The Role of Trust in Collaborative and Adversarial Behavior in Networks of Autonomous Agents

John S. Baras
Institute for Systems Research,
Department of Electrical and Computer Engineering
Department of Computer Science,
Fischell Department of Bioengineering
University of Maryland College Park

March 23, 2009
AFOSR Workshop
Adversarial and Stochastic Elements in Autonomous Systems

Outline

- Networks and Collaboration
  Constrained Coalitional Games
- Trust and Networks
- Security Aware Protocols via NUM
- Trust and Distributed Estimation
- Topology Matters
- Conclusions and Future Directions

A Network is …

- A collection of nodes, agents, …

  that **collaborate** to accomplish actions, gains, …

  that cannot be accomplished with out such collaboration

- Most significant concept for **dynamic autonomic networks**

The Fundamental Trade-off

- The nodes gain from collaborating
- But collaboration has costs (e.g. communications)
- **Trade-off: gain from collaboration *vs* cost of collaboration**

  Vector metrics involved typically

  ⇒ Constrained Coalitional Games

- Example 1: Network Formation -- Effects on Topology
- Example 2: Collaborative robotics, communications
- Example 3: Web-based social networks and services
- Example 4: Groups of cancer tumor or virus cells

● ● ●

Gain

- Each node potentially offers benefits V per time unit to other nodes: e.g. V is the number of bits per time unit.
- Potential benefit V is reduced during transmissions due to transmission failures and delay
- Jackson-Wolingsky connections model, gain of node *i*

$$w_i(G) = \sum_{j \in g} \delta^{r_{ij}-1} V$$

- r_{ij} is # of hops in the shortest path between i and j

$r_{ij} = \infty$ if there is no path between

- $0 \le \delta \le 1$ is the *communication depreciation rate*

Cost

- Activating links is costly
  - Example – cost is the energy consumption for sending data
  - Like wireless propagation model, cost c_{ij} of link ij as a function of link length d_{ij} :

$$c_{ij} = P d_{ij}^{\alpha}$$

    - P is a parameter depending on the transmission/receiver antenna gain and the system loss not related to propagation
    - α is path loss exponent -- depends on specific propagation environment.

Pairwise Game and Convergence

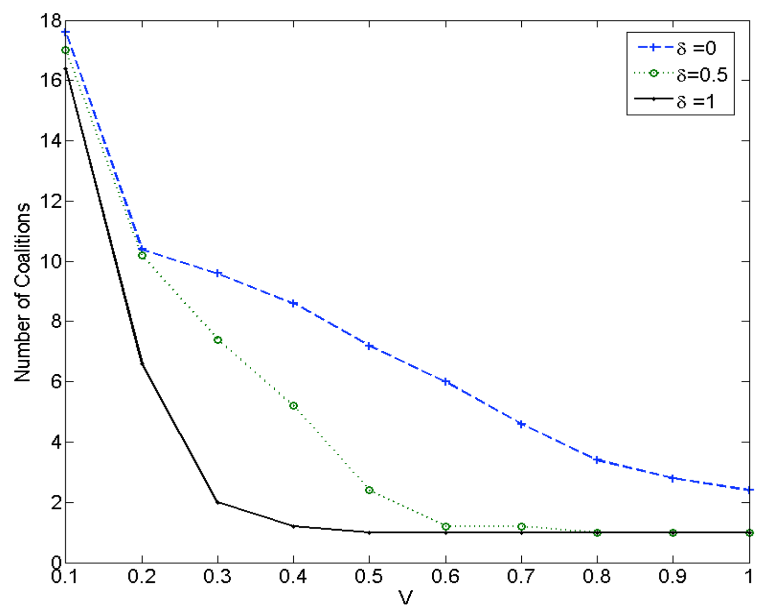
- Payoff of node i from the network G is defined as

$$v_i(G) = gain_i(G) - cost_i(G)$$

- Iterated process
  - Node pair ij is selected with probability p_{ij}
  - If link ij is already in the network, the decision is whether to sever it, and otherwise the decision is whether to activate the link
  - The nodes act myopically, activating the link if it makes each at least as well off and one strictly better off, and deleting the link if it makes either player better off
  - **End**: if after some time, no additional links are formed or severed
  - **With random mutations** , the game converges to a unique Pareto equilibrium (underlying Markov chain states)

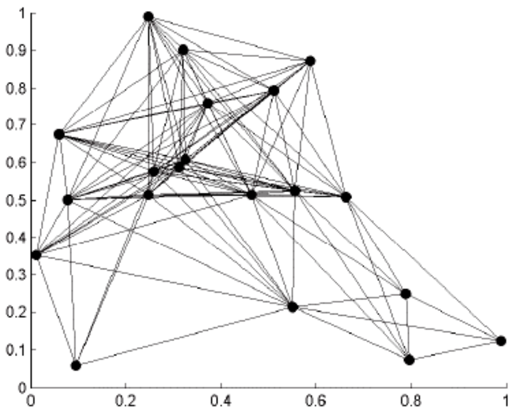
Coalition Formation at the Stable State

- The cost depends on the physical locations of nodes
  - Random network where nodes are placed according to a uniform Poisson point process on the [0,1] x [0,1] square.
- Theorem: The coalition formation at the stable state for n→∞
  - Given $\delta = 0$, $V = P\left(\frac{\ln n}{n}\right)^{\frac{\alpha}{2}}$ is a sharp threshold for establishing the grand coalition (number of coalitions = 1).
  - For $0 < \delta \le 1$, the threshold is less than $P\left(\frac{\ln n}{n}\right)^{\frac{\alpha}{2}}$.

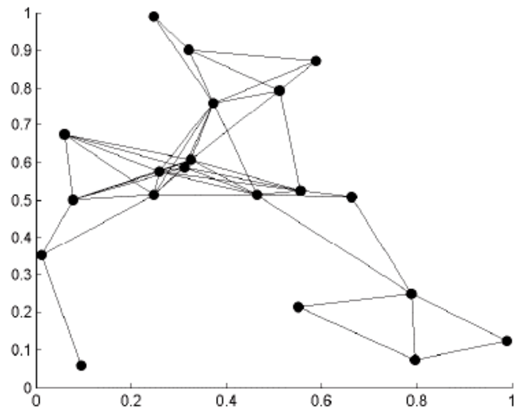


n = 20

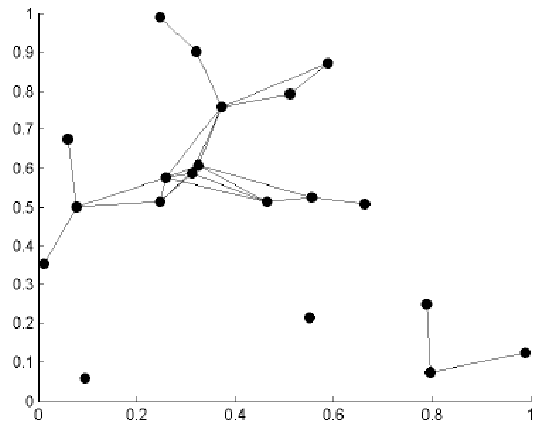
Topologies Formed



(a) P = 0.5 (low cost); complete graph



(b) P = 2 (middle cost); small world topology



(c) P = 4 (high cost); partitioned network

Outline

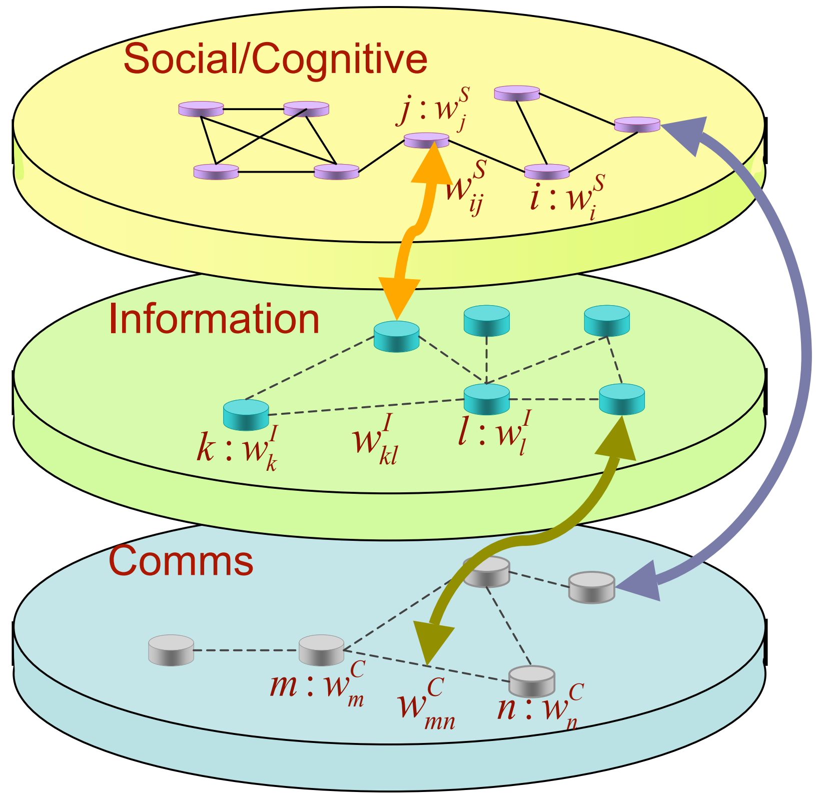
- Networks and Collaboration
  Constrained Coalitional Games
- Trust and Networks
- Security Aware Protocols via NUM
- Trust and Distributed Estimation
- Topology Matters
- Conclusions and Future Directions

Networks and Trust

- **Trust and reputation critical for collaboration**
- Characteristics of trust relations:
  - *Integrative* (Parsons1937) – main source of social order
  - *Reduction of complexity* – without it bureaucracy and transaction complexity increases (Luhmann 1988)
  - *Trust as a lubricant for cooperation* (Arrow 1974) – rational choice theory
- **Social Webs, Economic Webs**
  - MySpace, Facebook, Windows Live Spaces, Flickr, Classmates Online, Orkut, Yahoo! Groups, MSN Groups
  - e-commerce, e-XYZ, services and service composition
  - **Reputation** and **recommender** systems

Heterogeneous Dynamic Network Analysis and Trust

- Multiple Interacting Graphs
  - *Nodes*: agents, individuals, groups, organizations
  - Directed graphs
  - *Links*: ties, relationships
  - Weights on links : value (strength, significance) of tie
  - Weights on nodes : importance of node (agent)
- **Value directed graphs with weighted nodes**
- **Real-life problems: Dynamic, time varying graphs, relations, weights**

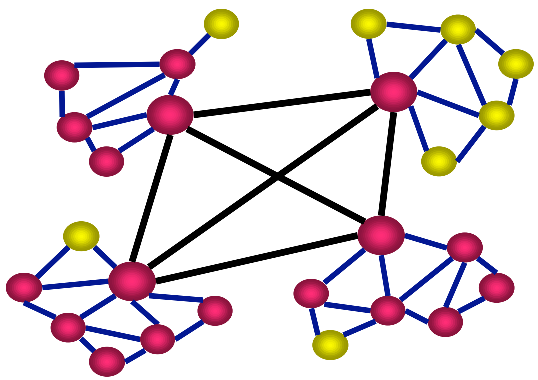


Organizational needs

Network architecture and operation

Next Generation Trust Analytics

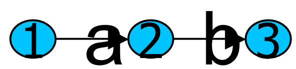
- Trust evaluation, trust and mistrust dynamics
  - Spin glasses (from **statistical physics**), phase transitions



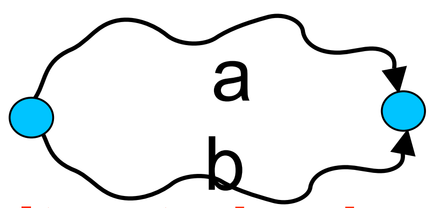
$$s_i(k+1) = f\left(\hat{J}_{ji}, s_j(k) \mid j \in N_i\right)$$

- **Indirect** trust; reputations, profiles; Trust computation via 'linear' **iterations in ordered semirings**

$a \otimes b \le a, b$



$a \oplus b \ge a, b$



2007 IEEE Leonard Abraham prize
New Book Draft

- **Direct trust: Iterated pairwise games on graphs** with players of many types

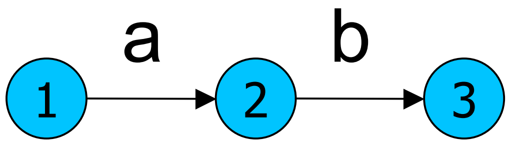
Semirings-Examples

- **Shortest Path Problem**
  - Semiring: $(\mathcal{R}_+, \min, +)$
  - $\otimes$ is + and computes **total path delay**
  - $\oplus$ is $\min$ and **picks shortest path**
- **Bottleneck Problem**
  - Semiring: $(\mathcal{R}_+, \max, \min)$
  - $\otimes$ is $\min$ and computes **path bandwidth**
  - $\oplus$ is $\max$ and **picks highest bandwidth**

Trust Semiring Properties: Partial Order

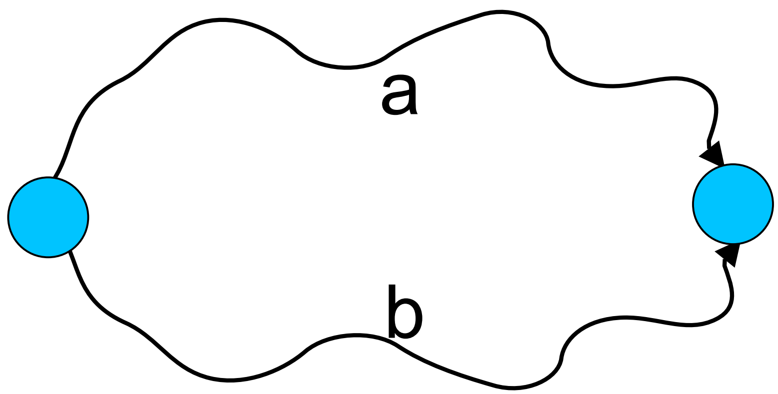
- Combined **along-a-path weight should not increase** :

$$a \otimes b \preceq a, b$$



- Combined **across-paths weight should not decrease** :

$$a \oplus b \succeq a, b$$



Computing Indirect Trust

- Path interpretation

$$t_{i\to j} = \bigoplus_{\text{path } p: i\to j} t^p_{i\to j}$$

- **Linear system interpretation**

$$t_{i\to j} = \bigoplus_{User\ k} t_{i\to k} \oplus w_{k\to j}$$

$$\vec{t}_n = W \otimes \vec{t}_{n-1} \oplus \vec{b}$$

Indicator vector of pre-trusted nodes

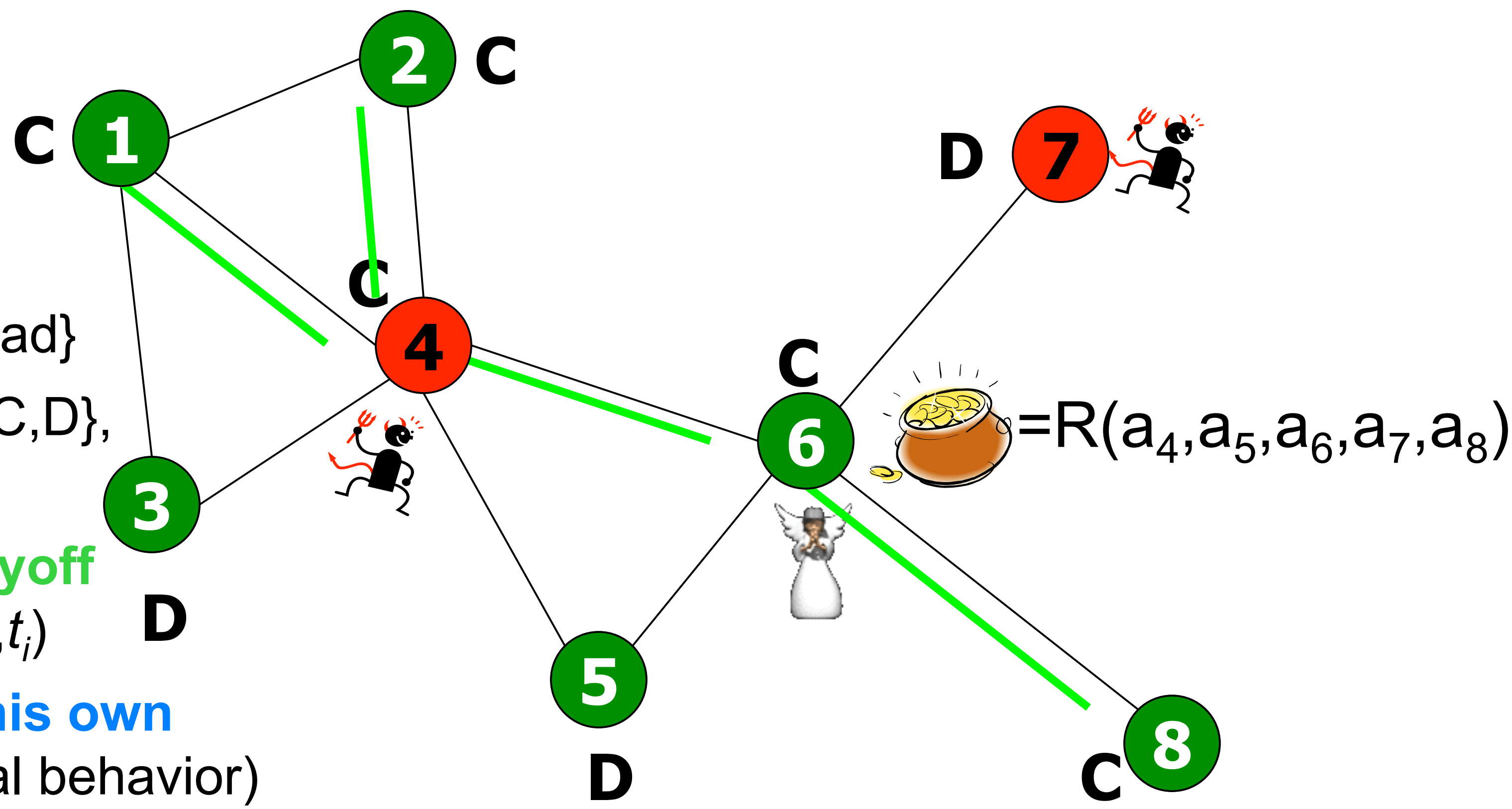
- Treat as a **linear system**
  – We are looking for its **steady state**.

Direct Trust

- **Direct trust** is based on **past** interactions between A, B.
- It is A's belief about B's **future** behavior.

User *i*

- of **type** $t_i \in$ {Good, Bad}
- **action** $a_i \in$ {C,D}, i=1,…N
- receives **payoff** $R_i = R(a_i, a_{\Gamma(i)}, t_i)$
- **maximize his own payoff** (local behavior)

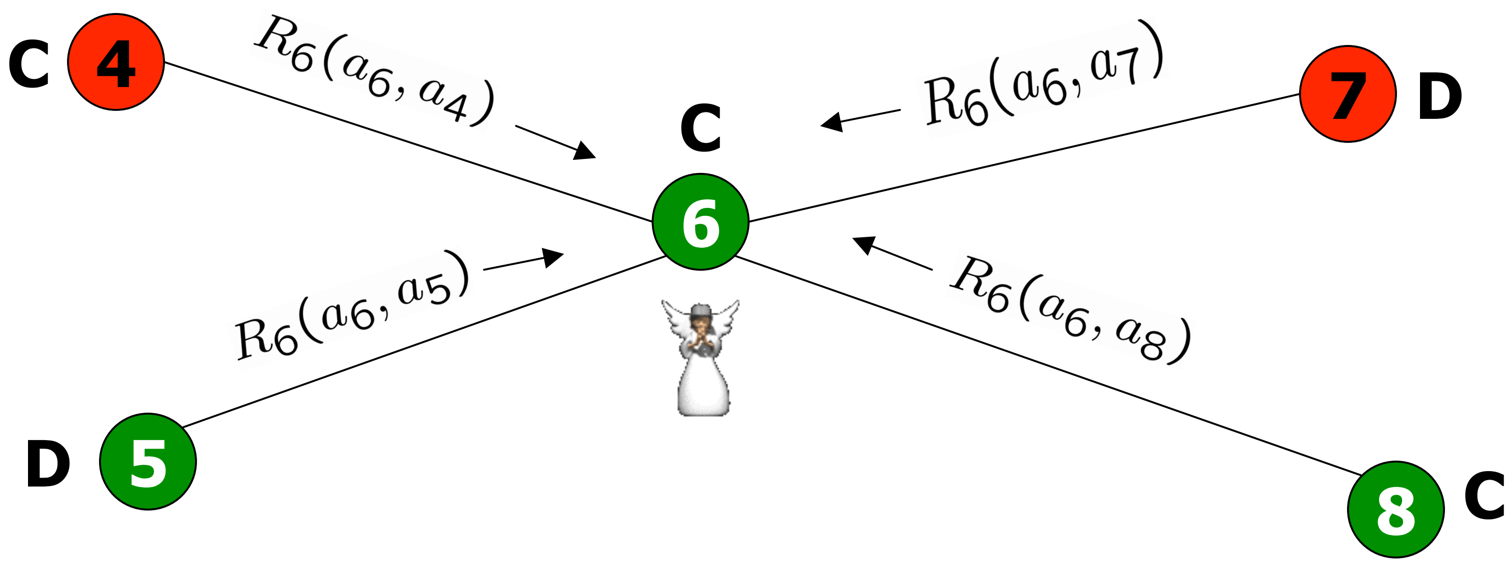


Only C-C links become active

Direct Trust: Games

- Payoff is decomposed as sum of **pairwise payoffs** along each link:

$$R_i(a_i, a_{\Gamma(i)}) = \sum_{j \in \Gamma(i)} R_i(a_i, a_j)$$



Direct Trust

- Problems we are studying:
  - **Repeated** interactions
  - **Take history into account (reputation, profiling)**

Strategy of User *i* for step *n*:

$$\sigma_i = \Pr\left[a_i = C | t_i, p_{\Gamma(i)}, \mathcal{H}^{1\ldots n-1}\right]$$

Probability (reputation) update for User i:

$$p_{\Gamma(i)}^{(n)} = f(p_{\Gamma(i)}^{(n-1)}, a_{\Gamma(i)}^{(n)})$$

Direct Trust

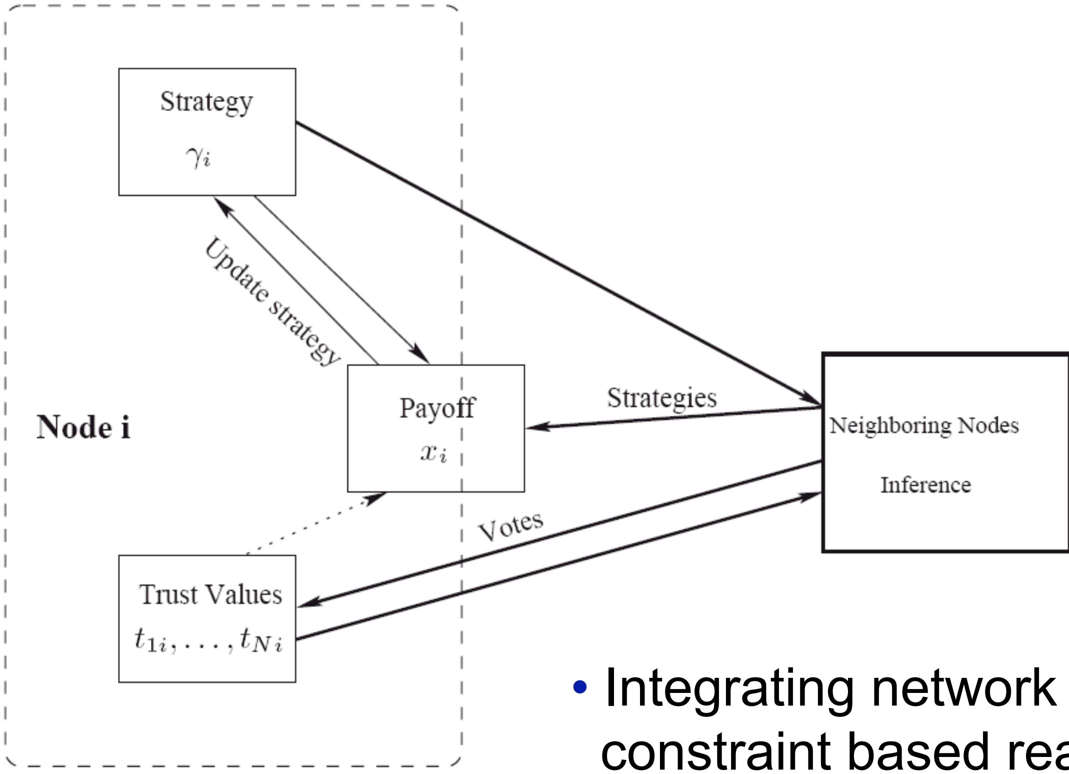
- **Two sequences evolving with time**:
  - **Vector of actions** (strategies), time 1:n

$$\mathcal{A}^{(1)} = \begin{pmatrix} a_1^{(1)} \\ \vdots \\ a_N^{(1)} \end{pmatrix}, \ \mathcal{A}^{(2)}, \ \ldots \ , \ \mathcal{A}^{(n)}$$

  - Set of **vectors of neighbor probabilities** (reputations), time 1:n

$$\mathcal{P}^{(1)} = \left\{ p_{\Gamma(1)}^{(1)}, \ldots, p_{\Gamma(N)}^{(1)} \right\}, \mathcal{P}^{(2)}, \ \ldots \ , \mathcal{P}^{(n)}$$

Constrained Coalitional Games: Trust and Collaboration



Two linked dynamics

- **Trust / Reputation propagation and Game evolution**

$$t+1) = f^i(x_i(t), \gamma_i(t), \gamma_j(t), t_{ij}(t))$$
$$t_{ik}(t) = g^i(t_{ij}(t), v_{jk}(t)) \quad \forall k \in N$$
$$x_i(t) = h^i(\gamma_i(t), \gamma_j(t))$$
$$v_{ij}(t) = p^i(\gamma_j(t), t_{ji}(t))$$

- Integrating network utility maximization (NUM) with constraint based reasoning and coalitional games

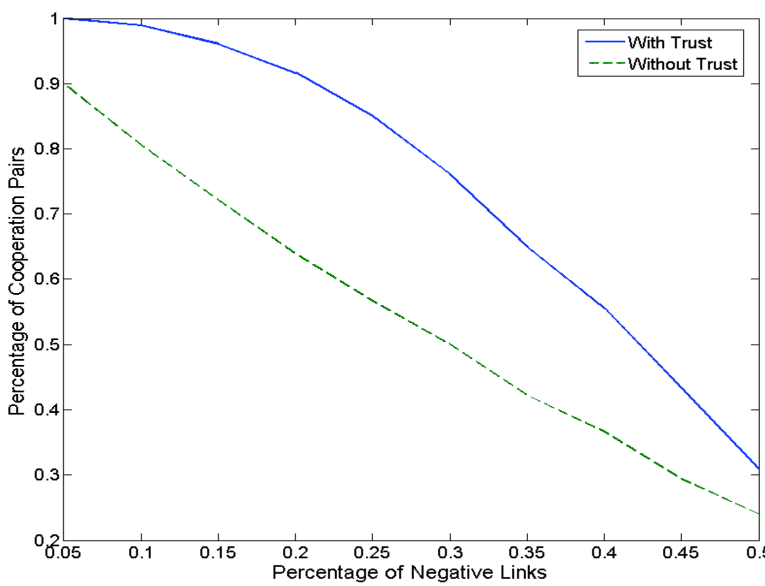
- Beyond linear algebra and weights**,** semirings of constraints, constraint programming, soft constraints semirings, policies, agents
- Learning on graphs and network dynamic games: behavior, adversaries
- Adversarial models, attacks, constrained shortest paths, …

Game Evolution

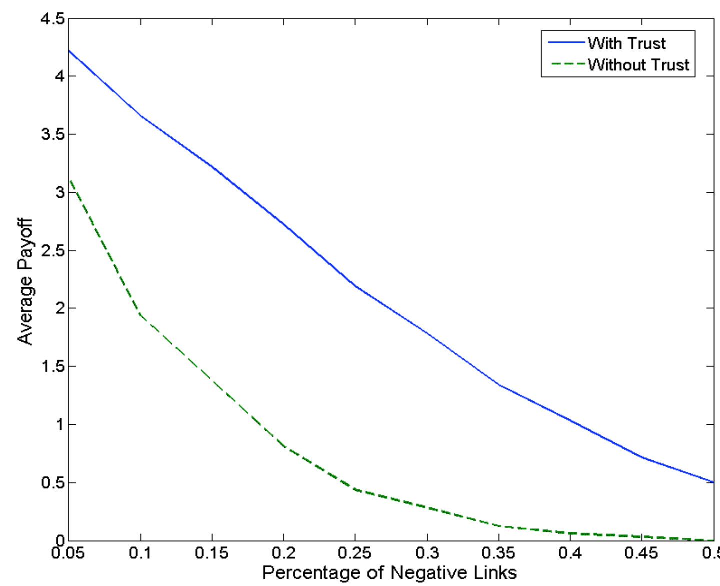
- Strategy of node *i*: $s_{ij} \in \{-1, 1\}, \forall j \in N_i$
  - s_{ij} = 1 (= -1) *i* **cooperates** (does not cooperate) with neighbor *j*
- Payoff for node *i* when interacting with *j* : $x_{ij} = J_{ij}\, s_{ij}\, s_{ji}$
  - $x_{ij} > 0$ (< 0) **positive** link (negative link)
  - **Node selfishness** → cooperate with neighbors on **positive** links
- Strategy updates: node *i* chooses $s_{ij}= 1$ only if all of the following are satisfied:
  - Neighbor *j* is trusted
  - $x_{ij} > 0$, or the cumulative payoff of *i* is less than the case when it unconditionally conducts $s_{ij}= 1$.
- Trust evaluation:
  - The deterministic voting rule
  - **Reestablishing period τ** : once a node is not trusted, in order to reestablish trust it has to cooperate for τ consecutive time steps

Results of Game Evolution

- **Theorem**: $\forall i \in N_i$ and $x_i = \sum_{j \in N_i} J_{ij}$, there exists τ_0, such that for a reestablishing period $\tau > \tau_0$
  - terated game converges to Nash equilibrium;
  - In the Nash equilibrium, all nodes cooperate with all their neighbors.
- Compare games with (without) trust mechanism, strategy update:



Percentage of cooperating pairs *vs* negative links



Average payoffs *vs* negative links

Outline

- Networks and Collaboration
  Constrained Coalitional Games
- Trust and Networks
- Security Aware Protocols via NUM
- Trust and Distributed Estimation
- Topology Matters
- Conclusions and Future Directions

Integrate Security into Network Utility Maximization Framework

- **NUM** : Optimization, utilities and duality for understanding protocol design and linkages
- Goal: extendNUM to MANET – time varying networks, uncertainties, non-convexities
- We use **'trust weights' in these optimizations** – whether they are joint MAC-routing or joint physical-MAC-routing optimizations
- These trust weights are developed by our **neighborhood-based collaborative monitoring and trust computation methods** and are disseminated via efficient methods for timely availability
- Effect of these trust weights on resulting protocols is that in the scheduling problems (MAC or routing) **trustworthy nodes will be automatically used**. Packets will not be routed as frequently to suspicious nodes. Or suspicious nodes will not be scheduled by the MAC protocol.
- Could be used to design **XYZ-metric aware** communication network protocols

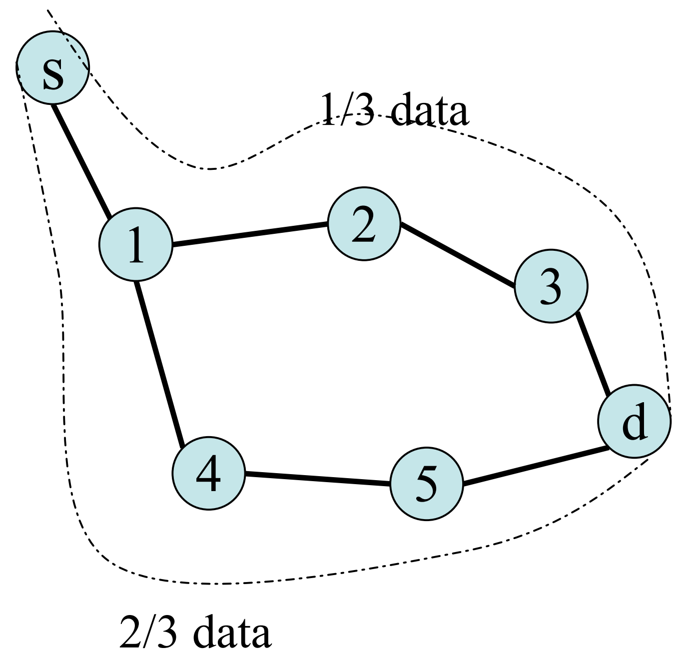
NUM without trust

- **Data flow**
  - F flows that share the network sources
  - Each flow f associated with a source node s_f and a destination node d_f
  - x_f is the rate with which data is sent from s_f to d_f over possibly multiple paths and multiple hops
- **Utility** function
  - Each flow is associated with a utility function $U_f(x_f)$
    - it reflects the “utility” to the flow f when its data rate is x_f
    - U_f is a strictly concave, non-decreasing , continuous differentiable
  - NUM is to maximize the utility function

$$\max_{x_f}\left(\sum_f U_f(x_f)\right)$$

Aggregate Trust Value

- Aggregate trust value of a flow (v_f)
  - Along paths
    - **multiplication** of node trust values along paths
  - Across paths
    - **Weighted summation** across all the paths the flow passes
    - **Weight** : the proportion of the flow passing the path



$$v_f = v_1\left(\frac{1}{3}v_2 v_3 + \frac{2}{3}v_4 v_5\right)$$

Trust - Aware NUM

- Trust aware NUM
  - $\max_{x_f}\left(\sum_f U x_f\right)$ ⟶ $\max_{x_f}\left(\sum_f U x_f g_f\right)$ $\quad (\hat{x}_f = g_f \, x_f)$

- **Dual decomposition (log change all variables)**

$$
\begin{aligned}
L(\lambda,\nu,\hat{\mathrm{x}},\mathrm{x},\mu,\mathrm{g}) = & \sum_f \max_{x'_f}\left\{\nu_f x'_f - \lambda^f_{s_f} e^{x'_f}\right\} + \sum_f \max_{\hat{x}'_f}\left\{U'_f(\hat{x}'_f) - \nu_f \hat{x}'_f\right\} \\
& + \max_{\mathrm{g}'} \sum_f \nu_f g'_f \\
& + \max_{\mu\in\Gamma} \sum_{(i,j)\in\mathcal{L}} \sum_{f\in\mathcal{F}} \mu^f_{ij}(\lambda^f_i - \lambda^f_j).
\end{aligned}
$$

Flow rates

Routes

Schedule (MAC)

  - **Dual objective function** $\quad h(\lambda,\nu) = \sup_{\substack{\mathrm{x}\in\Lambda \\ \hat{x}_f = g_f\cdot x_f}} L(\lambda,\nu,\hat{\mathrm{x}},\mathrm{x},\mu,\mathrm{g})$

The Institute for Systems Research

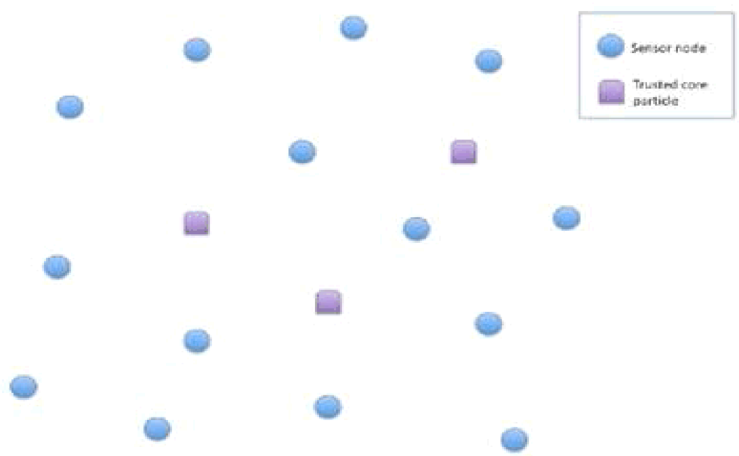
Outline

- Networks and Collaboration
  Constrained Coalitional Games
- Trust and Networks
- Security Aware Protocols via NUM
- Trust and Distributed Estimation
- Topology Matters
- Conclusions and Future Directions

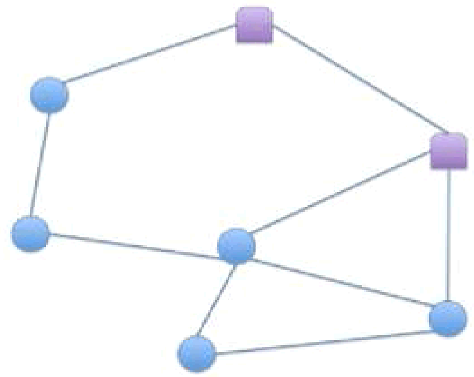
Distributed Kalman Filtering and Tracking

- **Realistic sensor networks**: Normal nodes, faulty or corrupted nodes, malicious nodes
- **Hierarchical scheme** – provide global trust on a particular context without requiring direct trust on the same context between all agents
- Combine techniques from fusion centric, collaborative filtering, estimation propagation
- **Trusted Core**
  - Trust Particles, higher security, additional sensing capabilities, broader observation of the system, confidentiality and integrity, multipath comms
  - Every sensor can communicate with one or more trust particles at a cost

Trust and Hierarchy



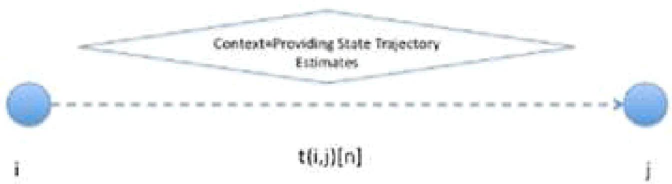
Sensor Network



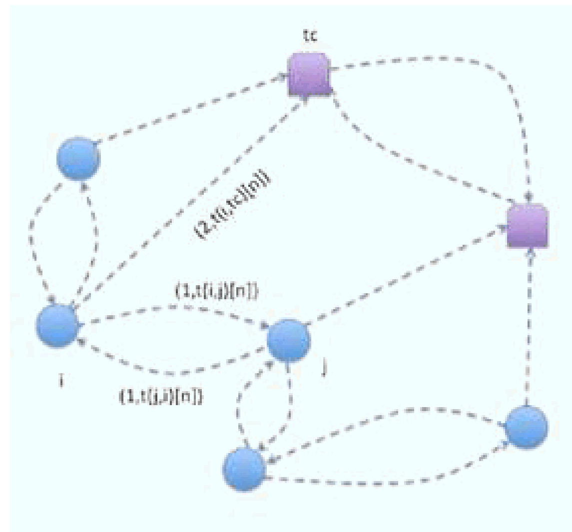
Communication Graph from disc model $G_c(V,E_c)$

- Distributed Kalman Filter Particles: Sensor nodes exchange estimates in their local neighborhood and trusted measurements from the trusted core
- Hierarchical scheme – provide global trust on a particular context without requiring direct trust on the same context between all agents

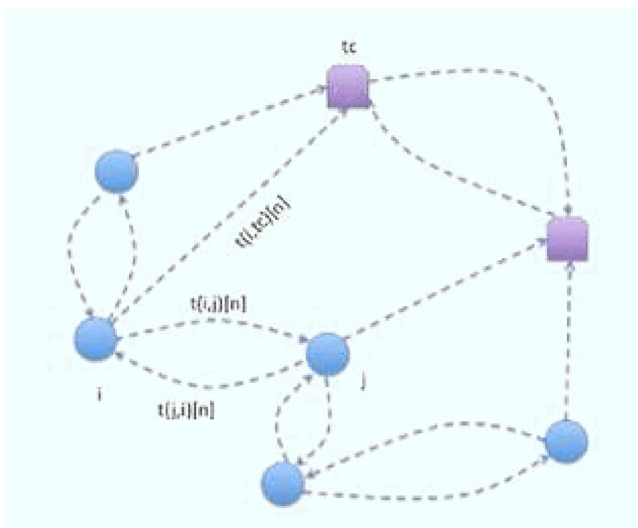
Trust and Induced Graphs



Trust relation



Induced Graph $G(V, A)$



Weighted Directed Dynamic Trust Graph $G_t\ (V, A_t)$

$$V_{tc} \subset V$$

$$w(i,j) = (c(i), t(i,j)[n])$$

Goals of Trusted System

1. All the sensors which abide by the protocols of sensing and message passing, should be able to track the trajectories.
2. This implies that those nodes which have poor sensing capabilities, *nodes with corrupted sensors, should* be aided by their neighbors in tracking.
3. Those nodes which are *malicious and pass false estimates,* should be quickly detected by the trust mechanism and their estimates should be discarded.

$$x[n+1] = A x[n] + B w[n]$$

$$z_i[n] = H_i x[n] + v_i[n]$$

$$z_{tc}[n] = H_{tc} x[n] + v_{tc}[n]$$

Trusted DKF and Particles

- Can use **any valid trust system** as trust update component
- Can replace DKF with **any Distributed Sequential MMSE or other filter**
- Trust update mechanism: Linear credit and exponential penalty

Algorithm 1 Trusted Kalman Filter

```
Init M[0], x̂_i = x(0), n = 0
repeat
   n ← n + 1;
   Prediction MSE
   P[n] = A M[n-1] A^T + B Q B^T
   Kalman Gain
   K[n] = P[n] H_i^T (R_i + H_i P[n] H_i^T)^-1
   Local correction
   ζ_i[n] = A x̂_i[n-1] + K[n](z_i[n] - H_i A x̂_i[n-1])
   The nodes exchanges the local estimates x̂_j, ∀j ∈ N+(i)
   Trust sensitive filtering
   x̂_i[n] = Σ_{j in N+(i)} w_ij × ζ_j[n]
   Estimation MSE
   M[n] = (I - K[n] H_i[n]) P[n]
until Forever
```

Init $M[0], \underline{\hat{x}}_i = \underline{x}(0), n = 0$

repeat

$n \leftarrow n + 1;$

Prediction MSE

$\mathbf{P}[n] = \mathbf{A}\mathbf{M}[n-1]\mathbf{A}^T + \mathbf{B}\mathbf{Q}\mathbf{B}^T$

Kalman Gain

$\mathbf{K}[n] = \mathbf{P}[n]\mathbf{H}_i{}^T(\mathbf{R}_i + \mathbf{H}_i\mathbf{P}[n]\mathbf{H}_i{}^T)^{-1}$

Local correction

$\underline{\zeta}_i[n] = \mathbf{A}\underline{\hat{x}}_i[n-1] + \mathbf{K}[n](\underline{z}_i[n] - \mathbf{H}_i\mathbf{A}\underline{\hat{x}}_i[n-1])$

The nodes exchanges the local estimates $\underline{\hat{x}}_j, \forall j \in \mathcal{N}^+(i)$

Trust sensitive filtering

$\underline{\hat{x}}_i[n] = \sum_{j in \mathcal{N}^+(i)} w_{ij} \times \underline{\zeta}_j[n]$

Estimation MSE

$\mathbf{M}[n] = (\mathbf{I} - \mathbf{K}[n]\mathbf{H}_i[n])\mathbf{P}[n]$

until Forever

Algorithm 2 Trust Update for the inclusive neighborhood

Init $t(i,j)[0] = \frac{1}{|\mathcal{N}^+(i)|}, \quad \forall j \in \mathcal{N}^+(i)$, and $k = 0$

repeat

Wait for Exponential time τ

$k \leftarrow k + \tau$

Request Estimate update from the TC

The TC replies with its trustworthy estimate $\underline{\hat{x}}_{tc}$

for all $j \in \mathcal{N}^+(i)$ **do**

$dev(j) = ||\underline{\zeta}_j - \underline{\zeta}_{tc}||_2$

$$t(i,j)[k] = \begin{cases} \min(max_T, t(i,j)[k-1]) + \delta & dev(j) \le Dev_T \\ t(i,j)[k-1]/2 & dev(j) > Dev_T \end{cases}$$

end for

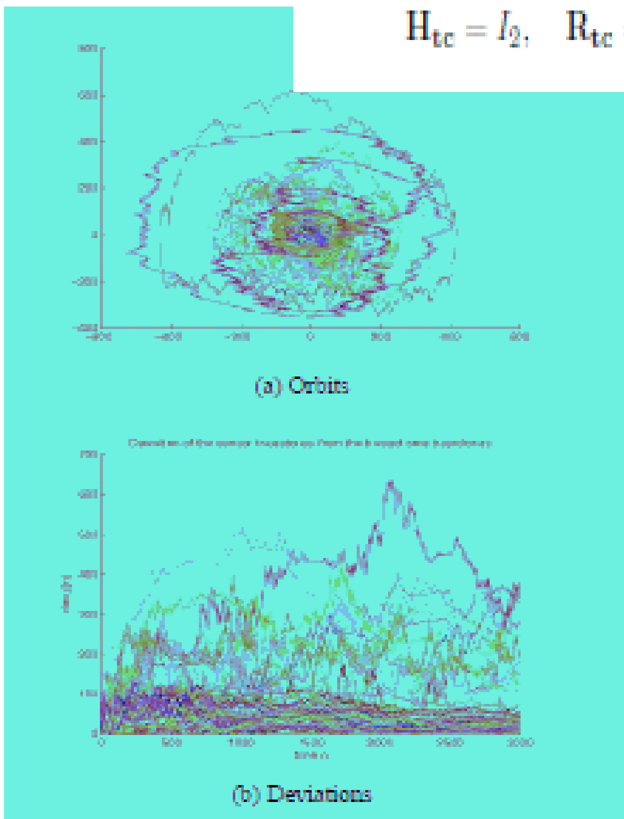
until Forever

Trusted DKF Performance

$$A = 2\begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix}$$

$$Q = 25I_2, \quad \underline{x}(0) = (15, -10)^T$$

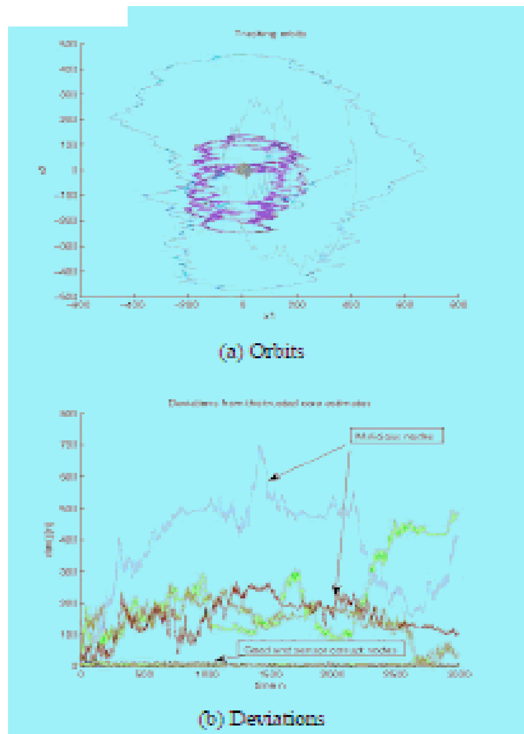
$$H_{tc} = I_2, \quad R_{tc} = 30I_2$$



(a) Orbits

(b) Deviations

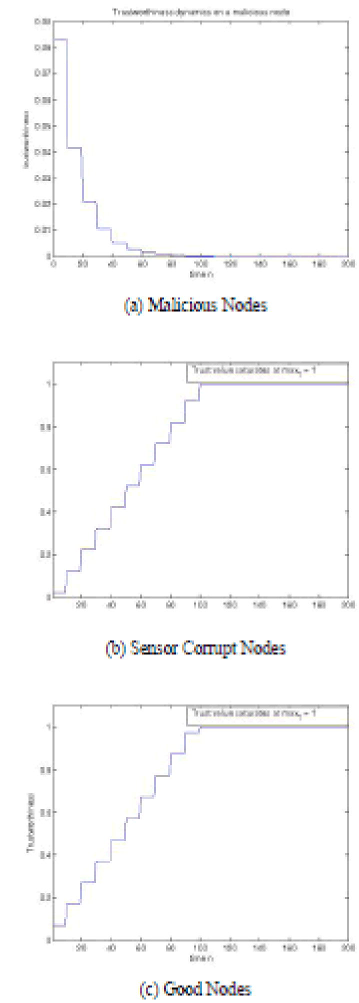
Open Loop Performance



(a) Orbits

(b) Deviations

Closed Loop Performance



(a) Malicious Nodes

(b) Sensor Corrupt Nodes

(c) Good Nodes

Trust System Performance

Outline

- Networks and Collaboration
  Constrained Coalitional Games
- Trust and Networks
- Security Aware Protocols via NUM
- Trust and Distributed Estimation
- Topology Matters
- Conclusions and Future Directions

Distributed Algorithms in Networked Systems and Topologies

- Distributed algorithms are essential
  - Group of agents with certain abilities
  - Agents **communicate with neighbors,** share/process information
  - Agents **perform local** actions
  - **Emergence** of global behaviors
- **Effectiveness** of distributed algorithms
  - The **speed** of convergence
  - **Robustness** to agent/connection failures
  - Energy/ communication **efficiency**
- **Group topology affects** group performance
- **Design problem**:

  Find graph topologies with favorable tradeoff between performance improvement (**benefit**) *vs* **cost** of collaboration
- **Example: Small Word graphs** in consensus problems

Consensus Problems: Design of Information Flow

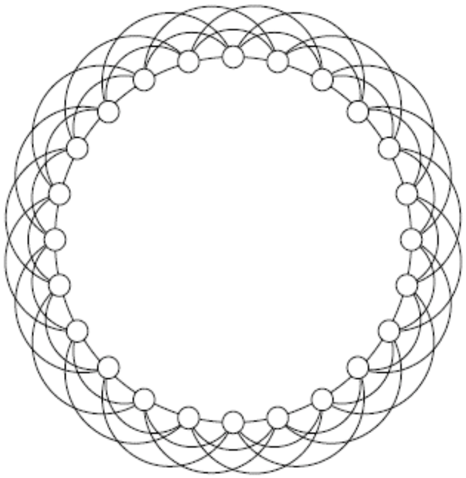
$$x(k+1) = F(k)x(k)$$

$$\boxed{\begin{gathered} F(k) = (I + D(k))^{-1}(A(k) + I) \\ F(k) = I - hL(k) \end{gathered}}$$

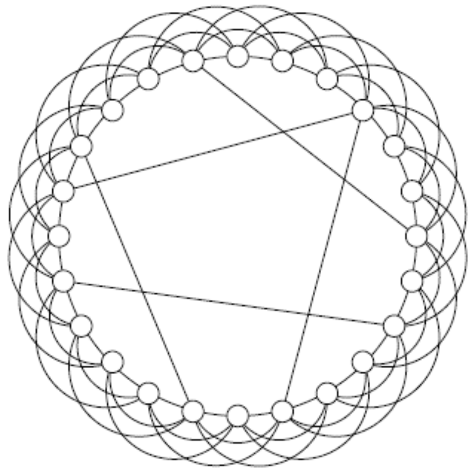
Symmetric communication

- **Fixed graphs**: Geometric convergence with rate equal to Second Largest Eigenvalue Modulus (SLEM)
- How does **graph topology affect** location of eigenvalues?
- How can we **design graph topologies** which result in good convergence speed?

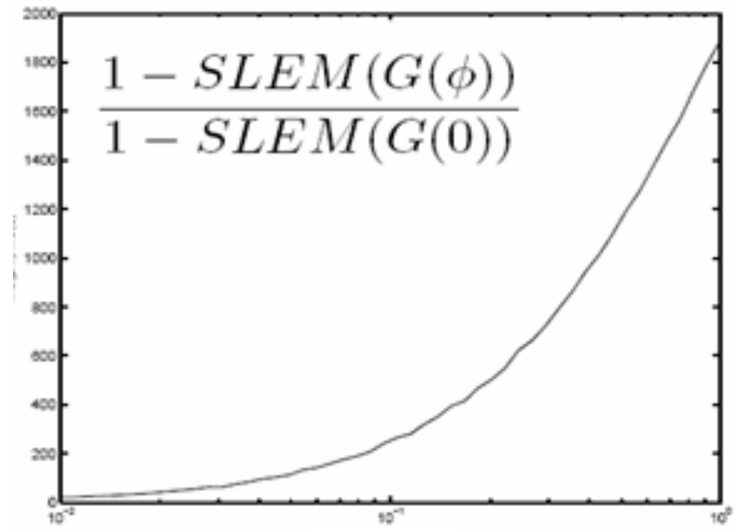
Small World Graphs



Simple Lattice
C(n,k)



Small world: Slight variation adding $nk\Phi$

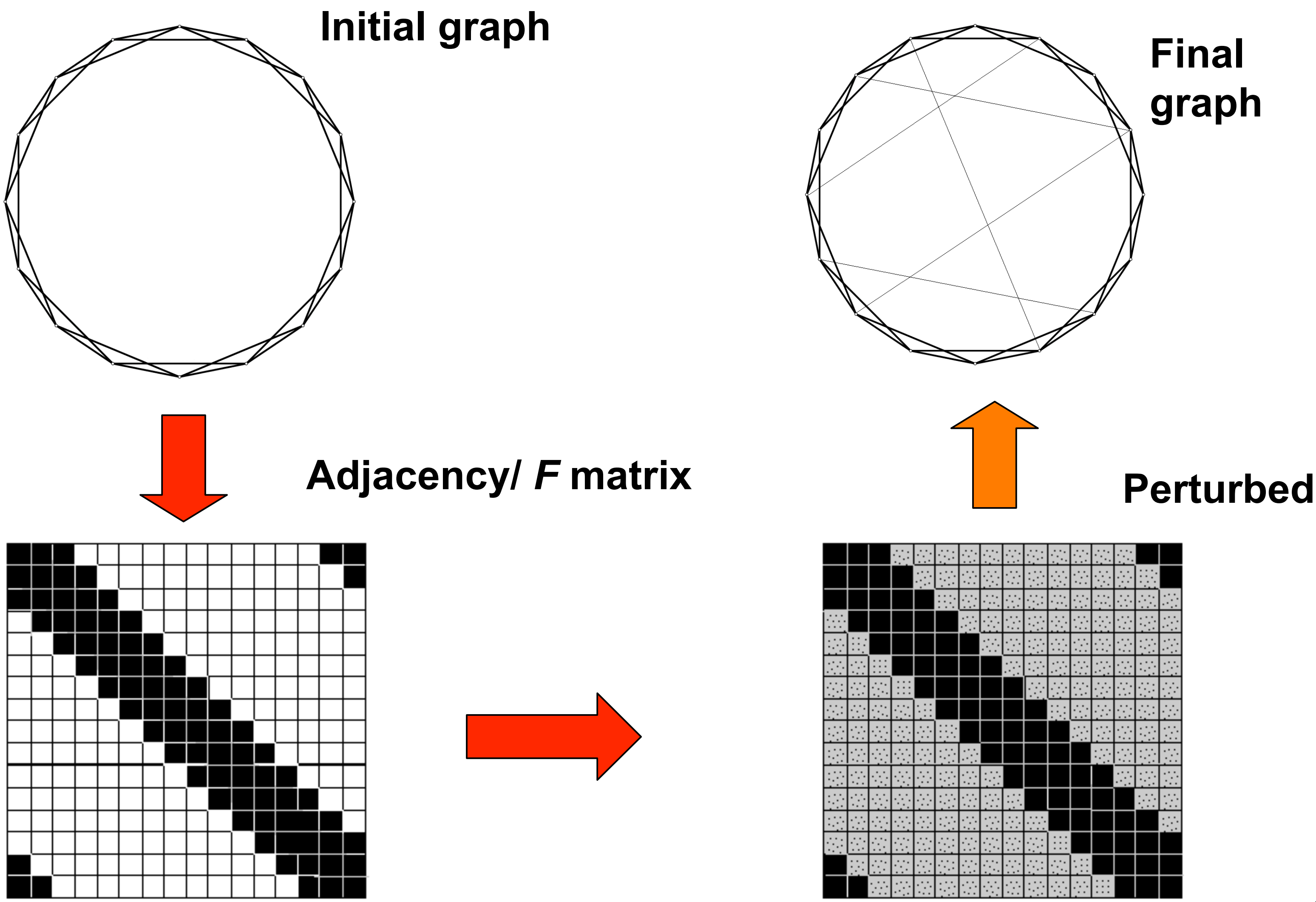


\phi

C(1000,5)

Adding a **small portion** of well-chosen links → **significant increase** in convergence rate

Mean Field Explanation and Perturbation Approach



Watts-Strogatz Small World networks

- Random graph approach
  (e.g. Durrett 2007, Tahbaz and Jadbabaie 2007)
- **Perturbation** approach (Higham 2003)
  - Start from lattice structure $G_0 = C(n,k)$ $\longleftrightarrow$ F_0
  - Perturb zero elements in the positive direction by $\varepsilon = \frac{K}{n^{\alpha}}$ for fixed $K > 0$ and $\alpha > 1$.
  - Perturb the formerly nonzero elements equally, such that the stochastic structure of the F matrix is preserved F_ε
  - Analyze the SLEM as a function of the perturbation as α varies

Distributed exploration of the graph structure

- **Self-organization for better performance and resiliency**
- Hierarchical scheme to design a network structure capable of running **distributed algorithms with high convergence speed**
- A two stage algorithm:

  1- Find the most effective choice of **local leaders**

  2- Provide nodes with information about their location **with respect to other nodes and leaders** and the choice of groups to form

- Divide N agents into K groups with M members each

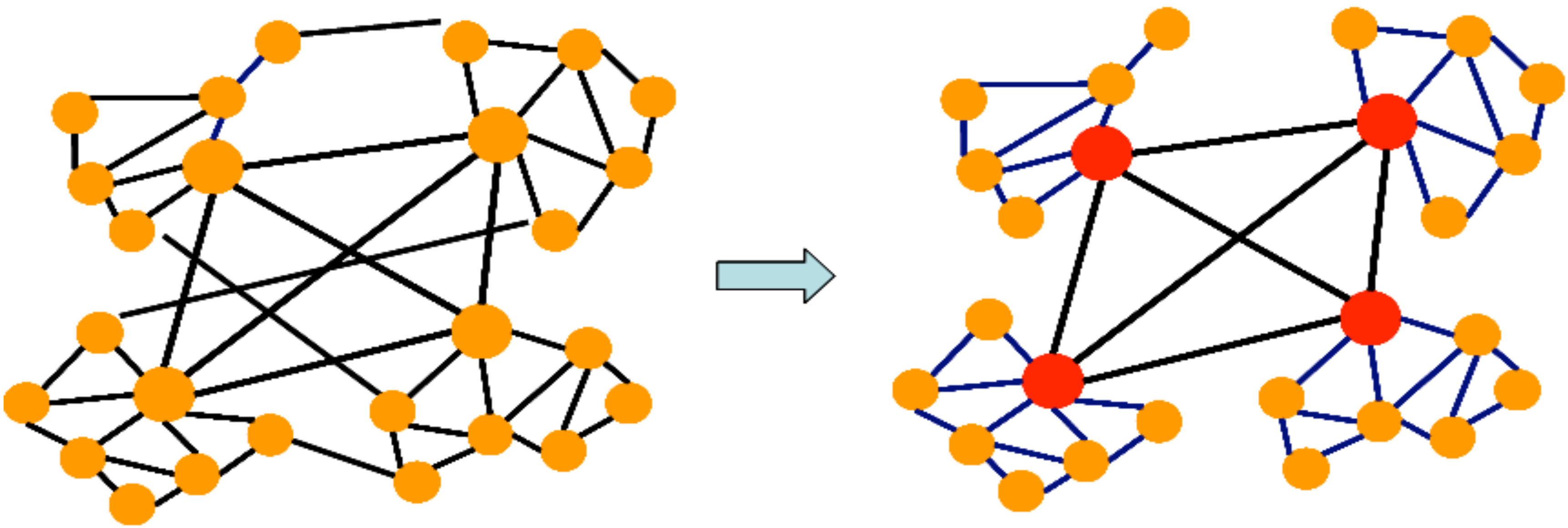
  $NK{=}MKM{\times}N$, $\leq$ □ , select 'leaders'

Distributed self - organization

Determine leaders → Find influence → Remove unnecessary links

Semi-decentralized

Decentralized: Dirichlet problem on the graph



Goal: design a scheme that gives each node a vector of compact global information

Two stage semi-decentralized algorithm

- **Stage 1**: **Determining *K* leaders**
  - Each node determines its social degree via local query
  - Dominant nodes in each neighborhood send their degrees to the central authority
  - Central authority computes their social scores

$$SC(k) = \alpha SD^{(2)}(k) + (1-\alpha) SD^{(3)}(k)$$

    Choice of α determines whether leaders in star-like neighborhoods are preferred

  - The central authority selects the *K* nodes with highest scores as social leaders and gives them an arbitrary order

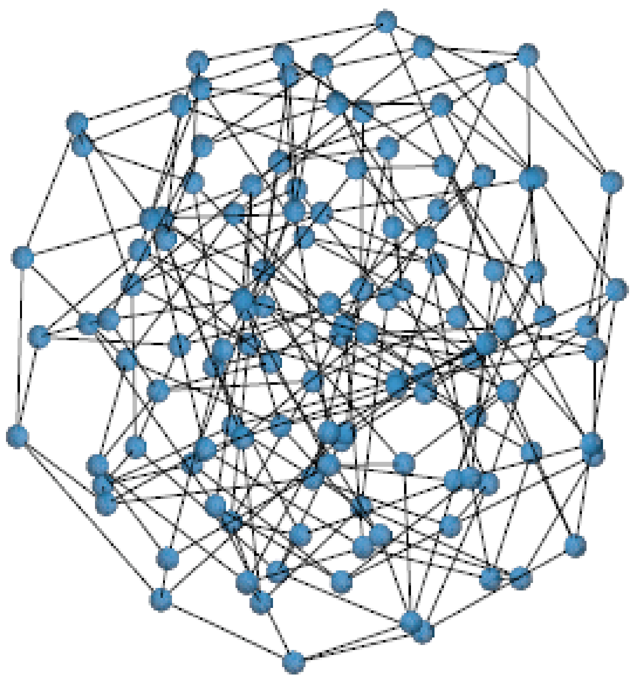
Expander Graphs

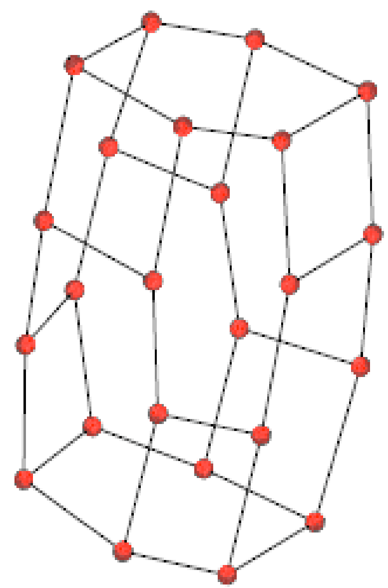
- Fast synchronization of a network of oscillators
- Network where any node is "nearby" any other
- Fast 'diffusion' of information in a network
- Fast convergence of consensus
- Decide connectivity with smallest memory
- Random walks converge rapidly …
- Graph G, ***Cheeger constant h(G)***
  - All partitions of G to S and S^c ,

    $h(G)$=min (#edges connecting S and S^c) /

    (#nodes in smallest of S and S^c)
- (k , N, ε) **expander** : $h(G) > \varepsilon$; **sparse but well connected**

Expander Graphs – Ramanujan Graphs

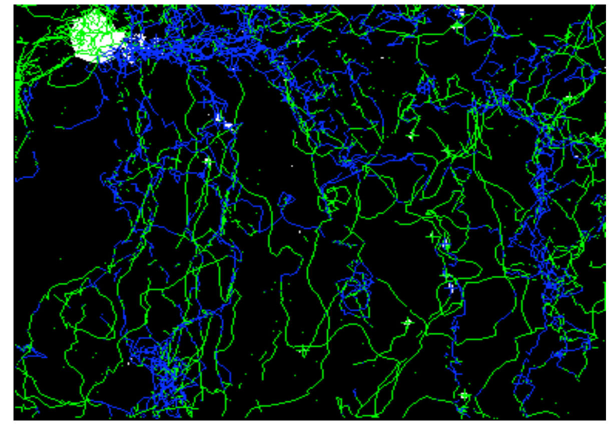


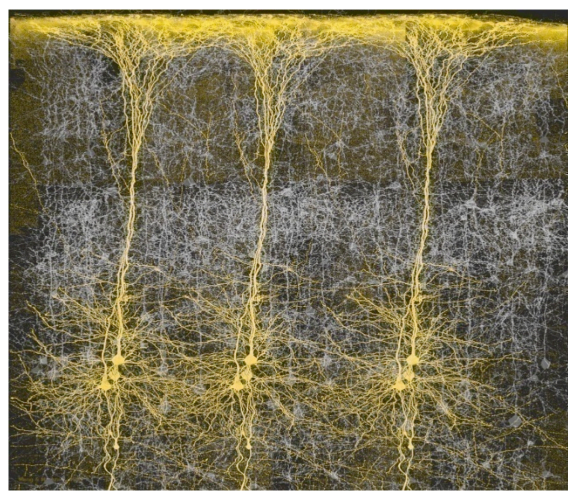


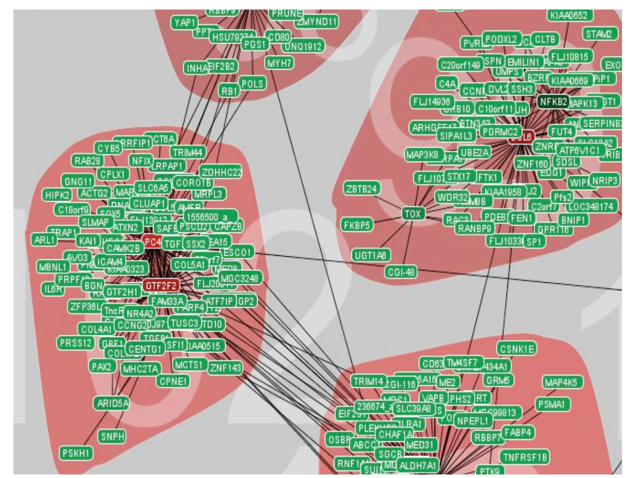
Outline

- Networks and Collaboration
  Constrained Coalitional Games
- Trust and Networks
- Security Aware Protocols via NUM
- Component-based Networking
- Topology Matters
- Conclusions and Future Directions

How Biology Does IT?







Control vs Communication

- Many graphs as **abstractions**
- **Collaboration graph** – or a model of what the system does (**behavior**)
- **Communication graph** – or a model of what the system consist of (**structure**)
- Nodes with **attributes** – several graphs
- **Key question 1**: Given behavior, what structure (subject to constraints) gives best performance?
- **Key question 2**: Given structure (and constraints) how well behavior can be executed?

Lessons Learned -- Future Directions

- **Constrained coalitional games** – unifying concept
- Generalized networks, **flows - potentials**, duality and network optimization (monotropic optimization)
- **Time varying graphs** – mixing – statistical physics
- **Understand autonomy** – better to have self-organized topology capable of supporting (scalable, fast) a rich set of distributed algorithms (small world graphs, expander graphs) than optimized topology
- Given a set of distributed computations **is there a small set of simple rules** that when given to the nodes they can self-generate such topologies?

Thank you!

baras@isr.umd.edu

301-405-6606

http://www.isr.umd.edu/~baras

Questions?