Networks Out Of Control: Evolution & Dynamics 3
Network Formation Games
Network Formation

- **Models for Networks:**
  - $G(n,p)$
  - $G(n,r)$
  - Lattice/Percolation
  - Watts-Strogatz

- **Commonalities:**
  - Defined by random processes.

- **Alternative mentality:**
  - Consider *motivations* of participants in the network, and model behavior as a function of these goals.
  - Will use game theory!
Games

- **Agents**: two or more participants.

- **Strategies**: options available to the agent.

- **Outcome**: global end result (function of all agent’s actions).

- **Utility**: real-valued function of outcome for a given agent.

In game theory:

- Agents want to maximize their own utility (selfish behavior)

- We study optimality, i.e., outcomes such that the average utility is maximized.

- We study equilibria, i.e., outcomes such that no agent benefits by changing their action.
Example: Coordination Game

- **2 Agents:** you and your friend
- **2 Strategies:** go to the Thai Food truck or go to the Pizza truck.
- **4 Outcomes:**

<table>
<thead>
<tr>
<th></th>
<th>Thai Food</th>
<th>Pizza</th>
</tr>
</thead>
<tbody>
<tr>
<td>Thai Food</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Pizza</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Example: Coordination Game

- **2 Agents:** you and your friend
- **2 Strategies:** go to the Thai Food truck or go to the Pizza truck.
- **4 Outcomes:**

**Utilities:**

<table>
<thead>
<tr>
<th></th>
<th>Thai Food</th>
<th>Pizza</th>
</tr>
</thead>
<tbody>
<tr>
<td>Thai Food</td>
<td>R1, C1</td>
<td>R3, C3</td>
</tr>
<tr>
<td>Pizza</td>
<td>R2, C2</td>
<td>R4, C4</td>
</tr>
</tbody>
</table>
Example: Coordination Game

- **2 Agents:** you and your friend
- **2 Strategies:** go to the Thai Food truck or go to the Pizza truck.
- **4 Outcomes:**

**Utilities:**

<table>
<thead>
<tr>
<th></th>
<th>Thai Food</th>
<th>Pizza</th>
</tr>
</thead>
<tbody>
<tr>
<td>Thai Food</td>
<td>3, 3</td>
<td>-1, -1</td>
</tr>
<tr>
<td>Pizza</td>
<td>-1, -1</td>
<td>2, 2</td>
</tr>
</tbody>
</table>

- **Optimum:** outcome where the sum of the utilities is maximized
Example: Coordination Game

- **2 Agents:** you and your friend
- **2 Strategies:** go to the Thai Food truck or go to the Pizza truck.
- **4 Outcomes:**
- **Utilities:**

<table>
<thead>
<tr>
<th></th>
<th>Thai Food</th>
<th>Pizza</th>
</tr>
</thead>
<tbody>
<tr>
<td>Thai Food</td>
<td>3, 3</td>
<td>-1, -1</td>
</tr>
<tr>
<td>Pizza</td>
<td>-1, -1</td>
<td>2, 2</td>
</tr>
</tbody>
</table>

- **Optimum:** outcome where the sum of the utilities is maximized
Example: Coordination Game

- **2 Agents:** you and your friend
- **2 Strategies:** go to the Thai Food truck or go to the Pizza truck.
- **4 Outcomes:**
- **Utilities:**

<table>
<thead>
<tr>
<th></th>
<th>Thai Food</th>
<th>Pizza</th>
</tr>
</thead>
<tbody>
<tr>
<td>Thai Food</td>
<td>3, 3</td>
<td>-1, -1</td>
</tr>
<tr>
<td>Pizza</td>
<td>-1, -1</td>
<td>2, 2</td>
</tr>
</tbody>
</table>

- **Optimum:** outcome where the sum of the utilities is maximized
- **Stable:** outcome where no one wants to defect.
Example: Coordination Game

- **2 Agents:** you and your friend
- **2 Strategies:** go to the Thai Food truck or go to the Pizza truck.
- **4 Outcomes:**
- **Utilities:**

<table>
<thead>
<tr>
<th></th>
<th>Thai Food</th>
<th>Pizza</th>
</tr>
</thead>
<tbody>
<tr>
<td>Thai Food</td>
<td>3, 3</td>
<td>-1, -1</td>
</tr>
<tr>
<td>Pizza</td>
<td>-1, -1</td>
<td>2, 2</td>
</tr>
</tbody>
</table>

- **Optimum:** outcome where the sum of the utilities is maximized
- **Stable:** outcome where no one wants to *defect.*
Example: Anti-Coordination Game

- **2 Agents:** you and your enemy
- **2 Strategies:** go to the Thai Food truck or go to the Pizza truck.
- **4 Outcomes:**
- **Utilities:**

<table>
<thead>
<tr>
<th></th>
<th>Thai Food</th>
<th>Pizza</th>
</tr>
</thead>
<tbody>
<tr>
<td>Thai Food</td>
<td>-10, 10</td>
<td>3, 0</td>
</tr>
<tr>
<td>Pizza</td>
<td>5, 0</td>
<td>-10, 10</td>
</tr>
</tbody>
</table>
Example: Anti-Coordination Game

- **2 Agents**: you and your enemy
- **2 Strategies**: go to the Thai Food truck or go to the Pizza truck.
- **4 Outcomes**:

- **Utilities**:

<table>
<thead>
<tr>
<th></th>
<th>Thai Food</th>
<th>Pizza</th>
</tr>
</thead>
<tbody>
<tr>
<td>Thai Food</td>
<td>-10, 10</td>
<td>3, 0</td>
</tr>
<tr>
<td>Pizza</td>
<td>5, 0</td>
<td>-10, 10</td>
</tr>
</tbody>
</table>

- **Optimum**: the sum of the utilities is maximized
Example: Anti-Coordination Game

◆ **2 Agents:** you and your enemy

◆ **2 Strategies:** go to the Thai Food truck or go to the Pizza truck.

◆ **4 Outcomes:**

◆ **Utilities:**

<table>
<thead>
<tr>
<th></th>
<th>Thai Food</th>
<th>Pizza</th>
</tr>
</thead>
<tbody>
<tr>
<td>Thai Food</td>
<td>-10, 10</td>
<td>3, 0</td>
</tr>
<tr>
<td>Pizza</td>
<td>5, 0</td>
<td>-10, 10</td>
</tr>
</tbody>
</table>

◆ **Optimum:** the sum of the utilities is maximized
Example: Anti-Coordination Game

- **2 Agents:** you and your enemy
- **2 Strategies:** go to the Thai Food truck or go to the Pizza truck.
- **4 Outcomes:**
- **Utilities:**

<table>
<thead>
<tr>
<th></th>
<th>Thai Food</th>
<th>Pizza</th>
</tr>
</thead>
<tbody>
<tr>
<td>Thai Food</td>
<td>-10, 10</td>
<td>3, 0</td>
</tr>
<tr>
<td>Pizza</td>
<td>5, 0</td>
<td>-10, 10</td>
</tr>
</tbody>
</table>

- **Optimum:** the sum of the utilities is maximized
- **Stable:** no one wants to *defect.*
Game Theory

- Any interaction can be modeled by a game
  - In particular: Networks!
Network Formation Games

- **Agents:** \( N = \{1, \ldots, n\} \)

- **Strategies:** A subset \( S_i \) of \( N \times N \) (potential edges)
  - We also refer to the strategy vector \( S = (S_1, S_2, \ldots, S_N) \).

- **Outcome:** A network \( G \) where \( V = N \) and \( E = \bigcup_i S_i \).

- **Utility:** real-valued functions \( u_i \) of the network \( G \).

- **Objectives:**
  - Agents select \( S_i \) in order to maximize \( u_i \) (selfish behavior)
  - We study optima: graphs \( G \) such that the sum of utilities is maximized.
  - We study stability: graphs \( G \) such that no agent wants to change their \( S_i \).
Network Formation Games

- **Local connection game (social networks)**
  - Agent can build edges from itself to other nodes (at a cost), and wants to be connected to all nodes via short paths.

- **Coauthor game (business/collaboration networks)**
  - Agents can partner together, but the more partners an agent has the less resources she has to put into the partnership.

- **Global connection game (infrastructure networks)**
  - Agents are no longer nodes, each agent wants to ensure some s-t path is built, and can build edges anywhere (at a shared cost).
Local Connection Game
Local Connection Game

- Local connection game (social networks)
  - Agent can build edges from itself to other nodes (at a cost), and wants to be connected to all nodes via short paths.

- Each of the $n$ agents is a node.
  - The strategy space for node $u$ is a subset $S_u$ of $V$.
  - This corresponds to building $uv$ edges for all $v$ in $S_u$.
  - Each edge has a cost $\alpha$
  - The total distance to other nodes (using all edges) also incurs a cost.
  - Overall, node $u$ wishes to minimize:
    $\alpha n_u + \sum_v d(u,v)$
    where $n_u$ is the number of connections $u$ makes and $d(u,v)$ is the distance between $u$ and $v$.

- The cost of the network is the sum of the costs of all agents:
  $\sum_u (\alpha n_u + \sum_v d(u,v))$
  $= \alpha m + \sum_{u,v} d(u,v)$
  where $m = |E|$.
Optimal Networks

- An optimal network minimizes:
  \[ \alpha m + \sum_{u \neq v} d(u, v) \]

- Goal: Characterize optimal networks.

- Approach:
  - Lower bound the cost,
  - Give network(s) that attain the lower bound.

- Lemma: optimal networks
  - If \( \alpha \leq 2 \), then the complete graph is an optimal network.
  - If \( \alpha \geq 2 \), then the star network is an optimal network.
Stable Networks

- Recall: Node $u$ wishes to minimize:
  $$\alpha n_u + \sum_v d(u,v)$$

- Goal: Find some stable networks

- Approach:
  - Is the complete graph stable?
  - Is the star graph stable?
    - Assume center node pays for all edges.
    - (in fact true for any star)

- Lemma:
  - If $\alpha \leq 1$, then the complete graph is stable.
  - If $\alpha \geq 1$, then the star graph is stable.
The Price of Stability is the ratio: \[
\frac{\text{value of best equilibrium}}{\text{value of optimal solution}}
\]

**Lemma:**
- If \( \alpha \geq 2 \), then the optimal network is a star.
- If \( \alpha < 2 \), then the optimal network is a complete graph.

**Lemma:**
- If \( \alpha \geq 1 \), then any star is a Nash equilibrium.
- If \( \alpha \leq 1 \), then the complete graph is a Nash equilibrium.

**Theorem:**
- For \( \alpha \geq 2 \) and \( \alpha \leq 1 \), the price of stability is 1.
- For \( 1 < \alpha < 2 \)?
  - Recall: \( \text{opt value} \geq (\alpha - 2)m + 2n(n-1) \)
  - The price of stability is at most 4/3.
Price of Anarchy

The Price of Anarchy is the ratio: \[
\frac{\text{value of worst equilibrium}}{\text{value of optimal solution}}
\]

Lemma:
- If \( \alpha \geq 2 \), then the optimal network is a star.
- If \( \alpha < 2 \), then the optimal network is a complete graph.

Lemma:
- If \( \alpha \geq 1 \), then any star is a Nash equilibrium.
- If \( \alpha \leq 1 \), then the complete graph is a Nash equilibrium.
Exercise
Local Connection Game

- Show that for $\alpha < 1$ the complete graph is the unique equilibrium.
  - Does this imply anything about the price of anarchy?

- Construct a Nash equilibrium that is not a star for some $N$ and some $\alpha > 2$. 
Co-Author Game
Coauthor Game

- Coauthor game (business/collaboration networks)
  - Agents can partner together, but the more partners an agent has the less resources she has to put into the partnership.

- Each of the n agents is a node.
  - Nodes benefit from partnerships (direct edge connections) to others due to “collaboration”.
  - The amount a node benefits is inversely proportional to the amount of partnerships (i.e., degree) one has.
Coauthor Game

- Let $n_i$ be the degree of node $i$

- Node $i$ wishes to maximize:

$$u_i(g) = \sum_{j: ij \in g} \left[ \frac{1}{n_i} + \frac{1}{n_j} + \frac{1}{n_i n_j} \right]$$

- If $n_i = 0$, $u_i(g) = 0$

- A network is optimal if it maximizes:

$$\sum_{i \in N} u_i(g) = \sum_{i: n_i > 0} \sum_{j: ij \in g} \left[ \frac{1}{n_i} + \frac{1}{n_j} + \frac{1}{n_i n_j} \right],$$
Optimal Networks

A network is optimal if it maximizes:

\[ \sum_{i \in N} u_i(g) = \sum_{i: n_i > 0} \sum_{j: i \neq j \in g} \left[ \frac{1}{n_i} + \frac{1}{n_j} + \frac{1}{n_i n_j} \right], \]

Goal: Find an optimal network:

Approach:

First, upper bound \( \sum_{i \in N} u_i(g) \)

\[ \sum_{i \in N} u_i(g) \leq 3N \]

Show some network attains this bound:

An optimal network on 2K agents consists of K pairs of nodes.

Is this network is an equilibrium?

No
Stable Networks

- Node $i$ wishes to maximize:
  \[
  u_i(g) = \sum_{j: ij \in g} \left[ \frac{1}{n_i} + \frac{1}{n_j} + \frac{1}{n_i n_j} \right] = 1 + \left( 1 + \frac{1}{n_i} \right) \sum_{j: ij \in g} \frac{1}{n_j}.
  \]

- Node $i$ would like to link to node $j$ if:
  \[
  \frac{n_i + 2}{n_j + 1} \geq \frac{1}{n_i} \sum_{k: k \neq j, ik \in g} \frac{1}{n_k}.
  \]

- Assume that $n_j \leq n_i$,
  - Does $i$ want to link to $j$?
    - Yes! What does this mean for stable networks?

- The only stable networks are complete networks.
Price of Stability / Price of Anarchy

- Recall: The optimal network has cost
  \[ \sum_{i \in N} u_i(g) \leq 3\hat{N} \]

- Recall: The only stable networks are complete networks.

- The price of stability is:
  - Is > 1/3

- The price of anarchy is:
  - In this case, is the same!
Does this notion of stability make sense?

- We showed conditions under which \( i \) wants to connect to \( j \). Does \( j \) also want to connect to \( i \)?
- In some cases, the notion of stability may be incomplete.

- **Pairwise stability:**
  - For all \( ij \) in \( g \), we have \( u_i(g) > u_i(g-ij) \) and \( u_j(g) > u_j(g-ij) \).
  - For all \( ij \) not in \( g \), we have \( u_i(g) < u_i(g+ij) \) and \( u_j(g) < u_j(g+ij) \).

- **Theorem:** The pairwise-stable networks can be decomposed into fully connected components with no cross-component edges such that if the number nodes in each of the \( t \) components is \( k_1 > k_2 > \ldots > k_t \), then \( k_{i-1} > k_i^2 \).

- **Proof:** Similar analysis as before, but show that \( j \) in the smaller component only wants to connect to \( i \) if the above is satisfied.
In the local connection game, the more connections other agents build, the fewer connections we build.

This is a game of *strategic substitutes*.

In the coauthor game, the more connections other agents build, the more connections we build.

This is a game of *strategic complements*.
Global Connection Game
Global Connection Game

- Agents are no longer nodes in the network, external agents with some desire over global properties of the network.
  - For example, the vertices are neighborhoods and edges are roads, and you would like the path from your home to your office to be well-maintained.

- There are $k$ agents, each with a source $s_i$ and sink $t_i$ node.
  - The strategy space for agent $i$ is the set of all paths $P$ from $s_i$ to $t_i$.
    - We let $P_i$ be the path she selects.
  - Let $k_e$ be the number of agents using edge $e$:
    $$u_i = \sum_{e \in P_i} \frac{c_e}{k_e}$$

- The sum of the agents costs is
  $$\sum u_i = \sum_{e \in \text{some } P_i} c_e .$$

- Maximizing this quantity is known as the Steiner tree problem.
Price of Anarchy

- For the given example:
  - What is the optimal solution?
  - What are the equilibria?
  - What is the price of anarchy?
  - What is the price of stability?

- Theorem: In any global connection game, the price of anarchy is at most $k$.
  - Proof:
    - Let $S$ be a stable network and $S^*$ be an optimal network,
    - Let $w_i$ be the weight of the shortest path from $s_i$ to $t_i$, and let $w(P_i^*)$ be the weight of the path the optimal solution selects for $i$.

- What about the price of stability?
Price of Stability

- For the given example:
  - What is the optimal solution?
  - What are the equilibria?
  - What is the price of stability?

- Theorem:
The price of stability is always at most $H_k$.

- Proof: Potential Method for Games
Theorem: A pure Nash equilibrium always exists and the price of stability is at most $H_k$.

Potential function method:
- Define a function on edges $\Phi_e = c_e H_{ke}$ and $\Phi = \Sigma_e \Phi_e$.
- If $i$ unilaterally changes its strategy to $S'$, then can show that
  \[ \Phi(S) - \Phi(S') = u_i(S') - u_i(S) \]
  (this is called a potential function).
- Can also show that:
  \[ \text{cost}(S) \leq \Phi(S) \leq H_k \text{cost}(S) \]

Theorem: A strategy that minimizes $\Phi$ is stable.

Theorem: If $A \text{cost}(S) \leq \Phi(S) \leq B \text{cost}(S)$, then the price of stability is at most $B/A$. 

Price of Stability
Local Connection Game:
The Price of Anarchy
Price of Anarchy

- The Price of Anarchy is the ratio: \( \frac{\text{value of worst equilibrium}}{\text{value of optimal solution}} \)

- Lemma:
  - If \( \alpha \geq 2 \), then the optimal network is a star.
  - If \( \alpha < 2 \), then the optimal network is a complete graph.

- Lemma:
  - If \( \alpha \geq 1 \), then any star is a Nash equilibrium.
  - If \( \alpha \leq 1 \), then the complete graph is a Nash equilibrium.

- Theorem: The price of anarchy is at most \( O(\sqrt{\alpha}) \).
  - Bound the diameter of an equilibrium graph.
  - Use this to bound the price of anarchy.
Price of Anarchy

- An optimal network minimizes:
  \[ \alpha |E| + \sum_{u \neq v} d(u,v) \]

- Theorem: The price of anarchy is at most \( O(\sqrt{\alpha}) \).
  - Bound the diameter of an equilibrium graph.
  - Use this to bound the price of anarchy.

- Lemma: If a Nash equilibrium has diameter \( d \), then its cost is at most \( O(d) \) times optimal.
Theorem: The price of anarchy is at most $O(\sqrt{\alpha})$.
- Bound the diameter of an equilibrium graph.
- Use this to bound the price of anarchy.

Lemma: If a Nash equilibrium has diameter $d$, then its cost is at most $O(d)$ times optimal.

Lemma: The diameter of a Nash equilibrium is at most $2\sqrt{\alpha}$.

Theorem: Price of anarchy is $O(1)$ when $\alpha$ is $O(\sqrt{n})$. 