

# Networks and Markets — Lecture 9: Games, Incentives, and Diffusion on Networks

Nikhil Garg, Cornell Tech

## Today: strategic behavior on networks

Two motivating questions:

- Which route should I take when driving?
- How do incentives (prices/tolls/platform rules) change network outcomes?

We model routing as a game where each driver chooses a path to minimize their own travel time.

## 1 Routing-game model

Let total traffic demand be normalized to 1. Each edge  $e$  has latency (cost)  $c_e(x_e)$  depending on flow  $x_e$ .

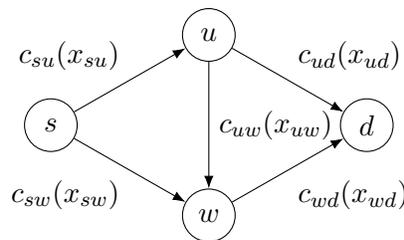


Figure 1: A routing game: users choose  $s$ - $d$  paths and each edge cost depends on congestion.

**Social cost (system objective).** For a flow vector  $x$ , define

$$C(x) = \sum_e x_e c_e(x_e).$$

With total demand normalized to 1, this equals the average travel time per driver.

**Wardrop/Nash equilibrium (nonatomic).** At equilibrium, all used paths have equal and minimal latency. No individual driver can improve by unilaterally switching route.

### Parallel-roads warm-up

Suppose two parallel roads from  $s$  to  $d$ :

$$c_1(x_1) = 5x_1, \quad c_2(x_2) = 2 + 0.001x_2, \quad x_1 + x_2 = 1.$$

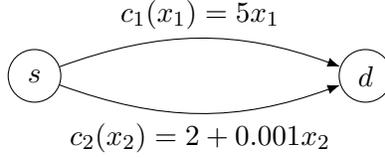


Figure 2: Two parallel roads with heterogeneous latency functions.

At the Nash equilibrium (both used),

$$c_1(x_1) = c_2(x_2).$$

So

$$5x_1 = 2 + 0.001x_2 = 2 + 0.001(1 - x_1) = 2.001 - 0.001x_1,$$

hence

$$5.001x_1 = 2.001 \quad \Rightarrow \quad x_1^* \approx 0.4001, \quad x_2^* \approx 0.5999.$$

## 2 Pigou network and price of anarchy

Consider two routes:

$$c_1(x) = x, \quad c_2(x) = 1, \quad x_1 + x_2 = 1.$$

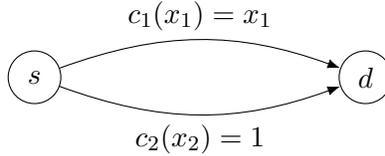


Figure 3: Pigou network: one congestible route and one constant-latency route.

### Nash equilibrium

In a nonatomic routing game, the Nash equilibrium is a flow pattern where no individual driver can reduce their travel time by unilaterally changing their route.

Thus, as before, we have  $c_1(x_1) = c_2(x_2)$  at equilibrium (if both routes are used; if only one route is used, then its cost at 0 traffic should be at least as large or bigger than the other route's cost).

If all drivers choose route 1, then  $c_1(1) = 1 = c_2$ . So an equilibrium is:

$$x_1^{\text{NE}} = 1, \quad x_2^{\text{NE}} = 0.$$

Total (average) travel cost:

$$C_{\text{NE}} = x_1 c_1(x_1) + x_2 c_2(x_2) = 1 \cdot 1 + 0 \cdot 1 = 1.$$

## Social optimum

Minimize total cost

$$C(x_1) = x_1^2 + (1 - x_1) \cdot 1 = x_1^2 + 1 - x_1.$$

First-order condition:

$$\frac{dC}{dx_1} = 2x_1 - 1 = 0 \Rightarrow x_1^{\text{OPT}} = \frac{1}{2}, \quad x_2^{\text{OPT}} = \frac{1}{2}.$$

Optimal cost:

$$C_{\text{OPT}} = \left(\frac{1}{2}\right)^2 + \left(1 - \frac{1}{2}\right) = \frac{1}{4} + \frac{1}{2} = \frac{3}{4}.$$

### 2.1 Price of anarchy

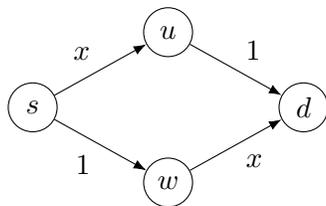
$$\text{PoA} = \frac{C_{\text{NE}}}{C_{\text{OPT}}} = \frac{1}{3/4} = \frac{4}{3}.$$

**4/3 bound.** It turns out that this is the worst-case PoA for nonatomic selfish routing with affine latency functions. With atomic players or more general nonlinear costs, PoA can be different and often larger.

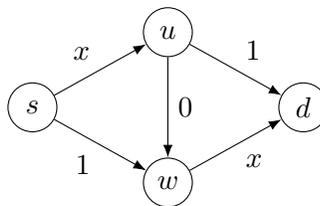
## 3 Braess paradox

Network setup:

- top-left edge latency  $x$ ,
- bottom-right edge latency  $x$ ,
- top-right edge latency 1,
- bottom-left edge latency 1,
- demand = 1 from  $s$  to  $d$ .



Without middle edge



With zero-latency middle edge

Figure 4: Braess network before and after adding the middle edge.

**Without the middle edge** There are two routes:

- top route: latency  $x_1 + 1$ ,
- bottom route: latency  $1 + x_2$ ,

with  $x_1 + x_2 = 1$ . At Wardrop equilibrium (both routes used):

$$x_1 + 1 = 1 + x_2 \Rightarrow x_1 = x_2 = \frac{1}{2}.$$

So each used route has latency

$$L_{\text{no middle}} = \frac{1}{2} + 1 = \frac{3}{2}.$$

**Add a zero-latency middle edge** Now add an extra edge (latency 0) between the two middle nodes. There are now three path types:

- top:  $s \rightarrow u \rightarrow d$  with flow  $x_t$ ,
- bottom:  $s \rightarrow w \rightarrow d$  with flow  $x_b$ ,
- zig-zag:  $s \rightarrow u \rightarrow w \rightarrow d$  with flow  $x_m$ ,

with  $x_t + x_b + x_m = 1$ .

Path latencies are

$$L_t = (x_t + x_m) + 1, \quad L_b = 1 + (x_b + x_m), \quad L_m = (x_t + x_m) + 0 + (x_b + x_m) = 1 + x_m.$$

At equilibrium, all flow goes on the zig-zag path:

$$x_m = 1, \quad x_t = x_b = 0.$$

Then

$$L_t = L_b = L_m = 2.$$

So the equilibrium travel time rises from  $\frac{3}{2}$  to 2 after adding the road.

So adding capacity (a new road) can make equilibrium outcomes worse. This is the *Braess paradox*.

## 4 Congestion pricing

Return to Pigou: latencies  $x$  and 1. Add toll  $p$  to the variable-latency route, so perceived cost is  $x + p$ .

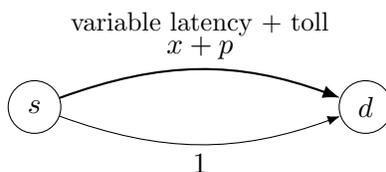


Figure 5: Congestion pricing on the Pigou network.

**Pigouvian toll formula (general).** Let total cost on edge  $e$  be

$$T_e(x) = x c_e(x).$$

Then

$$\frac{dT_e}{dx} = c_e(x) + x c'_e(x).$$

Interpretation:

- $c_e(x)$  is the marginal user's own travel cost (private cost),
- $x c'_e(x)$  is the extra delay imposed on existing users, who have mass  $x$  (external cost).

So the marginal externality imposed by an infinitesimal extra unit of flow is

$$t_e(x) = x c'_e(x).$$

So a standard efficiency rule is: set toll equal to this externality.

**Equilibrium under toll** If both routes are used:

$$x + p = 1 \Rightarrow x(p) = 1 - p.$$

**Toll for efficiency** Social optimum needs  $x^* = \frac{1}{2}$ . Set  $x(p) = \frac{1}{2}$ :

$$1 - p = \frac{1}{2} \Rightarrow p^* = \frac{1}{2}.$$

This matches the Pigouvian formula here: on the variable edge,  $c(x) = x$ , so

$$t(x) = x c'(x) = x,$$

and at the optimum  $x^* = \frac{1}{2}$ , we get  $t^* = p^* = \frac{1}{2}$ .

**Revenue** With unit demand, toll revenue is

$$R(p) = p \cdot x(p) = p(1 - p),$$

which is maximized at  $p = \frac{1}{2}$ .

## Takeaways

- Individual route choice can be inefficient relative to social optimum.
- Price of anarchy quantifies this inefficiency.
- More roads do not always reduce congestion (Braess paradox).
- Proper congestion pricing can restore efficient flow.

## 5 Diffusion and cascades on networks

Many network outcomes are not about routing but about *adoption*: new products, behaviors, ideas, or misinformation spreading through edges.

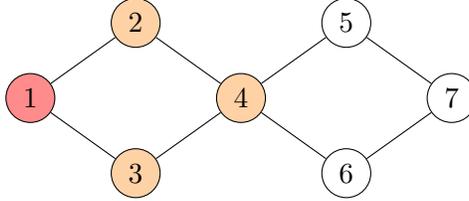


Figure 6: Diffusion intuition: a small seed set (darker nodes) can trigger larger local adoption.

## 5.1 Diffusion models

**Linear threshold model.** Each node  $i$  has a threshold  $\theta_i \in [0, 1]$ . Node  $i$  adopts when the fraction (or weighted share) of adopted neighbors exceeds  $\theta_i$ . This captures coordination and peer-pressure effects.

**Independent cascade model.** When a node first adopts, it gets one chance to activate each inactive neighbor with probability  $p_{ij}$ . Newly activated nodes try in the next round, and the process continues. This captures probabilistic contagion.

**Cascade intuition** Even a small initial seed set can trigger a large cascade if:

- influential/high-degree bridge nodes adopt early, or
- thresholds are low in key parts of the graph.

Graph structure (communities, bridges, degree heterogeneity) strongly affects spread size.

**Influence maximization (platform/marketing view)** Given budget  $k$ , choose seed nodes  $S$  with  $|S| = k$  to maximize expected total adoption.

## 6 Influence maximization

Diffusion models tell us how adoption spreads *given* an initial seed set. Influence maximization asks the intervention-design question:

If we can choose only a few initial adopters, which ones should we pick?

**Problem statement.** Suppose a platform (alternatively a “campaign” or a “development economist”) can “seed” at most  $k$  nodes.

Under a diffusion model such as independent cascade or linear threshold, let

$$\sigma(S) = \text{expected total number of adopters when the seed set is } S.$$

Then the optimization problem is

$$\max_{S \subseteq V: |S| \leq k} \sigma(S),$$

in other words, we want to choose seeds to maximize the expected spread of adoption.

**Why this is not just choosing high-degree nodes?** High-degree nodes are often useful, but overlap also matters, in two competing ways:

- Seeding nodes in different communities can reach more of the graph.
- However, seeding multiple nodes in the same community, under the linear threshold model, can trigger a cascade within that community, which may be more effective than seeding a single node in multiple communities.

And so your model of adoption matters!

**Toy example.** Suppose we have the independent cascade model, and we can pick  $k = 2$  seeds. If nodes  $a$  and  $b$  both sit in the same dense community, then seeding both may only slightly outperform seeding one of them. But if node  $c$  is a bridge into a second community, then the pair  $\{a, c\}$  may outperform  $\{a, b\}$  even if  $b$  has higher degree than  $c$ . In Figure 7, choosing  $\{a, b\}$  concentrates both seeds in the left cluster, while choosing  $\{a, c\}$  can reach both clusters.

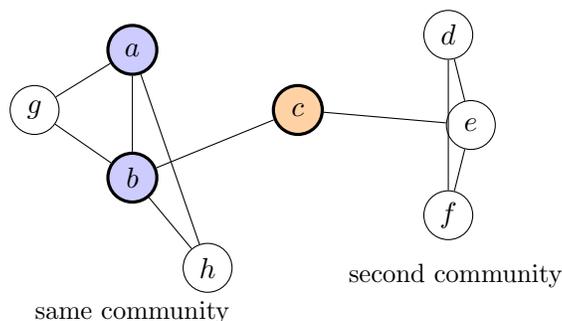


Figure 7: A toy seed-selection network. Node  $b$  has more local neighbors than  $c$ , but  $c$  is the bridge to the second community.

**Why exact optimization is hard.** There are exponentially many possible seed sets, and even evaluating expected spread can be computationally expensive. So in large networks we usually do not solve the problem exactly.

**Greedy algorithm.** A natural heuristic is:

1. Start with  $S = \emptyset$ .
2. Repeatedly add the node  $v$  with the largest marginal gain

$$\sigma(S \cup \{v\}) - \sigma(S).$$

3. Stop after selecting  $k$  seeds.

**“Submodularity”** Under standard versions of the independent cascade and linear threshold models,  $\sigma(S)$  is *monotone* (more seeds do not reduce spread) and *submodular* (diminishing marginal returns). Submodularity means that adding a new seed is typically more valuable when we have chosen few seeds than when we have already seeded many nearby nodes.

This diminishing-returns structure is powerful: the greedy algorithm is guaranteed to achieve at least a  $(1 - 1/e) \approx 63\%$  fraction of the optimal spread. So even though the exact problem is hard, we still get a strong approximation guarantee. This classic guarantee comes from Kempe, Kleinberg, and Tardos, *Maximizing the Spread of Influence through a Social Network* (journal paper based on their original KDD 2003 and ICALP 2005 papers).

**Practical considerations.** Real applications raise additional issues:

- **Model uncertainty:** the true diffusion process may differ from the assumed model.
- **Fairness:** maximizing total reach may ignore which communities receive attention.

In general, learning the network and diffusion model is a substantial challenge to actually doing effective influence maximization, and it's unclear to me (Nikhil) how much of the theoretical literature on influence maximization is actually useful in practice, given these challenges.

Akbarpour, Malladi, and Saberi (*American Economic Review*, November 2025), *Just a Few Seeds More*, study when detailed network data is actually valuable for seed selection. Their main message is that optimal optimization may not matter under certain models of uncertainty: just adding a few more random seeds can often achieve near-optimal spread, compared to optimal seeding with a few fewer seeds.

## Real-world examples (papers)

**Large-scale social influence experiment.** Bond et al. (*Nature*, 2012), *A 61-million-person experiment in social influence and political mobilization* (doi:10.1038/nature11421), ran a 61-million-person experiment on Facebook during the U.S. congressional elections. Users shown a social voting message (with friends' faces) were more likely to vote than users not shown that social signal, and effects propagated through friendship ties.

**Complex contagion in behavior adoption.** Centola (*Science*, 2010), *The Spread of Behavior in an Online Social Network Experiment* (doi:10.1126/science.1185231), used a controlled online health community and found that behavior spread faster in clustered-lattice networks than in random networks.

## 7 Learning graphs in practice

Many of the models above assume that the graph  $G$  is already known. In real applications, that graph is often only partially observed, and researchers have to *measure* or *infer* it from data. Different data sources produce different graphs, and the right one depends on the question being asked.

**Explicit platform ties.** Sometimes the graph is directly recorded by a platform through friendship links, follow links, or contact lists. For example, Ugander, Karrer, Backstrom, and Marlow (2011), *The Anatomy of the Facebook Social Graph*, study Facebook's declared friendship network. Even this, however, yields important choices: are we friends just because we are facebook friends, or is some level of interaction needed?

**Survey-elicited networks.** In offline settings, researchers often ask people whom they talk to, borrow from, seek advice from, or visit socially, and then aggregate those nominations into a graph. Banerjee, Chandrasekhar, Duflo, and Jackson (2013), *The Diffusion of Microfinance*, construct detailed village social networks in India this way. This approach is common in development economics and field experiments, but it can be expensive and sensitive to survey design. Banerjee and Duflo received a Nobel Prize in Economics in part for this work.

**Communication and administrative traces.** Another approach is to use meta data from surveillance of e-mail, phone, or messaging. Kossinets and Watts (2006), *Empirical Analysis of an Evolving Social Network*, recover a dynamic university interaction network from time-stamped e-mail headers. Similarly, Onnela et al. (2007), *Structure and Tie Strengths in Mobile Communication Networks*, build a large-scale call graph from mobile-phone records.

**Proximity and behavioral inference.** Sometimes ties are inferred from co-location, Bluetooth proximity, or mobility traces. Eagle, Pentland, and Lazer (2009), *Inferring Friendship Network Structure by Using Mobile Phone Data*, show that mobile-phone proximity and calling data can recover much of the underlying friendship network.

**Practical concerns.** Obviously there are huge privacy and ethical concerns with the above! It's not clear to me (Nikhil) how effective some of these approaches are, given the privacy concerns.