17.810/17.811 – Game Theory

Lecture 9: Additional Topics

Asya Magazinnik

MIT

Bounded Rationality

Algorithmic Game Theory

Evolutionary Game Theory

Agent-Based Modeling

Bounded Rationality

Boundedly rational players have limited:

- computational power
- foresight
- knowledge of the game they're playing
- willingness to play complicated strategies...

A literature at the intersection of economics and psychology incorporates limitations of human cognition into strategic games (Rubinstein 1998).

Example: Modeling Limited Foresight

Consider a stage game $\mathcal{G}$ that is repeated J times, but players can only see $K < J$ stages ahead. One approach to equilibrium:

Definition (Limited Foresight Equilibrium)

An equilibrium is a profile of strategies $(f_i^*)_{i \in N}$ such that for every history h after which player i has to move, f_i^* is some strategy that maximizes the sum of the payoffs in the K horizon given the other players' strategies.

Player i only takes the first step of this optimal strategy and recomputes at every turn.

Example: Strategies as Machines

The goal is to reduce complex decisionmaking to simple heuristics.

Definition (Machine strategies)

A machine for player i in an infinitely repeated game $\mathcal{G}$ is a four-tuple:

- Q_i is a finite set of states
- q_i^0 is the initial state
- $f_i(Q_i) = A_i$ is an output function that assigns an action to every state
- $\tau(Q_i, A_j)$ is a transition function that assigns a state to every pair of a current state and an action of the other player

Strategies as Machines: Grim Trigger

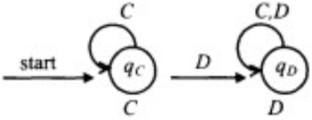
The grim trigger strategy can be conveyed as a machine:

$$Q_i = \{q_C, q_D\}$$

$$q_i^0 = q_C$$

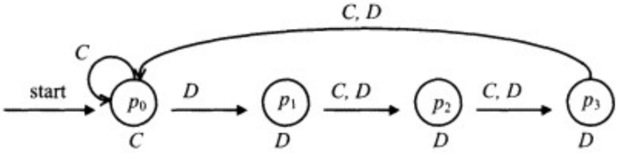
$$f_i(q_C) = C \text{ and } f_i(q_D) = D$$

$$\tau_i(q, a_j) = \begin{cases} q_C & (q, a_j) = (q_C, C) \\ q_D & \text{otherwise} \end{cases}$$



Strategies as Machines: Finite Punishment Strategies

The machine below corresponds to the strategy: play C as long as other player plays C; if other player defects, play D for three periods, then revert to C no matter what.



Further reading: Ariel Rubinstein, *Modeling Bounded Rationality* (1998).

Bounded Rationality

Algorithmic Game Theory

Evolutionary Game Theory

Agent-Based Modeling

Mechanism Design

Can we design mechanisms that get people to reveal their preferences and achieve good outcomes, taking into account their strategic incentives?

An extraordinarily rich and influential literature has created mechanisms for:

- matching buyers to products (e.g. auctions)
- matching organ donors to organ recipients
- matching job applicants to jobs (e.g. medical residents to hospitals)

These “mechanisms” are often relatively simple algorithms.

The Top Trading Cycle (TTC) Algorithm

Suppose we have n individuals, each assigned to a house; agents need not prefer their own house to all others.

Top Trading Cycle (TTC) Algorithm

initialize N to the set of all agents

while $N \neq \emptyset$ **do**

form the directed graph G with vertex set N and edge set $\{(i, \ell)$:
i's favorite house within N is owned by $\ell\}$

compute the directed cycles $C_1, \ldots, C_h$ of G[3]

```
// self-loops count as directed cycles
// cycles are disjoint
```

for each edge (i, ℓ) of each cycle $C_1, \ldots, C_h$ **do**

reallocate ℓ's house to agent i

remove the agents of $C_1, \ldots, C_h$ from N

The Top Trading Cycle (TTC) Algorithm

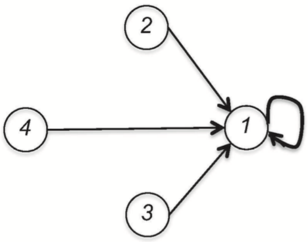
Preference orderings are given by:

Agent 1: 1, 2, 3, 4

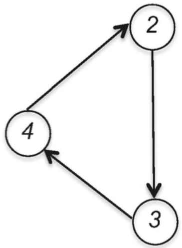
Agent 2: 1, 3, 4, 2

Agent 3: 1, 4, 3, 2

Agent 4: 1, 2, 3, 4



(a) First iteration



(b) Second iteration

What are desirable properties of an allocation mechanism?

The TTC algorithm has the desirable property that it is Dominant-Strategy Incentive Compatible (DSIC).

Definition (Dominant-Strategy Incentive Compatible)

A mechanism is *DSIC* if truthful behavior is always a dominant strategy for every player and if truthful players always obtain nonnegative utility.

Having no incentive to "game the system" — to be rewarded for revealing your true preferences — seems a desirable property. But a mechanism that never reassigns anything is also *DSIC*!

What are desirable properties of an allocation mechanism?

In addition to creating incentives for truthful revelation, we want to design mechanisms that capture all gains from trade.

Definition (Blocking coalition)

Consider an assignment of one distinct house to each agent. A subset of agents forms a blocking coalition for this assignment if they can internally reallocate their original houses to make some member better off while making no member worse off.

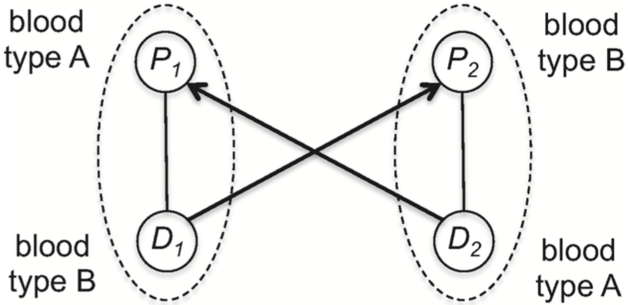
Definition (Core allocation)

A core allocation is an assignment with no blocking coalitions.

TTC is optimal in the sense that it is both **DSIC** and that it produces the unique **core allocation**.

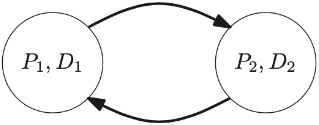
Stable Matching and Kidney Exchange

The problem: matching kidney donors to patients.

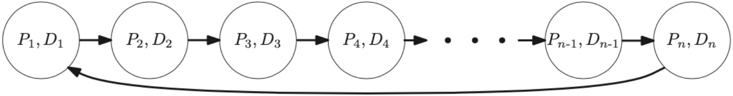


Can we apply the TTC algorithm here? Maybe.

A good case for the TTC algorithm:



A bad case for the TTC algorithm:

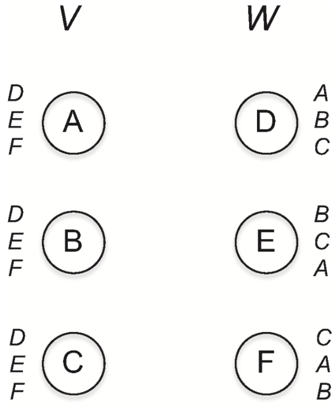


Stable Matching and Kidney Exchange

Let M be a perfect matching assigning each $v \in V$ to one $w \in W$.

$v \in V$ and $w \in W$ form a blocking pair if they are not matched in M, v prefers w to her match in M, and w prefers v to her match in M.

A perfect matching is stable if it has no blocking pairs.



The Deferred Acceptance Algorithm

Deferred Acceptance Algorithm

while there is an unmatched applicant $v \in V$ **do**
  v attempts to match with her favorite hospital w who has not rejected her yet
  if w is unmatched **then**
    v and w are tentatively matched
  else if w is tentatively matched to v' **then**
    w rejects whomever of v, v' it likes less and is tentatively matched to the other one
all tentative matches are made final

The Deferred Acceptance Algorithm

The Deferred Acceptance Algorithm has remarkable properties:

1. **Existence of a Stable Matching:** *For every collection of preference lists for the applicants and hospitals, there exists at least one stable matching.*
2. **Fast Computation of a Stable Matching:** *The algorithm completes with a stable matching after at most n^2 iterations, where n is the number of vertices on each side.*
3. **Applicant-Optimality:** *While there are potentially numerous stable matchings, the deferred acceptance algorithm matches every applicant to her most preferred option of all stable matchings.*
4. **Truthful Reporting for the Applicant:** *The applicant (v) is never strictly better off reporting falsely than truthfully, though the hospital (w) might be.*

Further Reading

Roger B. Myerson, "Perspectives on Mechanism Design in Economic Theory," *The American Economic Review* 98(3), 2008.

Tim Roughgarden, *Twenty Lectures on Algorithmic Game Theory*, New York: Cambridge University Press, 2016.

Bounded Rationality

Algorithmic Game Theory

Evolutionary Game Theory

Agent-Based Modeling

Evolutionary Game Theory

Evolutionary game theory focuses on populations that repeatedly play games rather than individuals.

Consider a symmetric two-player game G in normal form with a finite strategy set $S = \{s_1, ..., s_n\}$.

An individual of type τ uses a mixed strategy $\tau = \sum_i q_i s_i$, with all $q_i \geq 0$ and $\sum_i q_i = 1$.

A population as a whole uses strategy s_i with probability p_i; the population state is $\sigma = \sum_i p_i s_i$.

Thus an individual's expected payoff from playing strategy s_i against a population of type σ is given by:

$$u(s_i, \sigma) = \sum_{j=1}^{n} u(s_i, s_j) p_j$$

Evolutionary Stable Populations

Now suppose, in a population with state σ, we replace a fraction ϵ of the population with individuals of type τ. Then the new population state is given by:

$$(1-\epsilon)\sigma + \epsilon\tau$$

We say that a population state is evolutionarily stable if for every $\tau \neq \sigma$ there is a number $\epsilon_0 > 0$ such that if $0 < \epsilon < \epsilon_0$ then:

$$u(\sigma, (1-\epsilon)\sigma + \epsilon\tau) > u(\tau, (1-\epsilon)\sigma + \epsilon\tau)$$

This means that if a population of type σ is invaded by a small number of individuals of any other type τ, then individuals of type σ will have a **better payoff** against a random member of the mixed population than individuals of type τ.

Evolutionary Stable Populations

Theorem

If (σ, σ) is a pure strategy Nash equilibrium of a symmetric two-player game, then σ is an evolutionarily stable state of the corresponding evolutionary game.

Example: Stag Hunt

Stag Hunt		
1 / 2	Stag	Hare
Stag	2,2	0,1
Hare	1,0	1,1

There are two symmetric pure strategy equilibria, **Stag, Stag** and **Hare, Hare**. Accordingly, both pure populations of stag hunters and hare hunters are evolutionarily stable. (Why?)

There is additionally a mixed strategy equilibrium of the game where both players play stag half the time and hare half the time, but this is not an evolutionarily stable population.

Evolutionary Dynamics

We can think about the rate of growth of a successful invader.

As before, let $\sigma = \sum_i p_i s_i$ for strategies $i = 1, ..., n$ be the population state, but let us now regard the population state as changing with time:

$$\sigma(t) = \sum_i p_i(t) s_i$$

It is reasonable to expect that if $u(s_i, \sigma) > u(\sigma, \sigma)$ then individuals playing s_i should have an above-average number of offspring, so $p_i(t+1) > p_i(t)$.

In fact if we assume that the rate of growth is proportional to $u(s_i, \sigma) - u(\sigma, \sigma)$ then we obtain the replicator system:

$$\dot{p}_i = (u(s_i, \sigma) - u(\sigma, \sigma)) p_i$$

The replicator system is a differential equation on $\mathbb{R}^n$.

Contributions of Evolutionary Game Theory

Thus evolutionary game theory helps us think about:

- How social norms are maintained and challenged over time; which norms are resilient and which are vulnerable
- How traits that do not appear individually rational might survive in populations (e.g. altruism)

Further Reading: Jorgen Weibull, *Evolutionary Game Theory* (1995)

Bounded Rationality

Algorithmic Game Theory

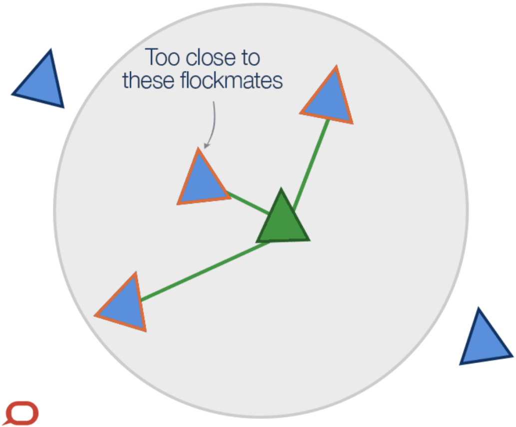
Evolutionary Game Theory

Agent-Based Modeling

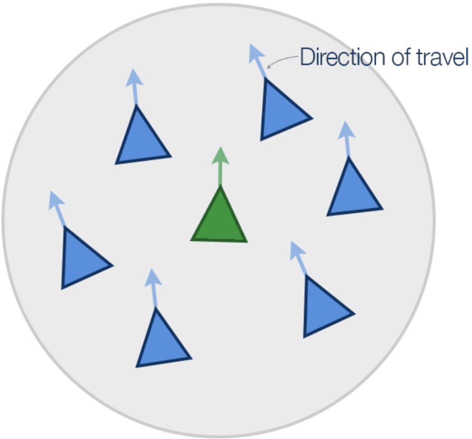
Some seemingly complex phenomena are simply aggregations of simple individual-level behavior. Take birds flocking.

A lifelike simulation can be built by following three simple rules.

Rule 1: Separation. Steer to avoid crowding local flockmates.

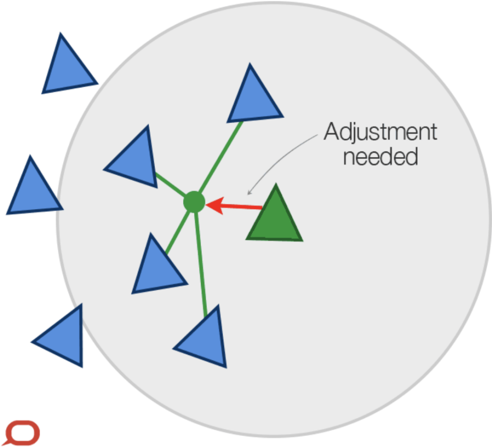


Rule 2: Alignment. Steer in the same direction of travel as local flockmates.



Agent-Based Modeling: Birds Flocking

Rule 3: Cohesion. Steer to the average position of local flockmates.



Agent-Based Modeling: Applications

If individual behavior approximately follows simple rules, one needs only to write down those rules and let computational systems do the work. Applications abound:

- Ecological systems (flight, migration, cooperation, communication, predation)
- Traffic congestion
- Disease transmission
- Financial markets
- Climate change

Further Reading: Uri Wilensky and William Rand, *An Introduction to Agent-Based Modeling: Modeling Natural, Social, and Engineered Complex Systems with NetLogo*, Cambridge: The MIT Press, 2015.

More Courses at MIT

Computer Science:

- **6.853** Topics in Algorithmic Game Theory

Philosophy:

- **24.222** Decisions, Games and Rational Choice

Political Science:

- **17.416** Theoretical Models in International Relations and Comparative Politics
- **17.905** Formal Approaches to American Political Institutions

Economics

- **14.18** Mathematical Economic Modeling

A huge thank you to Sean!

Stay in touch: asyam@mit.edu