# Congestion Games

Aaron Roth

University of Pennsylvania

January 25 2024

# Overview

- Today we'll give study a structured class of large games.

# Overview

- Today we'll give study a structured class of large games.
- We'll study a simple, natural dynamic, and show it converges to Nash equilibrium.

# Overview

- Today we'll give study a structured class of large games.
- We'll study a simple, natural dynamic, and show it converges to Nash equilibrium.
- Our first "computationally plausible" set of predictions in a large interaction.

# Large Games

**Q:** How many numbers do we need to write down to represent an $n$ player $k$ action game?

# Large Games

**Q:** How many numbers do we need to write down to represent an $n$ player $k$ action game?
**A:** We need $k^n$ numbers just to encode a single utility function.

# Large Games

**Q:** How many numbers do we need to write down to represent an $n$ player $k$ action game?
**A:** We need $k^n$ numbers just to encode a single utility function.

Unreasonable to expect anyone to understand such an object.

# Large Games

**Q:** How many numbers do we need to write down to represent an $n$ player $k$ action game?
**A:** We need $k^n$ numbers just to encode a single utility function.

Unreasonable to expect anyone to understand such an object.
So: we need to think about structured, concisely defined games.

# Example 1: Traffic Routing

# Congestion Games

Convention: Players have *cost functions* they want to minimize rather than *utility functions* they want to maximize.

# Congestion Games

Convention: Players have *cost functions* they want to minimize rather than *utility functions* they want to maximize.

## Definition

A congestion game is defined by:

1. A set of $n$ players $P$

# Congestion Games

Convention: Players have *cost functions* they want to minimize rather than *utility functions* they want to maximize.

## Definition

A congestion game is defined by:

1. A set of $n$ players $P$
2. A set of $m$ facilities $F$

# Congestion Games

Convention: Players have *cost functions* they want to minimize rather than *utility functions* they want to maximize.

## Definition

A congestion game is defined by:

1. A set of $n$ players $P$
2. A set of $m$ facilities $F$
3. For each player $i$, a set of actions $A_i$. Each action $a_i \in A_i$ represents a subset of the facilities: $a_i \subseteq F$.

# Congestion Games

Convention: Players have *cost functions* they want to minimize rather than *utility functions* they want to maximize.

## Definition

A congestion game is defined by:

1. A set of $n$ players $P$
2. A set of $m$ facilities $F$
3. For each player $i$, a set of actions $A_i$. Each action $a_i \in A_i$ represents a subset of the facilities: $a_i \subseteq F$.
4. For each facility $j \in F$, a cost function $\ell_j : \{0, \ldots, n\} \to \mathbb{R}_{\geq 0}$. $\ell_j(k)$ represents "the cost of facility $j$ when $k$ players are using it".

# Congestion Games

Convention: Players have *cost functions* they want to minimize rather than *utility functions* they want to maximize.

## Definition

A congestion game is defined by:

1. A set of $n$ players $P$
2. A set of $m$ facilities $F$
3. For each player $i$, a set of actions $A_i$. Each action $a_i \in A_i$ represents a subset of the facilities: $a_i \subseteq F$.
4. For each facility $j \in F$, a cost function $\ell_j : \{0, \ldots, n\} \to \mathbb{R}_{\geq 0}$. $\ell_j(k)$ represents "the cost of facility $j$ when $k$ players are using it".

Player costs are then defined as follows. For action profile $a = (a_1, \ldots, a_n)$ define $n_j(a) = |\{i : j \in a_i\}|$ to be the number of players using facility $j$. Then the cost of agent $i$ is:

$$c_i(a) = \sum_{j \in a_i} \ell_j(n_j(a))$$

# Example 2: Network Creation

# Congestion games

- Ok, so we can concisely describe a large game...

# Congestion games

- Ok, so we can concisely describe a large game...
- So what? What can we do with it?

# Congestion games

- Ok, so we can concisely describe a large game...
- So what? What can we do with it?
- Do they have pure strategy Nash equilibria?

# Congestion games

- Ok, so we can concisely describe a large game...
- So what? What can we do with it?
- Do they have pure strategy Nash equilibria?
- Can computationally bounded, uncoordinated players find one?

# Congestion games

- Ok, so we can concisely describe a large game...
- So what? What can we do with it?
- Do they have pure strategy Nash equilibria?
- Can computationally bounded, uncoordinated players find one?
- i.e. are pure strategy Nash equilibria computationally plausible predictions?

# Congestion games

- Ok, so we can concisely describe a large game...
- So what? What can we do with it?
- Do they have pure strategy Nash equilibria?
- Can computationally bounded, uncoordinated players find one?
- i.e. are pure strategy Nash equilibria computationally plausible predictions?
- Lets study a simple dynamic...

# Best (Better) Response Dynamics

The basic idea:

# Best (Better) Response Dynamics

The basic idea:

1. Players start playing *arbitrary* actions.

# Best (Better) Response Dynamics

The basic idea:

1. Players start playing *arbitrary* actions.
2. In arbitrary order, players take turns changing their action if doing so can improve their utility.

# Best (Better) Response Dynamics

The basic idea:

1. Players start playing *arbitrary* actions.
2. In arbitrary order, players take turns changing their action if doing so can improve their utility.
3. Forever...

# Best (Better) Response Dynamics

**Algorithm 1** Best Response Dynamics

**Initialize** $a = (a_1, \ldots, a_n)$ to be an arbitrary action profile.
**while** There exists $i$ such that $a_i \notin \arg\min_{a \in A_i} c_i(a, a_{-i})$ **do**
  **Let** $a'_i$ be such that $c_i(a'_i, a_{-i}) < c(a)$.
  **Set** $a_i = a'_i$.
**end while**
**Halt** and return $a$.

# Best (Better) Response Dynamics

**Algorithm 2** Best Response Dynamics

**Initialize** $a = (a_1, \ldots, a_n)$ to be an arbitrary action profile.
**while** There exists $i$ such that $a_i \notin \arg\min_{a \in A_i} c_i(a, a_{-i})$ **do**
  **Let** $a'_i$ be such that $c_i(a'_i, a_{-i}) < c(a)$.
  **Set** $a_i = a'_i$.
**end while**
**Halt** and return $a$.

## Claim

*If best response dynamics halts, it returns a pure strategy Nash equilibrium.*

# Best (Better) Response Dynamics

**Algorithm 3** Best Response Dynamics

**Initialize** $a = (a_1, \ldots, a_n)$ to be an arbitrary action profile.
**while** There exists $i$ such that $a_i \notin \arg\min_{a \in A_i} c_i(a, a_{-i})$ **do**
  **Let** $a_i'$ be such that $c_i(a_i', a_{-i}) < c(a)$.
  **Set** $a_i = a_i'$.
**end while**
**Halt** and return $a$.

## Claim

*If best response dynamics halts, it returns a pure strategy Nash equilibrium.*

## Proof.

Immediate from halting condition – by definition, every player must be playing a best response. □

# Best (Better) Response Dynamics

Does best response dynamics always halt?

# Best (Better) Response Dynamics

Does best response dynamics always halt?

No: Consider matching pennies/Rock Paper Scissors.

# Best (Better) Response Dynamics

Does best response dynamics always halt?
No: Consider matching pennies/Rock Paper Scissors.

**Theorem**

*Best response dynamics always halt in congestion games.*

# Best (Better) Response Dynamics

Does best response dynamics always halt?
No: Consider matching pennies/Rock Paper Scissors.

### Theorem

*Best response dynamics always halt in congestion games.*

### Corollary

*All congestion games have at least one pure strategy Nash equilibrium.*

# Analysis of BRD in Congestion Games

1. Consider the *potential function* $\phi : A \to \mathbb{R}$:

$$\phi(a) = \sum_{j=1}^{m} \sum_{k=1}^{n_j(a)} \ell_j(k)$$

(Note: *not* social welfare)

# Analysis of BRD in Congestion Games

1. Consider the *potential function* $\phi : A \to \mathbb{R}$:

$$\phi(a) = \sum_{j=1}^{m} \sum_{k=1}^{n_j(a)} \ell_j(k)$$

   (Note: *not* social welfare)

2. How does $\phi$ change in one round of BRD? Say $i$ switches from $a_i$ to $b_i \in A_i$.

# Analysis of BRD in Congestion Games

1. Consider the *potential function* $\phi : A \to \mathbb{R}$:

$$\phi(a) = \sum_{j=1}^{m} \sum_{k=1}^{n_j(a)} \ell_j(k)$$

   (Note: *not* social welfare)

2. How does $\phi$ change in one round of BRD? Say $i$ switches from $a_i$ to $b_i \in A_i$.
3. Well... We know it must have decreased player $i$'s cost:

$$\begin{aligned}
\Delta c_i &\equiv c_i(b_i, a_{-i}) - c_i(a_i, a_{-i}) \\
&= \sum_{j \in b_i \setminus a_i} \ell_j(n_j(a) + 1) - \sum_{j \in a_i \setminus b_i} \ell_j(n_j(s)) \\
&< 0
\end{aligned}$$

# Analysis of BRD in Congestion Games

$$\phi(a) = \sum_{j=1}^{m} \sum_{k=1}^{n_j(a)} \ell_j(k)$$

# Analysis of BRD in Congestion Games

$$\phi(a) = \sum_{j=1}^{m} \sum_{k=1}^{n_j(a)} \ell_j(k)$$

1. The change in potential is:

$$\begin{aligned}
\Delta\phi &\equiv \phi(b_i, a_{-i}) - \phi(a_i, a_{-i}) \\
&= \sum_{j \in b_i \setminus a_i} \ell_j(n_j(a) + 1) - \sum_{j \in a_i \setminus b_i} \ell_j(n_j(s)) \\
&= \Delta c_i
\end{aligned}$$

# Analysis of BRD in Congestion Games

$$\phi(a) = \sum_{j=1}^{m} \sum_{k=1}^{n_j(a)} \ell_j(k)$$

1. The change in potential is:

$$\begin{aligned} \Delta\phi &\equiv \phi(b_i, a_{-i}) - \phi(a_i, a_{-i}) \\ &= \sum_{j \in b_i \setminus a_i} \ell_j(n_j(a) + 1) - \sum_{j \in a_i \setminus b_i} \ell_j(n_j(s)) \\ &= \Delta c_i \end{aligned}$$

2. Therefore, the change in potential is strictly *negative*

# Analysis of BRD in Congestion Games

$$\phi(a) = \sum_{j=1}^{m} \sum_{k=1}^{n_j(a)} \ell_j(k)$$

1. The change in potential is:

$$\begin{aligned} \Delta\phi &\equiv \phi(b_i, a_{-i}) - \phi(a_i, a_{-i}) \\ &= \sum_{j \in b_i \setminus a_i} \ell_j(n_j(a) + 1) - \sum_{j \in a_i \setminus b_i} \ell_j(n_j(s)) \\ &= \Delta c_i \end{aligned}$$

2. Therefore, the change in potential is strictly *negative*
3. So... since $\phi$ can take on only finitely many values, this cannot go on forever.

# Analysis of BRD in Congestion Games

$$\phi(a) = \sum_{j=1}^{m} \sum_{k=1}^{n_j(a)} \ell_j(k)$$

1. The change in potential is:

$$\begin{aligned} \Delta\phi &\equiv \phi(b_i, a_{-i}) - \phi(a_i, a_{-i}) \\ &= \sum_{j \in b_i \setminus a_i} \ell_j(n_j(a)+1) - \sum_{j \in a_i \setminus b_i} \ell_j(n_j(s)) \\ &= \Delta c_i \end{aligned}$$

2. Therefore, the change in potential is strictly *negative*
3. So... since $\phi$ can take on only finitely many values, this cannot go on forever.
4. And hence BRD halts in congestion games...

# Analysis of BRD in Congestion Games

$$\phi(a) = \sum_{j=1}^{m} \sum_{k=1}^{n_j(a)} \ell_j(k)$$

1. The change in potential is:

$$\begin{aligned} \Delta\phi &\equiv \phi(b_i, a_{-i}) - \phi(a_i, a_{-i}) \\ &= \sum_{j \in b_i \setminus a_i} \ell_j(n_j(a)+1) - \sum_{j \in a_i \setminus b_i} \ell_j(n_j(s)) \\ &= \Delta c_i \end{aligned}$$

2. Therefore, the change in potential is strictly *negative*
3. So... since $\phi$ can take on only finitely many values, this cannot go on forever.
4. And hence BRD halts in congestion games...
5. Which proves the *existence* of pure strategy Nash equilibria!

# Efficiency

*But... How long does it take?*

# Efficiency

*But... How long does it take?*

Our proof gives only an exponential convergence bound... And it might really take that long!

# Efficiency

*But... How long does it take?*

Our proof gives only an exponential convergence bound... And it might really take that long!

Lets consider approximation...

# Approximation

## Definition

An action profile $a \in A$ is an $\epsilon$-approximate pure strategy Nash equilibrium if for every player $i$, and for every action $a'_i \in A_i$:

$$c_i(a_i, a_{-i}) \leq c_i(a'_i, a_{-i}) + \epsilon$$

i.e. nobody can gain more than $\epsilon$ by deviating.

.

# Approximate Best Response Dynamics

**Algorithm 4** FindApproxNash($\epsilon$)

**Initialize** $a = (a_1, \ldots, a_n)$ to be an arbitrary action profile.
**while** There exists $i, a'_i$ such that $c_i(a'_i, a_{-i}) \leq c_i(a_i, a_{-i}) - \epsilon$ **do**
  **Set** $a_i = \arg\min_{a \in A_i} c_i(a, a_{-i})$
**end while**
**Halt** and return $a$.

# Approximate Best Response Dynamics

**Algorithm 5** FindApproxNash($\epsilon$)

**Initialize** $a = (a_1, \ldots, a_n)$ to be an arbitrary action profile.
**while** There exists $i, a_i'$ such that $c_i(a_i', a_{-i}) \leq c_i(a_i, a_{-i}) - \epsilon$ **do**
  **Set** $a_i = \arg\min_{a \in A_i} c_i(a, a_{-i})$
**end while**
**Halt** and return $a$.

## Claim

*If FindApproxNash($\epsilon$) halts, it returns an $\epsilon$-approximate pure strategy Nash equilibrium*

# Approximate Best Response Dynamics

**Algorithm 6** FindApproxNash($\epsilon$)

**Initialize** $a = (a_1, \ldots, a_n)$ to be an arbitrary action profile.
**while** There exists $i, a'_i$ such that $c_i(a'_i, a_{-i}) \leq c_i(a_i, a_{-i}) - \epsilon$ **do**
  **Set** $a_i = \arg\min_{a \in A_i} c_i(a, a_{-i})$
**end while**
**Halt** and return $a$.

## Claim

*If FindApproxNash($\epsilon$) halts, it returns an $\epsilon$-approximate pure strategy Nash equilibrium*

## Proof.

Immediately, by definition. □

# Analysis

## Theorem

*In any congestion game, FindApproxNash($\epsilon$) halts after at most:*

$$\frac{n \cdot m \cdot c_{max}}{\epsilon}$$

*steps, where* $c_{max} = \max_{j,k} \ell_j(k)$ *is the maximum facility cost.*

# Analysis

## Theorem

*In any congestion game, FindApproxNash($\epsilon$) halts after at most:*

$$\frac{n \cdot m \cdot c_{max}}{\epsilon}$$

*steps, where $c_{max} = \max_{j,k} \ell_j(k)$ is the maximum facility cost.*

## Proof.

We revisit the potential function $\phi$. Recall that $\Delta c_i = \Delta \phi$ on any round when player $i$ moves.

# Analysis

## Theorem

*In any congestion game, FindApproxNash($\epsilon$) halts after at most:*

$$\frac{n \cdot m \cdot c_{max}}{\epsilon}$$

*steps, where $c_{max} = \max_{j,k} \ell_j(k)$ is the maximum facility cost.*

## Proof.

We revisit the potential function $\phi$. Recall that $\Delta c_i = \Delta\phi$ on any round when player $i$ moves.

Observe also that at every round, $\phi \geq 0$, and

$$\phi(a) = \sum_{j=1}^{m} \sum_{k=1}^{n_j(a)} \ell_j(k) \leq n \cdot m \cdot c_{max}$$

# Analysis

## Theorem

*In any congestion game, FindApproxNash($\epsilon$) halts after at most:*

$$\frac{n \cdot m \cdot c_{max}}{\epsilon}$$

*steps, where $c_{max} = \max_{j,k} \ell_j(k)$ is the maximum facility cost.*

## Proof.

We revisit the potential function $\phi$. Recall that $\Delta c_i = \Delta \phi$ on any round when player $i$ moves.
Observe also that at every round, $\phi \geq 0$, and

$$\phi(a) = \sum_{j=1}^{m} \sum_{k=1}^{n_j(a)} \ell_j(k) \leq n \cdot m \cdot c_{max}$$

By definition of the algorithm, we have $\Delta c_i = \Delta \phi \leq -\epsilon$ at every round, and so the theorem follows. □

# Thanks!

See you next class — stay healthy!