

Network Games and Algorithmic Game Theory Seminar

Week 1 - Network Formation: Stability, Efficiency, Difficulty

1 Models of Network Formation

In many economic models, the interaction structure is fixed in advance. Consumers face the same market price, firms compete in a given market, or players interact in a game with a fixed set of opponents. But many important economic relationships are chosen. Firms decide whom to trade with, researchers choose collaborators, banks choose counterparties, and individuals decide whom to communicate with.

If we allow agents in a model to choose their links, then the networks that result are themselves an outcome we are interested in studying. We can consider both economic and computational kinds of questions:

- (i) *Economic*: Which links do agents choose to make, and why? When is a network ‘stable’? When is a network ‘efficient’?
- (ii) *Computational*: Given the relevant conditions for ‘stability’ or ‘efficiency’, how difficult is it to find such a network?

We will first discuss two models of network formation from the economics literature, Jackson and Wolinsky (1996) and Bala and Goyal (2000). We will then look at a model from the theoretical computer science literature – Fabrikant et al. (2003) – to illustrate how network formation quickly becomes an algorithmic problem.

2 Graph Theory and Game Theory background

Let the set of agents be

$$\mathcal{N} = \{1, 2, \dots, n\}.$$

An undirected network is a graph $g = (\mathcal{N}, E)$, where E is a set of unordered pairs $\{i, j\}$ with $i \neq j$. We write $ij \in g$ when agents i and j are directly linked.

For each node i , let

$$N_i(g) = \{j : ij \in g\}$$

be the set of i ’s neighbors and let $d_i(g) = |N_i(g)|$ be its degree. For $i \neq j$, let $\ell_{ij}(g)$ be the length of a shortest path from i to j . If no path exists, set $\ell_{ij}(g) = \infty$.

Three networks will appear repeatedly: the *empty network* with no links, the *complete network* with every possible link, and a *star* in which one central node is linked to all $n - 1$ other nodes and there are no other links.

2.1 Payoffs and link changes

Suppose each agent i has payoff $u_i(g)$ in network g . If $ij \notin g$, write $g + ij$ for the graph obtained by adding the link ij . If $ij \in g$, write $g - ij$ for the graph obtained by deleting it.

The central modeling question is how links form. One possibility is bilateral consent: a link can be created only if both endpoints agree, while either endpoint can sever an existing link. This leads to pairwise stability.

Definition 1 (Pairwise stability). *A network g is pairwise stable if:*

(i) *for every existing link $ij \in g$,*

$$u_i(g) \geq u_i(g - ij) \quad \text{and} \quad u_j(g) \geq u_j(g - ij),$$

so neither endpoint wants to sever the link; and

(ii) *for every absent link $ij \notin g$, if one endpoint would strictly gain by adding the link, the other would strictly lose. Equivalently, it is never the case that*

$$u_i(g + ij) > u_i(g) \quad \text{and} \quad u_j(g + ij) \geq u_j(g),$$

or vice versa.

This is a local notion. Stability means that no single existing link should be cut and no single missing link should be mutually acceptable. It does *not* rule out coordinated deviations involving many links at once. (Compare this with the notion of Nash Equilibrium.)

Definition 2 (Efficiency). *Let aggregate welfare be*

$$W(g) = \sum_{i=1}^n u_i(g).$$

A network g^ is efficient if $W(g^*) \geq W(g)$ for every feasible network g .*

The distinction between stability and efficiency is one of the central themes of network economics. Stability is about incentives of individual agents; efficiency is about the network as a whole.

3 The Jackson–Wolinsky connections model

Jackson and Wolinsky (1996) introduced a particularly transparent model in which direct and indirect connections are useful, but benefits decay with distance. People get value by having closer connections, but they incur a cost by having many connections

Let $0 < \delta < 1$ measure how much value survives each step along a path, and let $c > 0$ be the cost to each endpoint of maintaining a direct link. Agent i 's payoff is

$$u_i(g) = w_{ii} + \sum_{j \neq i} \delta^{t_{ij}} w_{ij} - \sum_{j: ij \in g} c_{ij} \tag{1}$$

w_{ij} is the intrinsic value of individual j to individual i . t_{ij} is the number of nodes between i and j , and c_{ij} is the cost of maintaining a direct connection between i and j .

We will simplify this model a little bit, setting $w_{ij} = 1$ and $c_{ij} = c \forall i, j$. This gives us:

$$u_i(g) = 1 + \sum_{j \neq i: \ell_{ij}(g) < \infty} \delta^{\ell_{ij}(g)} - c d_i(g). \quad (2)$$

A direct neighbor contributes δ . A friend-of-a-friend at distance two contributes δ^2 . Someone at distance three contributes δ^3 , and so on.

The model captures a simple tradeoff. More links are useful because they shorten distances and may connect otherwise disconnected components. But links are costly, and an indirect route may provide enough benefit that a direct link is not worth paying for.

3.1 A four-node example

Take $n = 4$ and consider a star with hub 1. The hub's payoff is

$$u_1 = 3\delta - 3c.$$

Each leaf has one direct connection to the hub and two indirect connections to the other leaves, so

$$u_i = \delta + 2\delta^2 - c, \quad i = 2, 3, 4.$$

Would two leaves, say 2 and 3, add a direct link? Before adding it, each receives δ^2 from the other. After adding it, each receives δ but pays an extra cost c . The change in payoff is therefore

$$\Delta u_2 = \Delta u_3 = (\delta - \delta^2) - c.$$

Hence the two leaves do not want to add a direct edge when

$$c \geq \delta - \delta^2. \quad (3)$$

Would the hub cut one of its links? In the star, cutting the link to a leaf disconnects that leaf from the hub, so the hub loses direct benefit δ and saves c . Thus the hub does not cut when

$$c \leq \delta. \quad (4)$$

Combining (3) and (4), a star is pairwise stable whenever

$$\delta - \delta^2 \leq c \leq \delta.$$

This simple calculation already illustrates the role of network structure: the value of a link depends on what indirect paths are already available.

3.2 Characterizing stable networks

Proposition 1 (Pairwise stability in the symmetric connections model). *For $0 < \delta < 1$:*

- (i) *If $c < \delta - \delta^2$, the complete network is the unique pairwise-stable network.*

- (ii) If $\delta - \delta^2 < c < \delta$, a (full) star is pairwise stable, though other pairwise-stable networks may also exist.
- (iii) If $c > \delta$, the empty network is pairwise stable. Moreover, any nonempty pairwise-stable network must have every connected node be degree 2 or more.

Proof idea for part (i). Suppose ij is missing. If i and j are already connected indirectly, their shortest possible indirect distance is at least two, so adding ij raises each endpoint's gross benefit from the other by at least $\delta - \delta^2$. If $c < \delta - \delta^2$, both gain from adding the missing link. Thus a pairwise-stable graph can have no missing link, so it must be complete. $\square$

Proof idea for part (ii). The center of a star will not delete a link to a leaf because $c < \delta$. A leaf is also not willing to delete its link, since then it would have no direct or indirect connections. However, two leaves do not wish to form a direct link, as they already receive δ^2 , and $\delta - \delta^2 < c$. $\square$

Proof idea for part (iii). Clearly, the empty network is stable. For a non-empty network, consider a node i of degree 1, connected to node j . Node j only gets $\delta < c$ from node i , thus j severs the connection. $\square$

Note that already, pairwise stability is only *somewhat* capable of restricting possible networks.

4 Efficiency

Now consider total welfare, W . Each undirected link costs c to each of its two endpoints, so its total resource cost is $2c$.

For the complete graph K_n ,

$$W(K_n) = n(n-1)(\delta - c),$$

because n agents get $\delta - c$ from each of their $n - 1$ connections.

For a (full) star S_n , there are $n - 1$ links. There are $2(n - 1)$ ordered hub-leaf pairs at distance one and $(n - 1)(n - 2)$ ordered leaf-leaf pairs at distance two. Therefore

$$W(S_n) = 2(n - 1)\delta + (n - 1)(n - 2)\delta^2 - 2(n - 1)c.$$

For the empty graph,

$$W(\emptyset) = 0.$$

Theorem 1 (Efficient network structures in the connections model). *Ignoring knife-edge equality cases, the efficient network is:*

- (i) the complete graph if

$$c < \delta - \delta^2;$$

- (ii) an encompassing star if

$$\delta - \delta^2 < c < \delta + \frac{n-2}{2}\delta^2;$$

(iii) the empty graph if

$$c > \delta + \frac{n-2}{2}\delta^2.$$

Why these three networks? First, let's review some graph theory. If we have n vertices, the minimal possible number of edges to form a fully-connected graph is $n - 1$. A graph with exactly $n - 1$ edges is a 'tree'. Among 'trees', a full star has the smallest possible diameter of any non-complete connected graph, a diameter of 2.

So if it is worthwhile to connect, we may want a star. When would we add a link to a star? When $2(\delta - \delta^2)$, the benefit (to *both* nodes) of going from a connection of distance 2 to distance 1, is greater than $2c$. Then it is worth filling in all such missing links, leading to the complete graph. When $c > \delta - \delta^2$, those extra links are wasteful, so among connected networks the star is preferred.

Now, when would we remove a link from a star? Well, a star gives us:

$$W(S_n) = 2(n-1)\delta + (n-1)(n-2)\delta^2 - 2(n-1)c.$$

from the hub's $(n-1)$ direct links, and $(n-1)(n-2)$ indirect links of distance 2 through the hub. Dividing by $2(n-1)$, we get the threshold:

$$\delta + \frac{n-2}{2}\delta^2 - c > 0 \rightarrow \delta + \frac{n-2}{2}\delta^2 > c.$$

If c is above that threshold, even the cost-minimizing connected architecture is not worth building, so the empty network is efficient. $\square$

5 Stability versus efficiency

The network that is stable may not be the network that is efficient. Consider the following case:

Suppose

$$\delta < c < \delta + \frac{n-2}{2}\delta^2.$$

Then a star is efficient (as we characterized above), but the hub privately pays c for each leaf and receives only direct benefit $\delta < c$ from that leaf; thus the hub wants to sever links. Should the hub choose to sever links, we would have a (negative) network. When the hub maintains a link to one leaf, that link does not merely benefit the hub and the leaf. It also allows all the other leaves to reach that leaf in two steps. The hub does not internalize those third-party benefits.

6 A different formation rule: Bala and Goyal

Pairwise stability assumes that bilateral consent is necessary for link creation. Bala and Goyal (2000) take a different approach: an agent can *unilaterally* pay to create edges, but both agents on either side of the edge can benefit from it. Moreover, Bala and Goyal model network formation as a non-cooperative game, where a player's strategy is a set of nodes that they choose to connect with. Instead of pairwise stability, they use Nash equilibrium to define stability. Here is their "two-way flow model":

Let $N = \{1, \dots, n\}$ be the set of agents. Agent i 's strategy is a set

$$s_i \subseteq N \setminus \{i\},$$

where $j \in s_i$ means that i creates, or *sponsors*, a link with j .

A link between i and j exists whenever at least one endpoint sponsors it:

$$ij \in g \iff j \in s_i \text{ or } i \in s_j.$$

The model is called a *two-way flow* model because, once a link exists, benefits can flow in both directions, regardless of which who paid for the link. Moreover, benefits can pass through intermediate agents. Consequently, an agent can obtain benefits from everyone in her connected component.

Let $C_i(g)$ denote the connected component containing i , and let $d_i^+(g) = |s_i|$ denote the number of links sponsored by i . In the simplest linear version of the model, utility is

$$u_i(g) = |C_i(g)| - c d_i^+(g),$$

where $c > 0$ is the cost of sponsoring one link.

Proposition 2 (Bala and Goyal, 2000). *In the linear two-way flow model, if $0 < c < 1$, the unique nonempty strict-Nash architecture is a center-sponsored star. If $c > 1$, the empty network is the unique strict-Nash architecture.*

Proof idea. When $0 < c < 1$, the center wants to keep each spoke: access to each leaf yields $1 > c$, its cost. Each leaf already is linked to everyone else, so it has no reason to sponsor an additional edge.

The word ‘strict’ is doing a lot of work here. Consider a star where each leaf is paying for its connection to the hub. If a leaf deletes its connection, then it loses access to everyone else. However, that leaf could just move its link to go to another leaf. This yields the same payoff as connecting to the hub, so the leaf is indifferent. Such indifference is ruled out by strict-Nash.

7 A TCS model of strategic network creation

Fabrikant et al. (2003) introduce a network-creation game motivated by the decentralized formation of Internet-like networks. Each player is a node. Player i may pay to create edges to other nodes. Once edges are purchased, communication uses the resulting undirected graph.

Let S_i be the set of links bought by player i . Each purchased edge costs $\alpha > 0$. Player i also dislikes being far from other nodes. A standard version of the cost is

$$C_i(S) = \alpha |S_i| + \sum_{j \neq i} d_G(i, j), \tag{5}$$

where G is the graph induced by everyone's purchased edges and $d_G(i, j)$ is shortest-path distance.

This objective resembles the connections model, but with a different functional form. A player trades off the cost of buying edges against the benefit of reducing distances to everyone else.

A pure Nash equilibrium is a profile $S = (S_1, \dots, S_n)$ such that no single player can reduce its own cost by replacing its chosen set of links with another set.

At first glance, one might imagine finding equilibrium by repeated best responses: pick a player, compute the best set of links for that player, update, and continue. But even the first step can be computationally difficult.

Proposition 3 (Fabrikant et al.). *Computing a player's exact best response in the network-creation game is NP-hard.*

We will show that this is an NP-hard problem by showing that finding an exact best response is equivalent to finding a minimum dominating set.

Definition 3 (Dominating set). *Given a graph $H = (V, E)$, a set $D \subseteq V$ is a dominating set if every node in $V \setminus D$ has a neighbor in D . A minimum dominating set is one with the smallest possible cardinality.*

Proof idea. Suppose we have a graph $H = (V, E)$ with $|V| = m$, and we want to find its minimum dominating set. We can turn this into an instance of the network-creation by suppose that a set of players and their chosen strategies formed H , and now we add a new player i . Further suppose that no existing player chooses to buy a link to i . Thus i is choosing a set $S \subseteq V$ of vertices to purchase links to. We set the edge-cost parameter such that $1 < \alpha < 2$.

Step 1: Any best response must be a dominating set.

Suppose S is not a dominating set. Then there exists some $v \in V$ such that $v \notin S$ and $N_H(v) \cap S = \emptyset$. Therefore v cannot be reached from i in one or two steps, so $d_G(i, v) \geq 3$.

But i can purchase an additional edge directly to v , which would cost α but reduces her distance to v from at least 3 to 1, lowering her distance cost by at least 2. Therefore

$$c_i(S \cup \{v\}) - c_i(S) \leq \alpha - 2 < 0.$$

So S cannot have been a best response. Hence every best response S^* must dominate H .

Step 2: Among dominating sets, the player prefers one of minimum size.

Suppose S is a dominating set with $|S| = k$. Every vertex in S is distance 1 from i . Every vertex outside S has a neighbor in S , and is therefore distance 2 from i .

Thus the distance part of i 's cost is $k + 2(m - k)$ while the cost of purchasing the k links is, αk . Hence

$$c_i(S) = \alpha k + k + 2(m - k) = 2m + (\alpha - 1)k$$

Because $\alpha > 1$, this expression is strictly increasing in k . Therefore, among all dominating sets, i 's best response is precisely a dominating set of minimum cardinality.

Thus an algorithm that computed an exact best response for i could be used to compute a minimum dominating set of H . Since minimum dominating set is NP-hard, computing an exact best response is also NP-hard. $\square$

References

- Bala, Venkatesh and Sanjeev Goyal (2000). “A Noncooperative Model of Network Formation”. In: *Econometrica* 68.5, pp. 1181–1229. DOI: [10.1111/1468-0262.00155](https://doi.org/10.1111/1468-0262.00155).
- Fabrikant, Alex et al. (2003). “On a Network Creation Game”. In: *Proceedings of the Twenty-Second Annual ACM Symposium on Principles of Distributed Computing*, pp. 347–351. DOI: [10.1145/872035.872088](https://doi.org/10.1145/872035.872088).
- Jackson, Matthew O. and Asher Wolinsky (1996). “A Strategic Model of Social and Economic Networks”. In: *Journal of Economic Theory* 71.1, pp. 44–74. DOI: [10.1006/jeth.1996.0108](https://doi.org/10.1006/jeth.1996.0108).