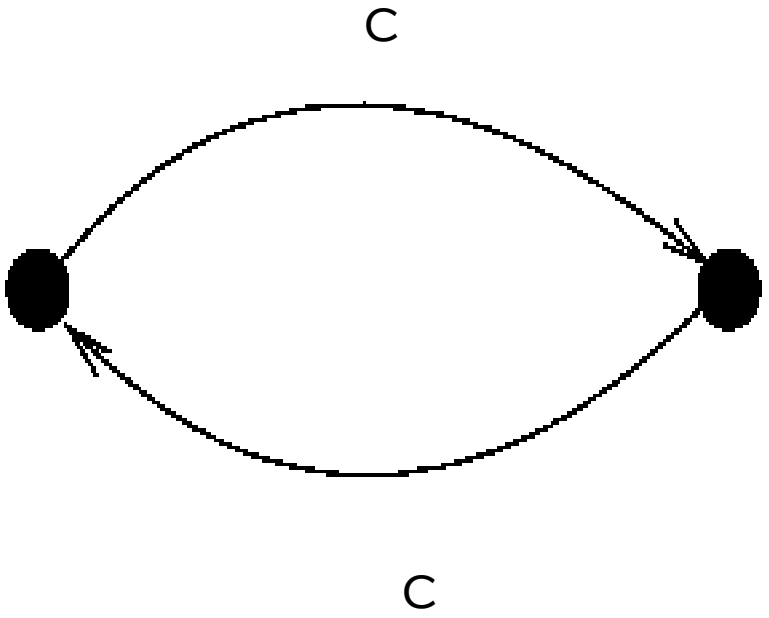
- Economics and bargaining are described using game theory
  - Rational agents, base judgement on perceived value
  - Selfish (autonomous) individuals, place their own gains first
- Archetypal example
  - Prisoner's dilemma
  - Bargaining games (Nash equilibrium)

Multi-agent systems

- Have “commitments”
- The idea seems to be like promises, except
  - A model of distributed computation
  - Task oriented
  - More like programming
  - More about dependency and delegation than autonomy
- This is not a model of voluntary cooperation
- Has no notion of value judgement

Promises and games

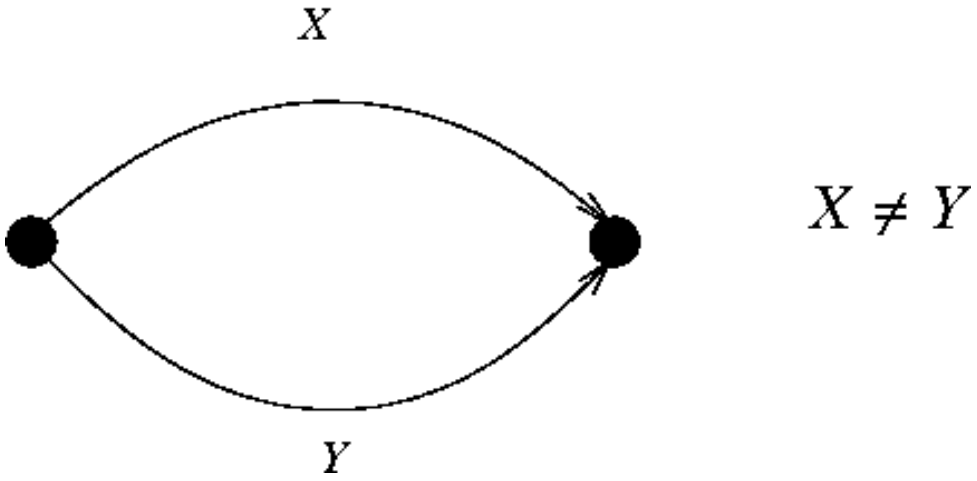
- A 2 player game involves moves and responses by its players
- Two choices: keep or break promise
  - Cooperate / Defect



	C_1	D_1
C_2	(R,R)	(S,T)
D_2	(T,S)	(P,P)

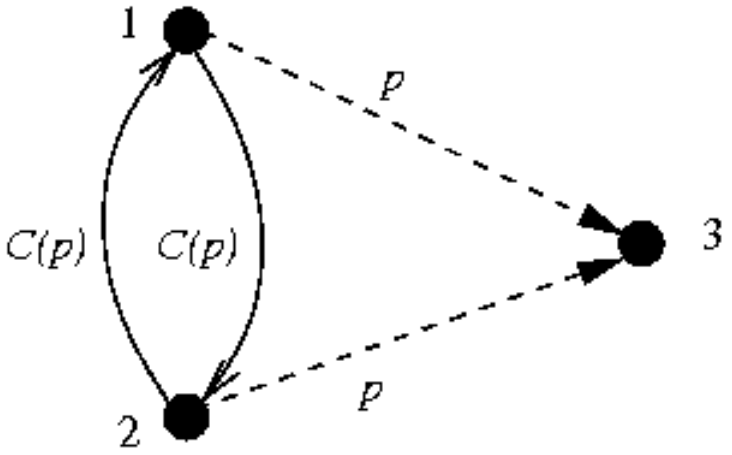
Typed/labelled graph

- Promise types:
  - Service promises (promise to constrain behaviour)
  - Cooperative promise (promise to do the same as)
  - Usage promise (promise to make use of)
- Atomicity rule:
  - Only *one* promise of a given type per pair:
  - Broken promise => two different promises



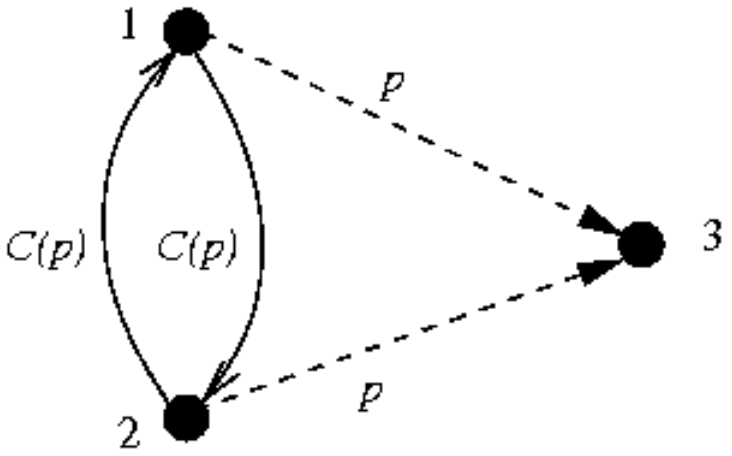
Cooperation and 3rd parties

- When two nodes agree to cooperate C(p) it can be viewed as something that can be verfied by a third party – or monitor
- Trust is a form of valuation of agreement
- Adjudicator = 3rd party



Roles and 3rd parties

- Works both ways: pledge allegiance to a 3rd party also implies local cooperation.
- Thus common promises to an external agent imply harmonization of **roles**
- Define a role
  - Appointed role (observer)
  - Cooperative role (allegiance)
- Roles can tell us a lot
- (*Hold this thought)*



How is value measured?

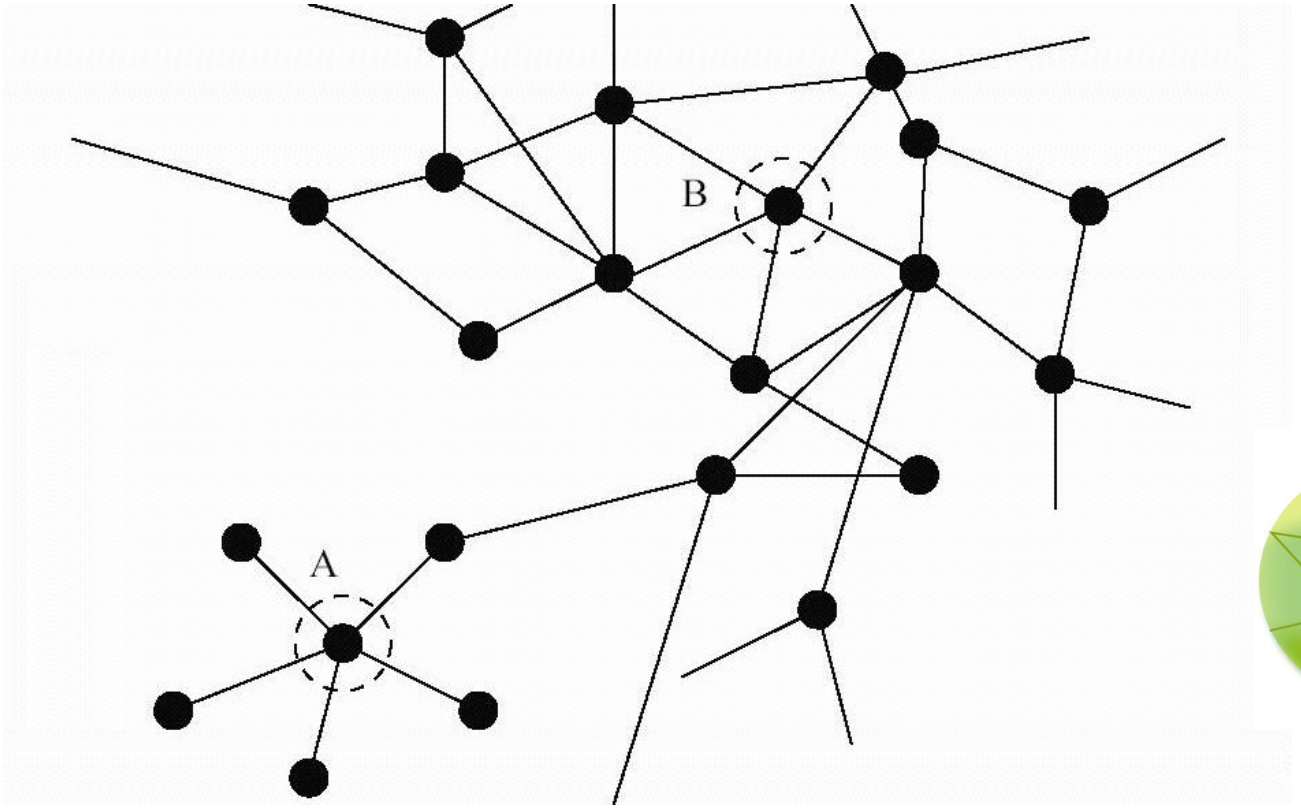
- Promises are initially *typed constraints*
- The currency of value transfer is a ***function*** of the constraint – what does it mean to the agent
  - Different agents can measure differently
  - Local policy determines the importance
- Global measures with respect to an imaginary third party can be computed using graph theory
  - Centrality <-> objective to external observer
  - Topological valuations <-> reliability
- Common currency graph

Example: BGP

- Autonomous peer system
  - Access promises
  - Transit promises
- Peering agreements
  - “Once a customer, never a peer”
  - (See W.B. Norton analysis of peering)

Social importance - Centrality

- Measure of scalar importance, based on social importance – global statistical **roles**
- Prototype tool for computing - Archipelago



Archipelago

Adjacency matrix

- Adjacency matrix contains the whole structure of the graph

$$A_{ij} = \begin{array}{c} \\ \text{A} \\ \text{B} \\ \text{C} \end{array} \begin{array}{c} \begin{array}{ccc} \text{A} & \text{B} & \text{C} \end{array} \\ \begin{pmatrix} 0 & 1 & 0 \\ 1 & 0 & 1 \\ 0 & 1 & 0 \end{pmatrix} \end{array}$$

- Use two characteristics that relate to security and reliability
  - Eigenvector decomposition (importance)
  - Percolation (connectivity)

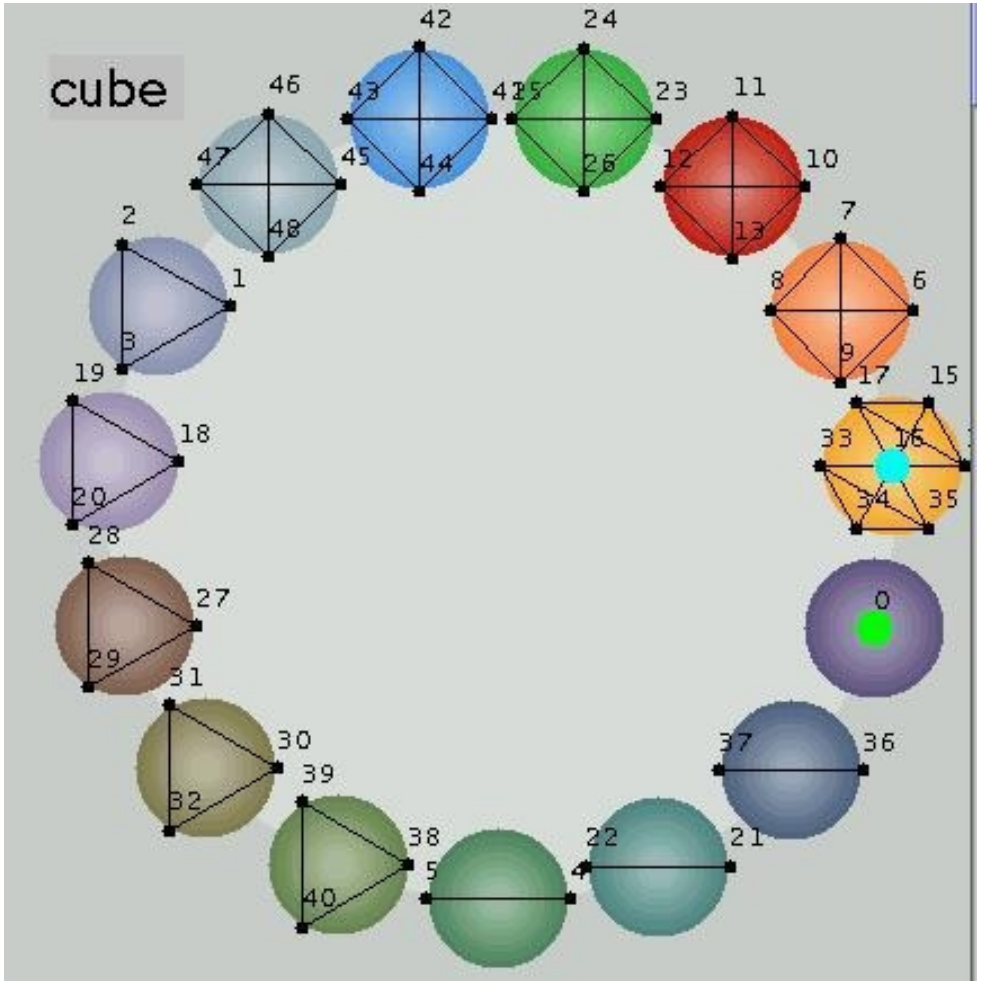
Eigenvector centrality

- Matrix *A* sums neighbours recursively
- Gives eigenvector equation
  - Principal eigenvector = centality

$$A\,\vec{v} = \lambda\vec{v}$$

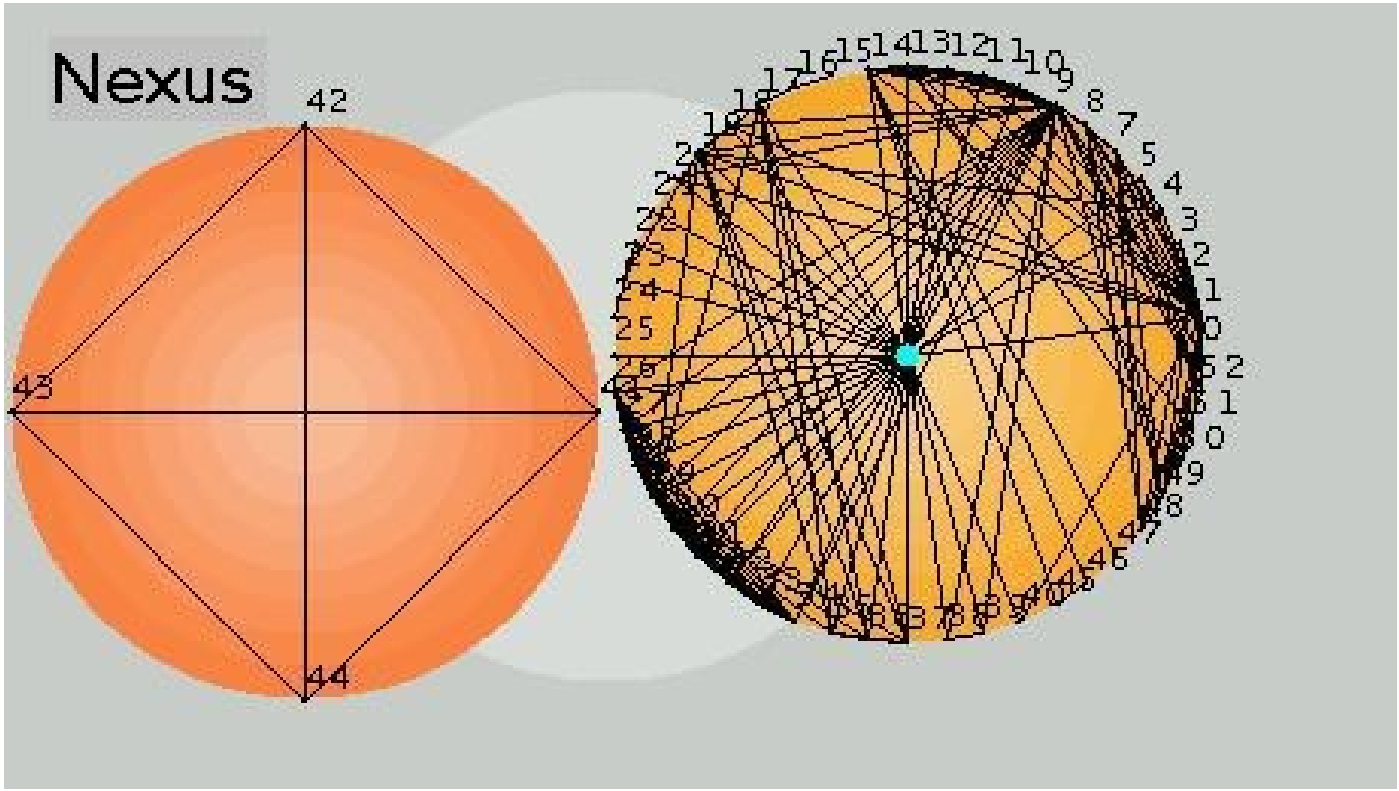
Scan of student system

Isolated work groups with autonomous cooperation



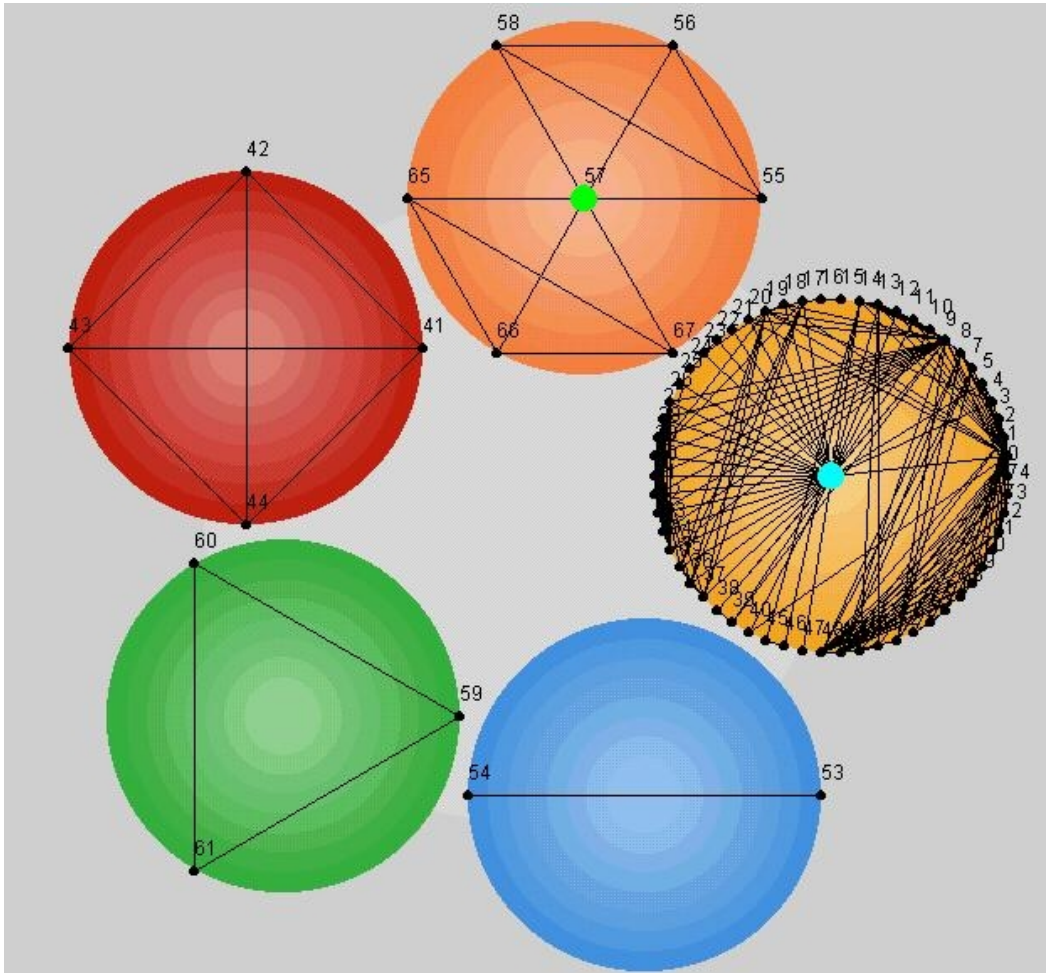
Scan of staff system

Staff “trust” each other far more (distrust only students!)

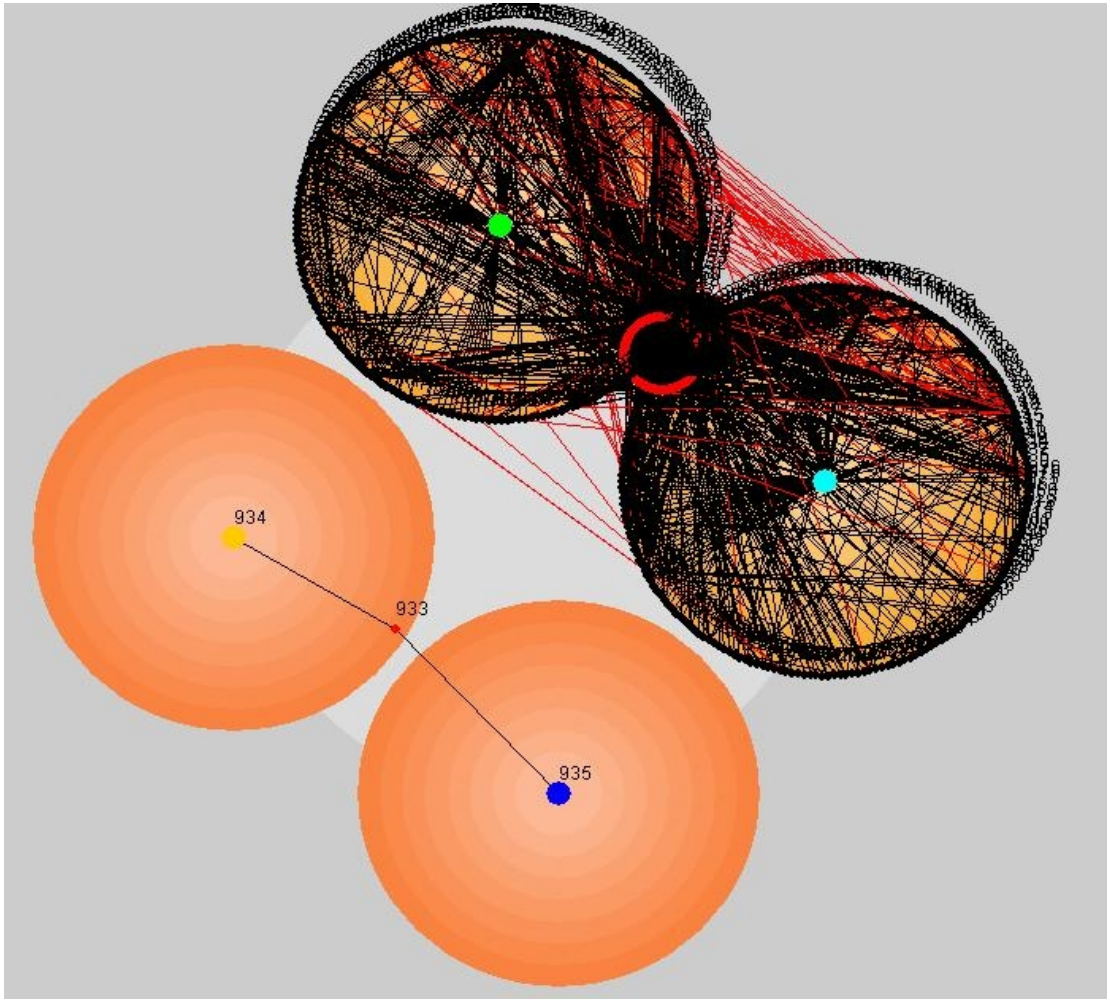


Staff + student system

Implicit links in previously separate groups



Gnutella peer-to-peer



Summary

- Promises (as games) describe steady equilibria, not causal development
- Cooperative agreement builds stability
- Common currency graph <-> reliability
- Warfare in peering promises – experience BGP
  - Predict these problems before they arise
  - Determine a policy to minimize uncertainty