15-780: Graduate AI
Computational Game Theory II

Michael Benisch

Carnegie Mellon University,
School of Computer Science

mbenisch@cs.cmu.edu

Recap: What is a game?

- Multi-agent model of incentives

- Multi-Objective Optimization vs. Game Theory
  - ❑ Agents own variables and objective functions

- Games are natural models of real-world processes

mbenisch@cs.cmu.edu

Recap: Key Concepts

- Normal-form game
  - ❑ Exhaustive representation with n matrices
  - ❑ Size is exponential in number of players
- Extensive form game
  - ❑ Represents sequential actions more compactly
  - ❑ Conversion to NFG can result in exponential blow-up
  - ❑ Can encode imperfect information (information sets) and random events (chance nodes)

mbenisch@cs.cmu.edu

Recap: Key Concepts

- **Mixed strategy**: play strategies in supports with probability > 0 (total mixture must sum to 1)

- **Best response**: strategy that provides highest expected utility given actions of other players
  - Every opponent profile has a pure strategy BR

- **Nash equilibrium**: profile of (potentially mixed) strategies such that no agent can unilaterally improve

mbenisch@cs.cmu.edu

Recap: Finding Nash Equilibria

- Theorem: Every NFG has at least one NE in potentially mixed strategies [Nash].
  - Knowing supports gives feasibility problem

- Algorithms for finding NE in 2-player games
  - Lemke-Howson (like Simplex)
  - Porter-Nudelman-Shoham (support-enumeration)
  - MIP Nash (mixed-integer program formulation)

- {0,1} 2-player Nash is PPAD-Complete

mbenisch@cs.cmu.edu

Recap: Criticisms of Nash equilibrium

- **Not necessarily unique**: some games have multiple NEs, which will agents settle into?
  - Social-welfare maximizing?
  - Pareto-optimal?

- **Can be hard to compute**

- NE is **not consistent**
  - One player can unilaterally move system from one equilibrium to another

mbenisch@cs.cmu.edu

Outline

- Solving games with AI
  - ❑ Alternative solution concepts
  - ❑ Learning games empirically
  - ❑ Compact forms
- Building games with AI
  - ❑ Mechanism design problem and Revelation Principle
  - ❑ Game theoretic properties of auctions: 1^{st} price, 2^{nd} price, eBay
  - ❑ Implementation in dominant strategies
  - ❑ Vickrey-Clarke-Groves Mechanism
  - ❑ Automated Mechanism Design

mbenisch@cs.cmu.edu

Solving Games with AI:
Alternative Solution Concepts

mbenisch@cs.cmu.edu

Iterated Dominance

- **Iterated Dominance**: iteratively remove strategies that are (weakly) dominated (games solvable this way are called "Dominance solvable")
  - **Strict ID:** path independent, **Weak ID:** path dependent

	L	C	R
U	3, 1	0, 1	0, 0
M	1, 1	1, 1	5, 0
D	0, 1	4, 1	0, 0

(Removal order marked on the matrix: ① R, ② M, ③ L, ④ U)

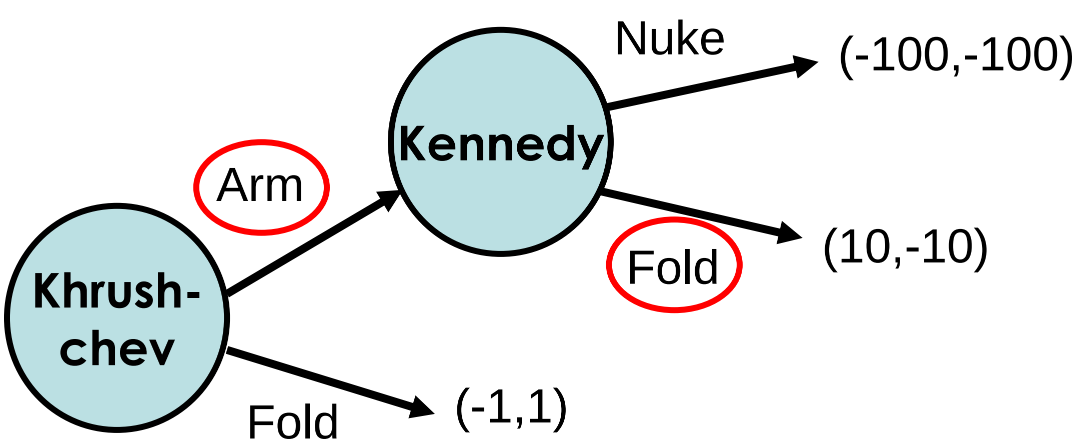
mbenisch@cs.cmu.edu

Rationalizability

- **Rationalizability:** rationality restricts players to playing only strategies that are best responses to rationalizable strategies *(note the recursive definition)
  - ❑ **In 2-player games**: set of rationalizable strategies equals set of strategies that survive iterated dominance
- Nash equilibrium strategies are rationalizable
  - ❑ We can preprocess games for NE search by removing non-rationalizable strategies

Sub-Game Perfect Equilibrium

- **Proper sub-game:** everything following an internal node in an extensive form game
- **Sub-game Perfect Equilibrium [Selten 72]:** EFG strategy profile in NE in every proper sub-game



Two Nash equilibria:
- Khru: Fold, Ken: Nuke
- Khru: Arm, Ken: Fold

One SGP equilibrium:
- Khru: Arm, Ken: Fold

Nuke is not a credible threat!

mbenisch@cs.cmu.edu

CURB Sets

(Closed Under Rational Behavior)

Basu & Weibull, '91 : "A mutually consistent set of player beliefs."

Def. *S* is CURB if it contains all rationalizable strategies with supports in *S*

Def. *S* is *minimally* CURB if for every subset *S′*, *S′* is not CURB

CURB set Facts:

- A full game is trivially CURB
- All games contain a minimal CURB set
- Not all strategies are in a minimal CURB set
- Pure Nash equilibrium = CURB set with one strategy per player

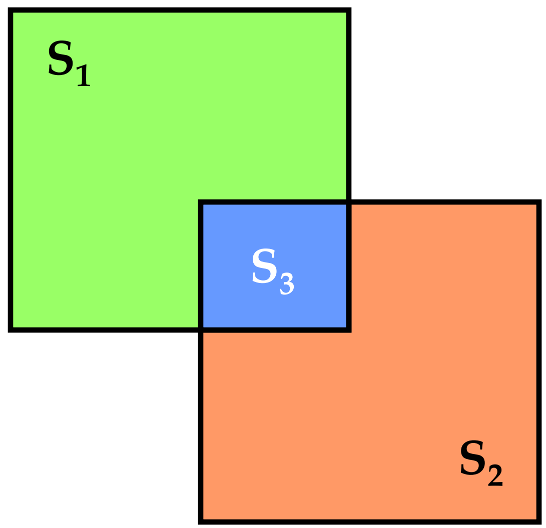
mbenisch@cs.cmu.edu

Example CURB Sets

(1,0)	(0,0)	(0,0)	(0,1)
(0,0)	(1,0)	(0,1)	(0,0)
(0,0)	(0,1)	(1,0)	(0,0)
(0,1)	(0,0)	(0,0)	(1,0)

Computing CURB Sets

- **Thrm**: If two CURB Sets share strategies, their intersection must also be CURB [Benisch et. al.]



- **Thrm**: all minimal CURB sets can be found in polynomial time [Benisch et. al.]

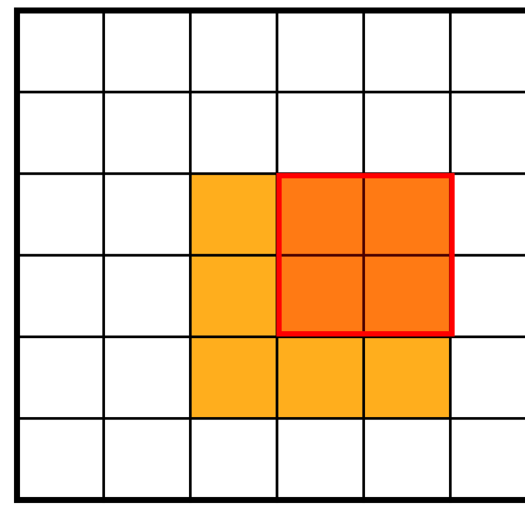
mbenisch@cs.cmu.edu

CURB sets and NE

Basu & Weibull, '91 : Every minimal CURB set contains supports of a mixed-strategy Nash equilibrium

- **Thrm:** The complexity of finding a Nash equilibrium in a 2-player game is super-polynomial *only* in the size of its smallest CURB set

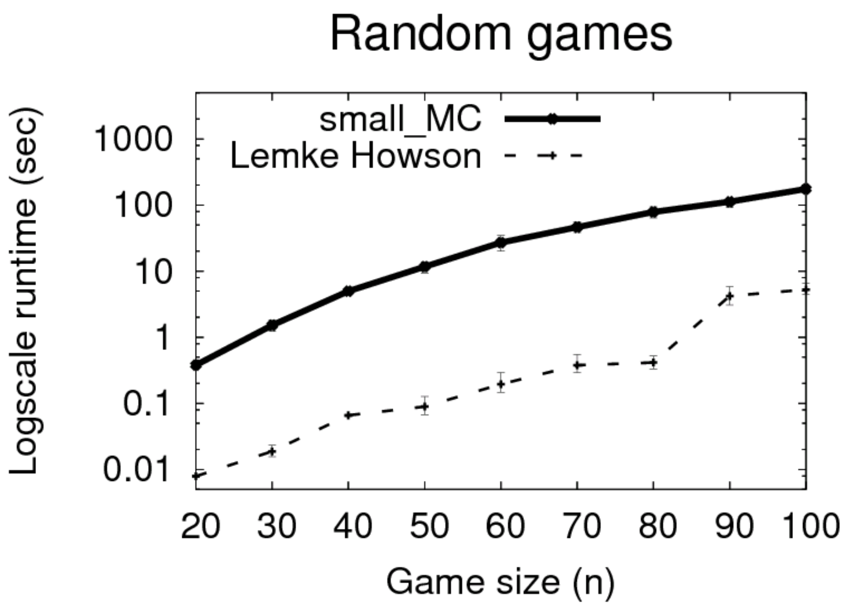
[Benisch et. al.].

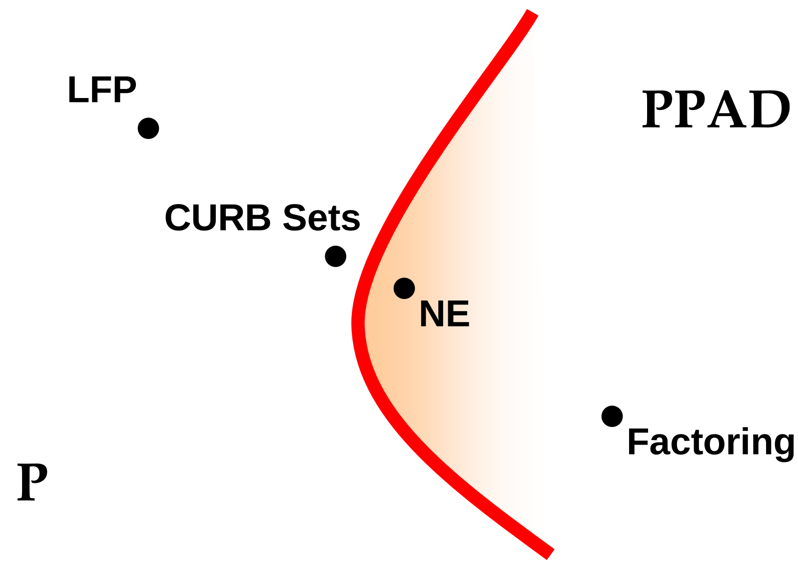


Minimal CURB sets capture all the complexity of finding an NE
(…but can lead to arbitrarily large or small savings in a particular instance)

mbenisch@cs.cmu.edu

CURB sets and NE (cont'd)



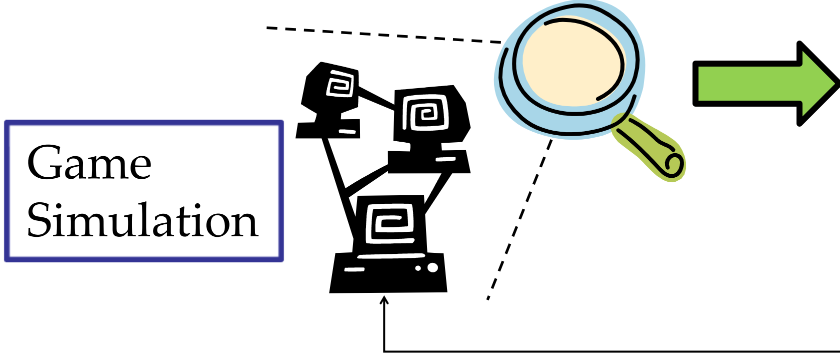


Empirical Game Theory

- We have many tools for analyzing games, but how can we apply them in real situations?

- Ask experts for models of real world games
  - **GAMUT**: library of economic game models

- Learn approximate games empirically [Wellman, et. al.]
  - Simulate or observe outcomes from strategy profiles



	C	D
c	(3,3)	(0,5)
d	(5,0)	(1,1)

Compact Forms

- **Compact Forms:** game representations that exploit structure in some games.

- Compact forms can ease reasoning about games
  - Typically allow exponentially faster algorithms for computation of NE
  - Occasionally allow better theoretical analysis

- Compact forms also reduce representation size
  - Typically exponential reduction in representation size

- Can we learn games in a compact form?

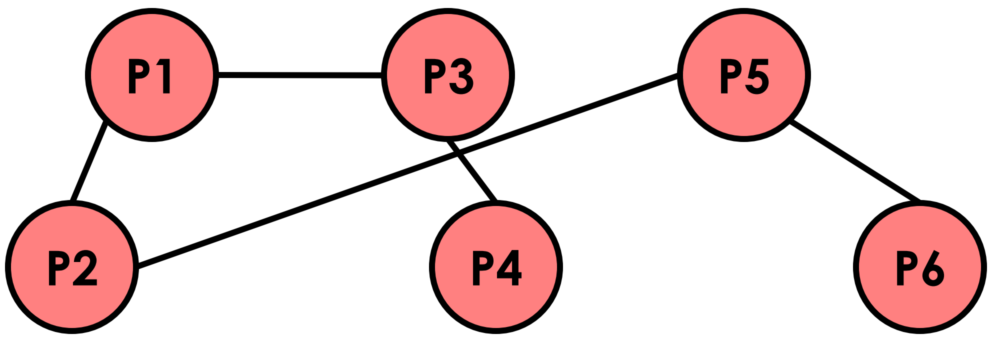
Compact Forms: Examples

- Some examples of compact forms:
  - **Extensive form games:** represent sequential structure
  - **Graphical games:** players represented by nodes in a graph can only influence neighbors [Kearns et. al.].
  - **Local effect games (LEGs):** actions effect agents who take other locally related actions (e.g. where to build a Starbucks) [Leyton-Brown et. al.]
  - **Game factors (new):** underlying games that are completely independent [Davis, Benisch, et.al.].
    - Normal form games can be "factored" in polynomial time.

mbenisch@cs.cmu.edu

Compact Forms: Examples

- **Graphical Games:** neighbors effect utilities



- **Factor games:** independent strategic interactions
  - **Direct flight game:** airlines pay cost to service direct flights and split resulting revenue

Building Games with AI: Mechanism Design

mbenisch@cs.cmu.edu

Social Choice Problems

- A group of (selfish) agents collectively choosing among outcomes solve a social choice problem
  - Auctions, voting, allocation of goods, tasks, resources
- Agents have preferences over outcomes
- Social choice function takes preferences as input and picks "optimal" outcome
- Will agents tell the truth about their preferences?

mbenisch@cs.cmu.edu

Non-Truthful Mechanisms

- Non-truthful mechanisms are used commonly, sometimes they have other properties (e.g. higher expected revenue when bidders are asymmetric)

- Example: 1st-price auction (equivalent to Dutch auction/descending auction) protocol,
  - Auctioneer has one item up for auction
  - Sealed bids are accepted
  - Highest bidder wins, pays bid price

- **Optimal strategy:** shaving, valuation – f(bidders)

mbenisch@cs.cmu.edu

Mechanism Design Problems

- Build a mechanism that properly implements a social choice function even when agents can lie

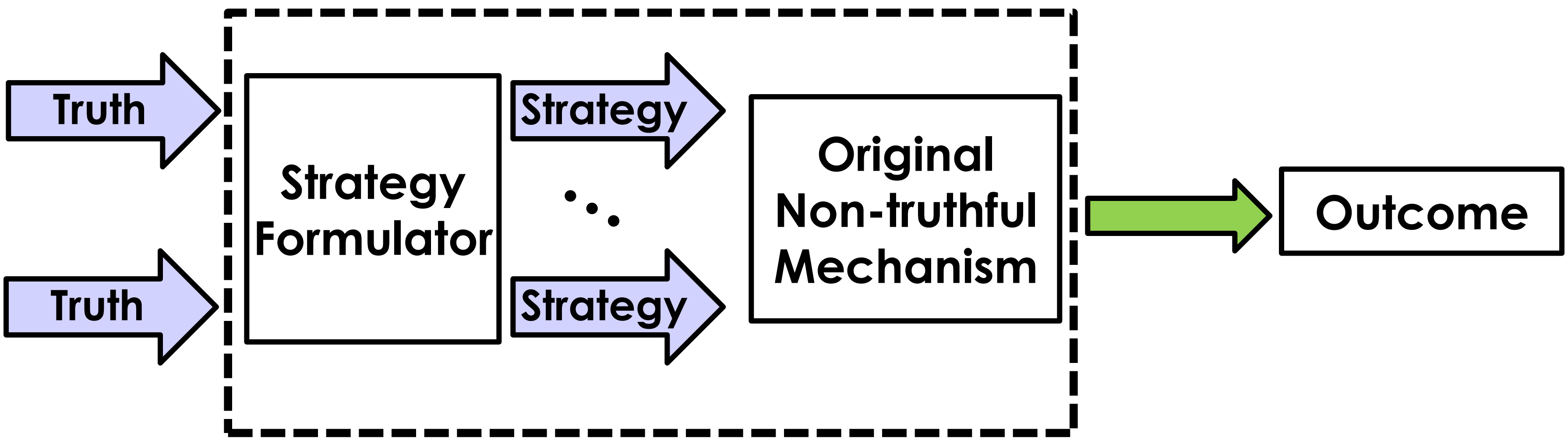
- Make lying against the best interest of the agents in various ways:
  - All non-truthful strategies are dominated
  - Only truthful strategies are NE
  - At least one truthful strategy is a NE

- Alternatively, mechanism can guarantee some property even if agents lie, but…

mbenisch@cs.cmu.edu

Revelation Principle

- Any outcome that can be enforced by a Nash (or dominant strategy) equilibrium in a *one shot* mechanism where agents lie can be enforced in Nash (dominant strategy) equilibrium in a mechanism where agents tell the truth



Dominance Implementation

- A mechanism is implementable in dominant strategies if truth-telling is (weakly) dominant

- Agents do not need to counter-speculate about each other, in particular they can ignore:

  - The preferences of other agents
  - The rationality of other agents
  - The capabilities of other agents
  - …

mbenisch@cs.cmu.edu

Gibbard-Satterthwaite Impossibility

- Consider the following three desiderata of a social choice function, *f*:
  - *f* is implementable in dominant strategies
  - Every outcome can be chosen by some input to *f*
  - *f* is not dictatorial (depends on the preferences of more than one agent)

- **Theorem:** with more than 3 outcomes no such *f* exists for all possible preferences.
  - **Proof:** based on Arrow's theorem for social choice.

mbenisch@cs.cmu.edu

Vickrey Auctions

- But we can still design dominant strategy mechanisms in some cases
  - When preferences of agents have special structure
  - When computing a good lie is hard

- Example: 2^{nd}-price auction (a.k.a. Vickrey auction, English/ascending auction) protocol,
  - Auctioneer has one item up for auction
  - Sealed bids are accepted
  - Highest bidder wins, pays 2^{nd} highest price

mbenisch@cs.cmu.edu

Vickrey Auctions

- Proof that 2^{nd}-price auction is truth-dominant:
  - Consider when bidder values good at $v and highest competing bid is $v′
  - If v < v′ then bidder can only win by bidding more than v′ and paying more than valuation, so she should bid truth
  - If v > v′ then bidder will only pay v′ on any bid between the two values and could lose by bidding less, so she can only be hurt by lying

mbenisch@cs.cmu.edu

Auction Mechanism

- Almost same as 2^{nd}-price auction
  - ❑ Proxy-bidding designed to act as strategy formulator from Revelation Principle

automatically bids on your behalf up to your maximum bid.

  - ❑ **Optimal bidding strategy**: bid truth to proxy at last second and nothing prior (sniping)

mbenisch@cs.cmu.edu

Vickrey-Clarke-Groves Mechanism

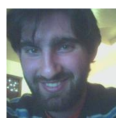
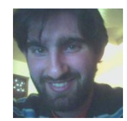
- Generalization of the 2nd-price auction for allocating multiple goods (e.g. pair of shoes)
  - Implements social welfare maximizing outcome for:
    - **quasi-linear preferences:** u(x + money) = v(x) + money
  - Agents submit all preferences and pay the difference between the total utility of other agents under the outcome selected with their bids considered and the outcome selected without their bids considered
  - Payment of agent *i* = $_{j \ i} v_j(s(v))$ - $_{j \ i} v_j(s(v_{-i}))$
    - *s*(*X*) = social welfare maximizing outcome given preferences *X*

VCG Example

Goods to be allocated:

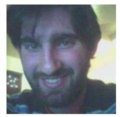
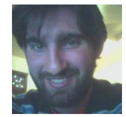
Agents:

Outcomes: / Preferences:

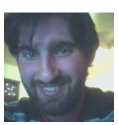
			
		275	0
		0	285
		250	50
		30	250

mbenisch@cs.cmu.edu

VCG Example

Outcomes:		Preferences:	
			
		275	0
		0	285
		250	50
		30	250

Payments:

 : 50 – 285 = –235

 : 250 – 275 = -25

 : 235 + 25 = +260

mbenisch@cs.cmu.edu

Automated Mechanism Design

- **Main idea:** search for values of payment function and allocation rule that satisfy certain constraints
  - ❑ **Incentive Compatibility:** no agents have an incentive to lie about private info in equilibrium
  - ❑ **Individually Rational:** agents are no worse off participating (in expectation) than not
- **Key Insight:** formulating this question as an optimization problem and using standard AI techniques to answer it [Conitzer and Sandholm].

mbenisch@cs.cmu.edu

Automated Mechanism Design

- **Instance inputs:**
  - Set of possible outcomes
  - Set of agents with distribution over preference profiles
  - Objective function maps outcomes to values (e.g. maximize social welfare)

- **Outputs:**
  - Mechanism that maps revealed preferences to outcomes and payments to and from each agent
  - Mechanism must satisfy IR and IC constraints
  - Mechanism must (nearly) maximize objective function in expectation

mbenisch@cs.cmu.edu