Collaboration in Networks of Autonomous Agents: Coalitional Games, Constraints and Architectures

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

Joint work with: Tao Jiang, Pedram Hovareshti

November 2, 2008
Frontiers in Distributed Communication, Sensing and Control
Yale University, New Haven
Outline

- Networks and Collaboration
- Constrained Coalitional Games
- Security Aware Protocols via NUM
- Topology Matters
- Conclusions and Future Directions
What is a Network …?

• In several fields or contexts:
  - social
  - economic
  - communication
  - sensor
  - biological
  - physics and materials
A Network is ...

- A collection of nodes, agents, … that **collaborate** to accomplish actions, gains, … that cannot be accomplished without such collaboration.

- Most significant concept for **autonomic networks**.
The Fundamental Trade-off

- The nodes **gain** from collaborating
- To collaborate they need to **communicate**, and this represents **cost**
- **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 communications
- **Example 3:** Web-based social networks and services
Example: Social Networks and Trust

• Trust and reputation critical for collaboration

• Characteristics of trust relations:
  – *Integrative* (Parsons 1937) – 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
Example: Autonomic Networks

• Autonomic: self-organized, distributed, unattended
  – Sensor networks
  – Mobile ad hoc networks

• Autonomic networks depend on collaboration between their nodes for all their functions
  – The nodes gain from collaboration: e.g. multihop routing
  – Collaboration introduces cost: e.g. energy consumption for packet forwarding
Example: Social Webs

• In August 2007, there were totally 330,000,000 unique visits to social web sites. (Source: Nielsen Online)
  – 9 sites with over 10,000,000 unique visits
  – MySpace, Facebook, Windows Live Spaces, Flickr, Classmates Online, Orkut, Yahoo! Groups, MSN Groups

• Main types of social networking services
  – e-commerce, e-XYZ
  – means to connect with friends: usually with self-description pages
  – recommender systems linked to trust/reputation
Social Network Models and Analysis

• 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* (Bunskens 2002)

• *Real-life problems*: Dynamic, time varying graphs and relations, weights
Outline

- Networks and Collaboration
- Constrained Coalitional Games
- Security Aware Protocols via NUM
- Topology Matters
- Conclusions and Future Directions
Coalitions, Networks and Constraints

- **Cooperative Game** in characteristic function form $\Gamma = \{\mathcal{N}, v\}$, $\mathcal{N} = \{1, 2, \ldots, N\}$, $v : 2^N \rightarrow \mathbf{R}$, on all subsets $\mathcal{S}$ (coalitions) of $\mathcal{N}$

- All coalitions cannot be formed
- To collaborate agents need to communicate
- Communication Network $(N, L)$
  - Edges – links between payers
  - $i$ and $j$ directly connected
  - $i$ and $j$ path connected

- Cooperation components
- Links between players in $\mathcal{S}$, $L(\mathcal{S})$
- Network $(\mathcal{S}, L(\mathcal{S}))$ induces a partition of $\mathcal{S}$
Constrained Coalitional Games

- Network-restricted cooperation game or constrained coalition \( \{\mathcal{N}, v^L\} \)
- \( \{\mathcal{N}, v, L\} \) communication situation

- Characteristic function

\[
v^L(S) = \sum_{C \in S / L} v(C) \quad \text{for each} \quad S \subseteq \mathcal{N}
\]

- Myerson value: Shapley value of \( \{\mathcal{N}, v^L\} \)
Network Formation

• Form links pairwise
• Iterative game
• Better understanding of topologies – dynamics – topology control
• **Network formation with costs** for establishing links
  
  \[
  \mathcal{N}, v, L, c \quad \{\mathcal{N}, v^{L,c}\}
  \]

  
  \[
  v^{L,c}(S) = \sum_{C \in S/L} v(C) - c |L(S)| \quad \text{for each} \quad S \subseteq \mathcal{N}
  \]

  
  • **Stability** vs **efficiency** of the resulting network
  • Small world graphs, expander graphs …
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 $i$

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

- $r_{ij}$ is the number of hops in the shortest path between $i$ and $j$.
  - $r_{ij} = \infty$ if there is no path between $i$ and $j$.
- $0 \leq \delta \leq 1$ is the communication depreciation rate.
• 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

  • $\alpha$ is path loss exponent -- depends on specific propagation environment.
Pairwise Game and Convergence

• Payoff of node $i$ from the network $G$ is defined as

$$v_i(G) = \text{gain} - \text{cost} = w_i(G) - c_i(G)$$

• 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)
Pairwise Game

- The pairwise game is modeled as an *iterated process*. Individual nodes activate and delete links based on the improvement that the resulting network offers them relative to the current network.

- A strategy of node $i$ is a vector defined as
  \[ \gamma_i = (\gamma_{i,1}, \ldots, \gamma_{i,i-1}, \gamma_{i,i+1}, \ldots, \gamma_{i,n}). \]
  - $\gamma_{i,j} = 1$ (or 0): node $i$ wants (or does not want) to form a link with node $j$
  - A link $ij$ is formed only if $\gamma_{i,j} = 1$ and $\gamma_{j,i} = 1$.

- A strategy profile $\gamma^{(t)} = (\gamma_{1}^{(t)}, \ldots, \gamma_{n}^{(t)})$ at time period $t$ corresponds to the network $G^{(t)}$ at time $t$.

\[
\begin{align*}
\gamma_1 &= \{1, 1\} \\
\gamma_2 &= \{0, 1\} \\
\gamma_3 &= \{1, 1\}
\end{align*}
\]
Convergence of the Iterated Pairwise Game

- **Pairwise stability**
  - No more link is added and no existing link is deleted

- **Lemma**: the iterated pairwise game converges to a pairwise stable network or a cycle of networks.
  - The converging pairwise stable network may be inefficient

- Random mutations are introduced, the game converges to a unique Pareto equilibrium (Markov chain states strategy profiles $\gamma$)
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 $[0,1] \times [0,1]$ square.
- **Theorem**: The coalition formation at the stable state for $n \rightarrow \infty$
  - Given $\delta = 0$, $V = P\left(\frac{\ln n}{n}\right)^{\alpha/2}$ is a sharp threshold for establishing the grand coalition (number of coalitions = 1).
  - For $0 < \delta \leq 1$, the threshold is less than $P\left(\frac{\ln n}{n}\right)^{\alpha/2}$.

\[ n = 20 \]
(a) $P = 0.5$ (low cost); complete graph
(b) $P = 2$ (middle cost); small world topology
(c) $P = 4$ (high cost); partitioned network
Stability of Coalitions

- **Core stability**
  - A network $G$ is **core stable** if there is no subset of nodes $S$ who prefer another network $\hat{G}$ to $G$ and who can change the network from $G$ to $\hat{G}$ without the cooperation from the rest of the set of nodes $N \setminus S$.

\[
\chi_{i}(\hat{G}) \geq \chi_{i}(G) \text{ for all } i \in S \text{ and there is at least one strict inequality}
\]

If $ij \in \hat{G}$ but $ij \notin G$, then $i, j \in S$

If $ij \notin \hat{G}$ but $ij \in G$, then $i \in S$ and/or $j \in S$

- Core stability allows that a node is able to interact and coordinate with any other nodes in the same coalition.

- Core stability is **stronger** than pairwise stability.
Outline

- Networks and Collaboration
- Constrained Coalitional Games
- Security Aware Protocols via NUM
- 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: extend NUM 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
Trust Aware Cross - Layer Optimization

- Resource allocation in wireless networks
  - Wireless spectrum is scarce
  - **Network utility maximization (NUM)** driven network design
    - In wired networks
      - Linear flow constraints
    - In wireless networks
      - Nonlinear flow constraints coupled with transmission power and channel parameters

- Multihop wireless networks
  - Distributed control
    - **Distributed scheduling algorithm**
  - Security
    - **Trust** of nodes on transmission routes
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

\[
\max_x \sum_{f} U_f(x_f)
\]
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

\[
g_f = v_1 \left( \frac{1}{3} v_2 v_3 + \frac{2}{3} v_4 v_5 \right)
\]
Trust - Aware NUM

- Trust aware NUM
  \[ \max_x \sum_f U_f(x_f) \rightarrow \max_x \sum_f U_f(g_f x_f) \quad (\hat{x}_f = g_f x_f) \]

- Dual decomposition (log change all variables)

\[
L(\lambda, \nu, \hat{x}, x, \mu, g) = \sum_f \max_{x'_f} \left\{ \nu_f x'_f - \lambda^f_{sf} e^{x'_f} \right\} + \sum_f \max_{\hat{x}'_f} \left\{ U'_f(\hat{x}'_f) - \nu_f \hat{x}'_f \right\} \\
+ \max_{g'} \sum_f \nu_f g'_f \\
+ \max_{\mu \in \Gamma} \sum_{(i,j) \in L} \sum_{f \in \mathcal{F}} \mu^f_{ij} (\lambda^f_i - \lambda^f_j).
\]

- Dual objective function

\[
h(\lambda, \nu) = \sup_{\hat{x} \in \Lambda} L(\lambda, \nu, \hat{x}, x, \mu, g) \quad \text{subject to} \quad \hat{x}_f = g_f x_f
\]
Outline

- Networks and Collaboration
- Constrained Coalitional Games
- Security Aware Protocols via NUM
- Topology Matters
- Conclusions and Future Directions
Effects of Graph Topology on Convergence of Network Algorithms

- Distributed algorithms frequently arise in networked systems
  - Group of agents with simple/complex abilities
  - Agents sense their “local” neighborhood
  - Communicate with neighbors and process the information
  - Perform a local action
  - Emergence of a global behavior.
    Example: iterative actions leading to convergence to an agreement about “coordination variables” in consensus problems
  - Group topology affects group performance
Effects of Graph Topology (cont.)

- **Effectiveness** of these algorithms depends on:
  - The speed of convergence
  - Robustness to agent/connection failures
  - Energy/communication efficiency

- **Design problem**: Favorable tradeoff between performance improvement (benefit) of collaborative behaviors *vs.* costs of collaboration
  - Small world graphs achieve such tradeoff
  - Two-level hierarchy to provide efficient communication

- **Many applications**: communication and sensor networks, networked control, biology, sociology, economics
The Importance of Being Well-Connected

- Local majority voting (Peleg ’96)
  - Each of $n$ citizens has an opinion about voting Yes or No
  - Rule: Each citizen’s vote is based on the majority of its neighbors, including itself
  - What is the minimum number of No-voters that can guarantee a No result?
  - A few number of well connected nodes can determine the outcome of the process!
The Importance of Being Well-Connected (cont.)

White circles: NO voters
Black circles: YES voters

Order of voting matters!

Iterative polling: Oscillation or

If NO voters do not follow the protocol, then 2 NO voters, are sufficient to change the other n-2 YES voters’ opinion.

Even if NO voters follow the protocol a negligible minority of $\sqrt[3]{2n}$ can result in one step convergence to NO.
Consensus Schemes: Vicsek’s Leaderless Coordination Model

- A flock of \( n \) agents moving at the same speed \( s \), but with different headings.
- Each agent updates its heading angle as an average of its neighbors including itself.
- \( D \) is diagonal matrix of nodes’ degrees.
- \( A \) is adjacency matrix.

\[
\theta_i(t | 1) = < \theta_i(t) > = \frac{1}{1 + n_i(t)} \left[ \theta_i(t) + \sum_{j \in N_i(t)} \theta_j(t) \right]
\]

\[
\theta(t + 1) = F_{\sigma(t)} \theta(t)
\]

\[
F_p = (I + D_p)^{-1}(A_p + I)
\]

\( G = (V, E) \)

\( \mathbb{G} = \{G_0, \ldots, G_{M-1}\} \)

\( \mathbb{M} = \{0, \ldots, M - 1\} \)

\( \sigma : \mathbb{N} \cup \{0\} \rightarrow \mathbb{M} \)
The Importance of Being Efficiently-Connected

\[ F_p = (I + D_p)^{-1}(A_p + I) \]

Simple Lattice \( C(n,k) \)

Small world model: rewiring a portion \( \Phi \)

Conjecture: Short paths between distant parts of the network will result in fast information spreading which helps global coordination.

Small world: Slight variation adding \( nk\Phi \)
Spectral Gap gain \[ \frac{1 - SLEM(G(\phi))}{1 - SLEM(G(0))} \] vs \( \phi \)

Simulations support conjecture

In the low range of \( \phi \) (0 < \( \phi \) < 0.01) there is no spectral gap gain observed and the SLEM is almost constant and a drastic increase in the spectral gap is observed around \( \phi = 0.1 \).
Mean Field Explanation for Fast Convergence in Small World Networks

- Experiments show that adding a small number of well chosen links to ring structured graphs results in high convergence rate
- Eigenvalue analysis difficult due to non-symmetric matrices
- Use a slightly different model to explain
  - Start from ring structure \( G_0 = C(n, 1) \)
  - Perturb zero elements in the positive direction by \( \epsilon = \frac{K}{n^\alpha} \) for fixed \( K > 0 \), \( \alpha > 1 \).
  - Perturb the formerly nonzero elements equally, such that the stochastic structure of the \( F \) matrix is preserved
  - Analyze the SLEM as a function of the perturbation as \( \alpha \) varies
Mean Field Explanation for Fast Convergence in Small World Networks

• Refer to the perturbations as ε-shortcuts in the limit $n \to \infty$
  – For $\alpha > 3$ the effect of ε-shortcuts on convergence rate is negligible
  – For $\alpha = 2$ the shortcuts dominantly decrease SLEM
  – For $\alpha = 1$ almost all of the nodes communicate effectively and thus SLEM is very small

• ε-shortcuts are loosely analogous to the shortcuts in Small World networks

• $\alpha = 2$ is the onset of small world effect
Hierarchical Scheme

- **Idea**: selecting a few well-connected, well-protected and controlled agents as leaders
- Dividing the agents into $O(\sqrt{n})$ groups each with $O(\sqrt{n})$ members
- **Leaders**:
  - should be well connected to the members of group
  - should be able to communicate to the other leaders
- The reduction of the size of each group to $O(\sqrt{n})$ results in faster inter-group convergence
- The ease of communication between the leaders upon demand results in fast overall convergence
Distributed Exploration of the Graph Structure

• Hierarchical scheme can be used to design a network structure capable of running distributed algorithms with high convergence speed

• The hierarchical scheme can also be used to improve the performance for a given network structure

• A two stage algorithm is proposed to:
  – Find the most effective choice of local leaders
  – Provide nodes with information about their location with respect to other nodes and leaders and the choice of groups to form
Social Degrees and Local Leaders

• **Social degree of order** $k$ (Generalization of Blondel et al.)
  
  – Social degree of order $2$ : $SD^{(2)}(v)$ = number of neighbors of node $v$
  
  – Social degree of order $k>2$ : $SD^{(k)}(v)$ = number of cycles of length $k$ passing through node $v$

• **Leader nodes:**
  
  – A node is called a leader of order $k$ if its social degree of order $k$ is greater than that of its neighbors
  
  – Social degrees of order 2 and 3 can be determined by a simple query
Regular Nodes and Influence Vectors

• Suppose $M$ nodes are selected as local leaders, the remaining $n-M$ nodes are referred to as regular nodes.

• Regular nodes need to determine how well they are located with respect to local leaders and how they are influenced by them.

• Knowing each node’s distance to leader is useful but not sufficient, since it does not include information on how “well-connected” is a regular node w.r.t leader nodes.

• Influence vector as a metric for well-connectedness
  – Consider a random walk on the graph starting from regular node $i$, with leader nodes as absorbing states, the influence of leader $I_k (k=1, \ldots, M)$ on regular node $i$, is the probability that the random walk hits $I_k$ before other leaders.
Two Stage Semi-decentralized Algorithm

- **Stage I: Determining M leaders**
  - Each node determines its social degrees of orders 2 and 3 by doing a local query
  - If a node finds itself to be a leader of order 2 or 3, sends its degrees to the central authority
  - For a given $0<\alpha<1$ and $\beta=1-\alpha$, and for all the above nodes, the central authority computes their social scores:
    \[ SC(k) = \alpha \cdot SD^{(2)}(k) + \beta \cdot SD^{(3)}(k) \]
    (choice of $\alpha$ and $\beta$ determines the preference between leaders in star-like neighborhoods vs. leaders of better-connected neighborhoods)
  - The central authority selects the $M$ nodes with highest social scores as social leaders and gives them an arbitrary order
Two Stage Semi-decentralized Algorithm (cont.)

• Stage II: Determining influence vectors
  – Based on its order, each leader takes its influence vector to be the fixed vector $e_i$
  – Starting from arbitrary values, all the regular nodes $i=1,\ldots,n-M+1$, update their influence vectors entries $k=1,\ldots,M$ using the following rule:
    \[
    x^k_i(t+1) = \frac{1}{n_i + 1} \left[ x^k_i(t) + \sum_{j \in N_i(t)} x^k_j(t) \right]
    \]
  • For connected graphs, for $t$ large enough, $x^k_i(t)$ converges to the influence of leader $k$ on node $i$
  • Upon calculation of influence vectors, each regular node determines its local leader and stops its communication with neighbors who have other leader. This will result in decomposing the graph into two level hierarchy with efficient communication pattern
Outline

• Networks and Collaboration
• Constrained Coalitional Games
• Security Aware Protocols via NUM
• Topology Matters
• Conclusions and Future Directions
Taxonomy of Networked Systems

Infrastructure / Communication Networks
- Internet / WWW
- MANET
- Sensor Nets
- Robotic Nets
- Hybrid Nets: Comm, Sensor, Robotic and Human Nets

Social / Economic Networks
- Social Interactions
- Collaboration
- Social Filtering
- Economic Alliances
- Web-based social systems

Biological Networks
- Community Epiddemic
- Cellular and Sub-cellular
- Neural
- Insects
- Animal Flocks
Biological Networks
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 consists 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?
Expander Graphs – Ramanujan Graphs

“Sparse but well connected”
Thank you!

baras@isr.umd.edu
301-405-6606
http://www.isr.umd.edu/~baras

Questions?