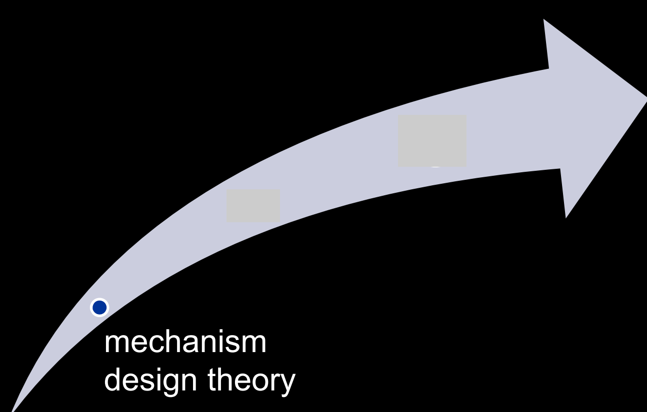
When Analysis Fails: Heuristic Mechanism Design via Self-Correcting Procedures

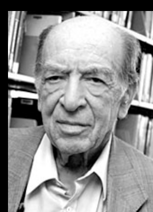
David C. Parkes
Harvard University

Špindlerův Mlýn SOFSEM'09



Mechanism design theory

- Leonid Hurwicz (1960, 1972)
  - communication system, incentive compatibility

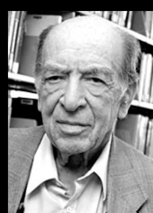


Mechanism design theory

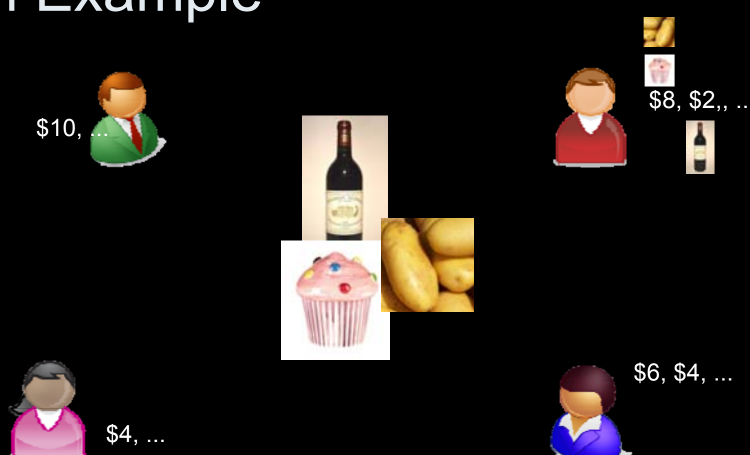
- Leonid Hurwicz (1960, 1972)
  - communication system, incentive compatibility
- Eric Maskin (1977)
  - Nash implementation


Mechanism design theory

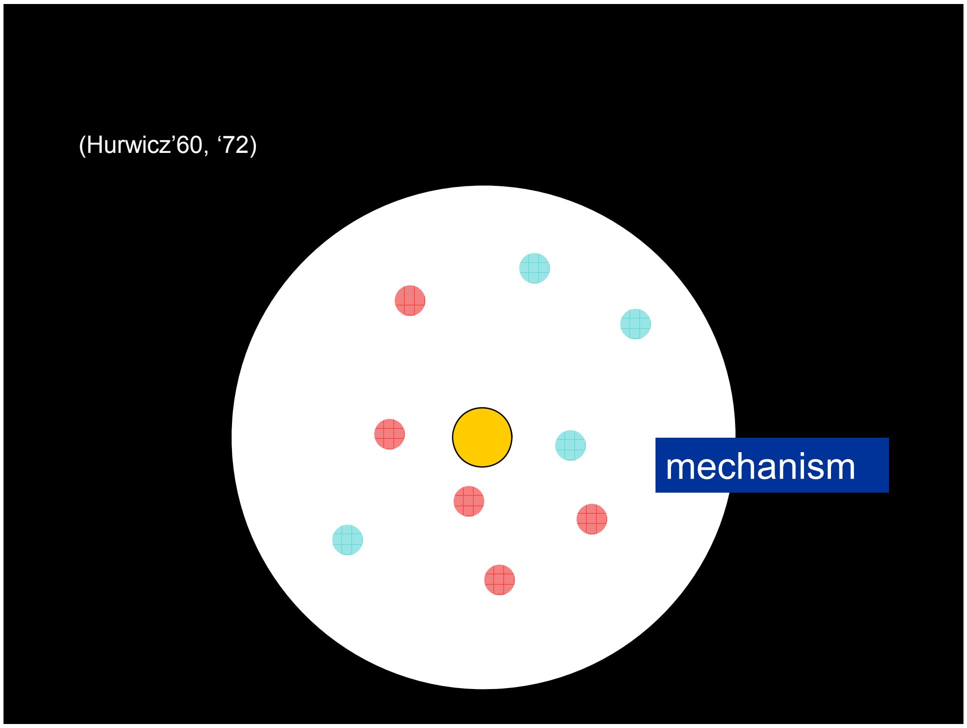
- Leonid Hurwicz (1960, 1972)
  - communication system, incentive compatibility
- Eric Maskin (1977)
  - Nash implementation
- Roger Myerson (1979, 1981)
  - Bayesian mechanism design

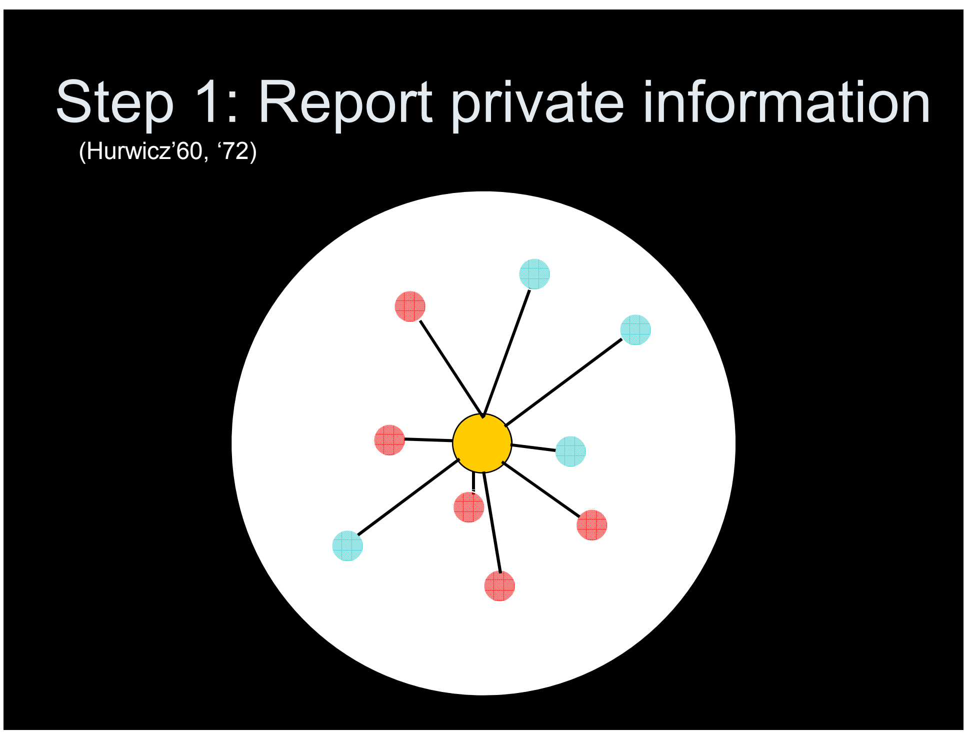


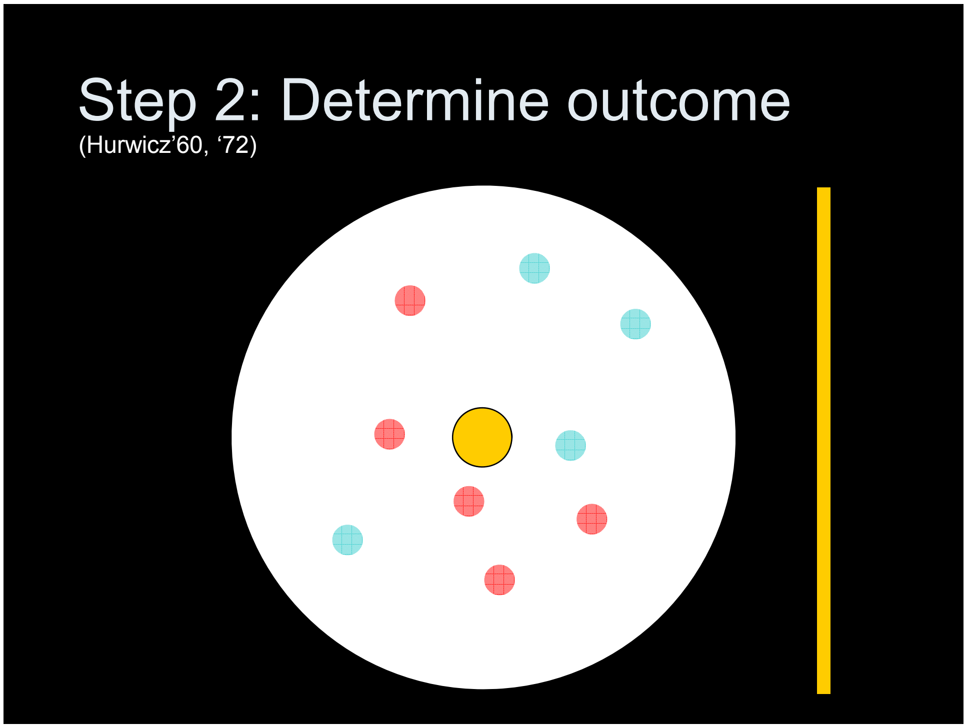
An Example

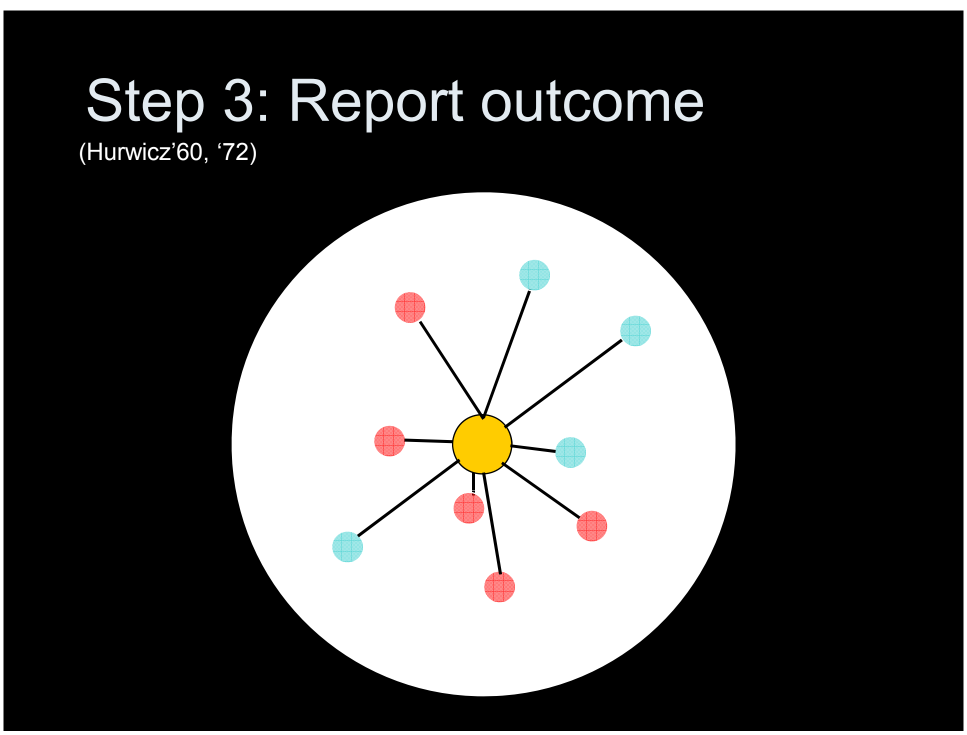


Goal: achieve desirable outcomes despite private information and self interest


(Hurwicz’60, ‘72)
mechanism


Step 1: Report private information
(Hurwicz’60, ‘72)


Step 2: Determine outcome
(Hurwicz'60, '72)


Step 3: Report outcome
(Hurwicz'60, '72)

The central question

- Outcomes X, Valuations $v_i \in V_i$
- Design a protocol (or "game") in which:
  - each agent makes a *claim* about its valuation
  - a center picks an outcome based on reports

The central question

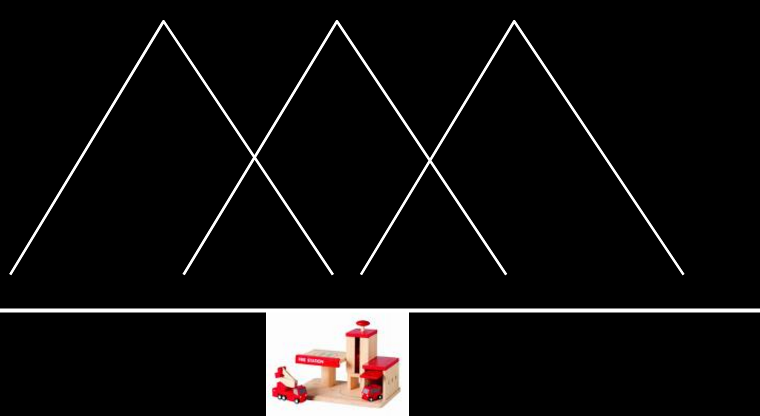
- Outcomes X, Valuations $v_i \in V_i$
- Design a protocol (or "game") in which:
  - each agent makes a *claim* about its valuation
  - a center picks an outcome based on reports
- Q: what functions $g: V^n \to X$ can be achieved in the equilibrium of <u>some</u> game?
  - "inverse game theory"

Example: Single-peaked preferences

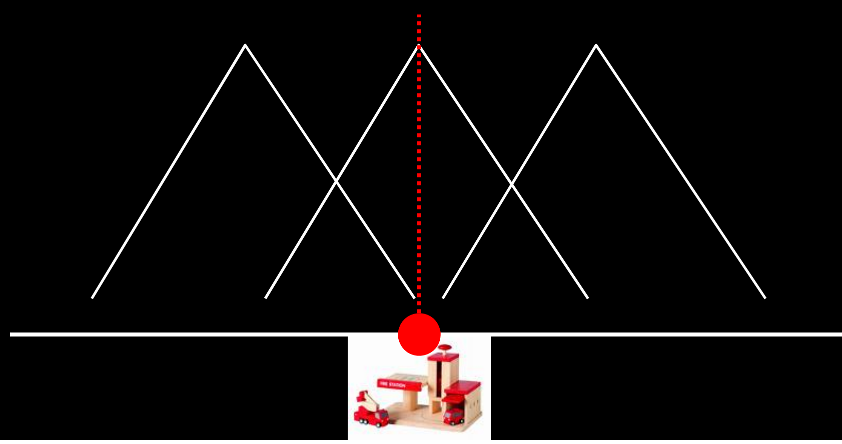
(Moulin'80)

Example: Single-peaked preferences

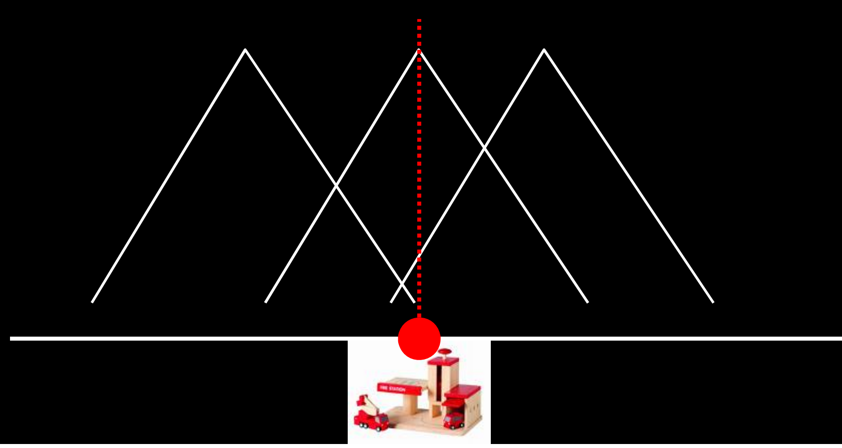
(Moulin'80)



Example: Single-peaked preferences
(Moulin'80)

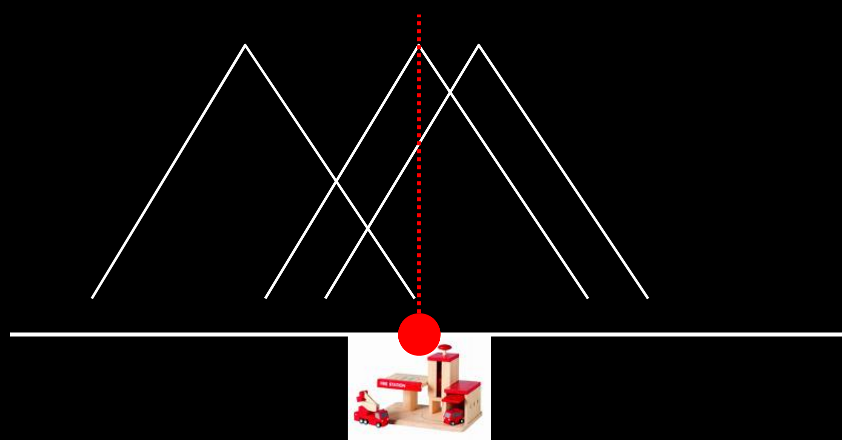


Example: Single-peaked preferences
(Moulin'80)



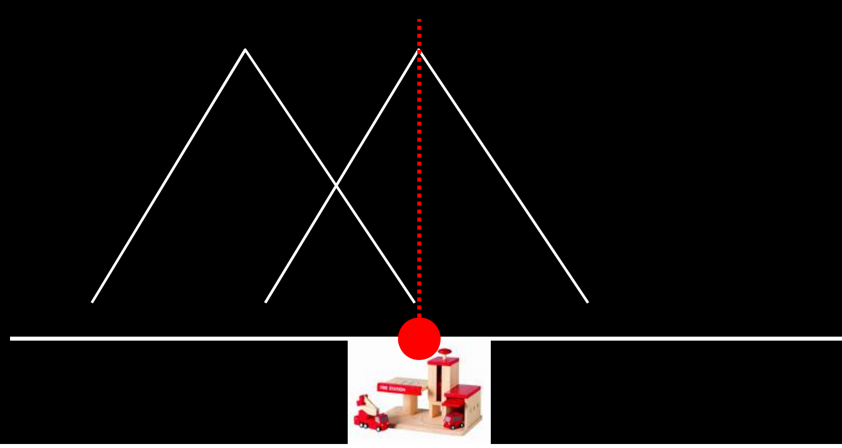
Example: Single-peaked preferences

(Moulin'80)



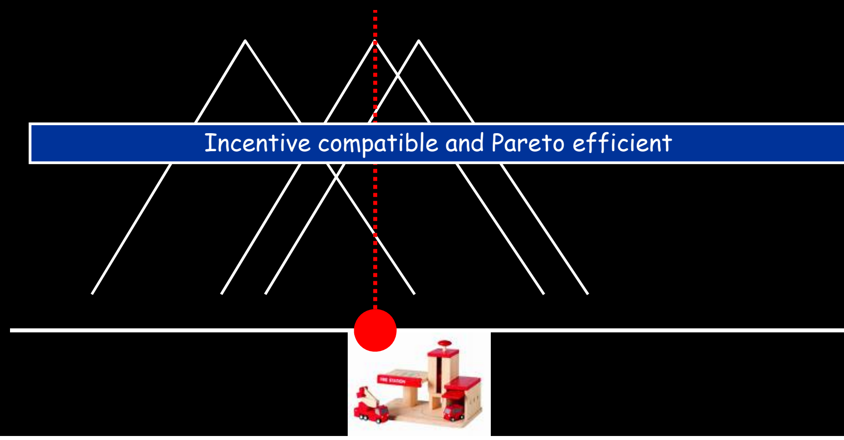
Example: Single-peaked preferences

(Moulin'80)



Example: Single-peaked preferences

(Moulin'80)



Incentive compatible and Pareto efficient

Example: Second-price auction

(Vickrey'61)

Incentive-compatible and Pareto efficient

Sell to  for $8.

Incentive Compatibility

- M=(g,t) is incentive compatible if the game has a truthful equilibrium
- An incentive-compatible M "implements" social choice function g: $V^n \to X$

- revised Q: what SCFs can be implemented?

IC $\equiv$ Monotonicity

(Myerson'81)

Example: Revenue optimality
(Myerson'81)

revenue-optimal auction across all incentive-compatible auctions
$\equiv$
maximizing "virtual valuation"

Example: Revenue optimality
(Myerson'81)

revenue-optimal auction across all incentive-compatible auctions
$\equiv$
maximizing "virtual valuation"

- Distribution function F
- Virtual valuation $\phi_i(v_i) = v_i - [1 - F_i(v_i)] / f_i(v_i)$

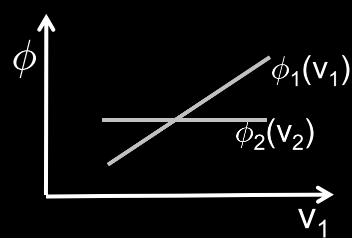
Example: Revenue optimality
(Myerson'81)

revenue-optimal auction across all incentive-compatible auctions

$\equiv$

maximizing "virtual valuation"

- Distribution function F
- Virtual valuation $\phi_i(v_i) = v_i - [1 - F_i(v_i)] / f_i(v_i)$



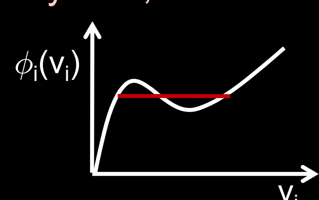
Example: Revenue optimality
(Myerson'81)

revenue-optimal auction across all incentive-compatible auctions

$\equiv$

maximizing ironed "virtual valuation"

- Distribution function F
- Virtual valuation $\phi_i(v_i) = v_i - [1 - F_i(v_i)] / f_i(v_i)$
- If monotonicity fails, then iron:



Negative results as well...

(Gibbard-Satterthwaite '73)

- If at least 3 outcomes, and all valuations $v_i \in R^X$ are possible, then the only SCF that can be implemented are dictatorial.

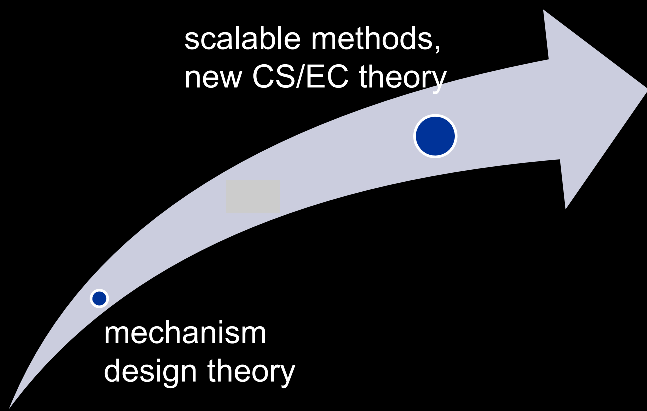
Modern Applications of MD

- Wireless spectrum allocation (everywhere)
- Airport landing slots (not yet)
- MBA course registration at Harvard
- School lunches in Chile
- Internet advertising
- Logistics sourcing
- "Cap and trade" pollution markets
- Entertainment tickets, collectibles
- ...

Mechanism = Algorithm

New challenges

- Large, combinatorial problems
  - rarely is private information a single number
  - rarely is winner-determination as simple as finding the “median” or the “max”
- Pragmatics $\rightarrow$ New desiderata:
  - informational efficiency
  - computational tractability
  - privacy, trust, ...



Good Progress

- New “indirect” versions of mechanisms
  - iterative auctions, methods of preference elicitation
- Fast winner determination
  - special cases (structure)
  - “branch-and-cut” search to solve very large winner determination problems quickly in practice

Good Progress

- New "indirect" versions of mechanisms
  - iterative auctions, methods of preference elicitation
- Fast winner determination
  - special cases (structure)
  - "branch-and-cut" search to solve very large winner determination problems quickly in practice
- New theory
  - e.g., require polynomial time + incentive-compatible
    - find monotonic approximation algorithms
  - e.g., "prior-free" mechanisms
  - new CS domains, e.g. network routing

But a remaining Bottleneck

- Analytic!
- Mechanism design <u>theory</u> becomes increasingly difficult in real-world domains
- Few new mechanisms of practical interest are analytically derived

But a remaining Bottleneck

- Analytic!
- Mechanism design theory becomes increasingly difficult in real-world domains
- Few new mechanisms of practical interest are analytically derived

- Example: how large of a revenue-optimal combinatorial auction has been designed?

But a remaining Bottleneck

- Analytic!
- Mechanism design theory becomes increasingly difficult in real-world domains
- Few new mechanisms of practical interest are analytically derived

- Example: how large of a revenue-optimal combinatorial auction has been designed?
  – two items ☹

One idea: “Automated MD”

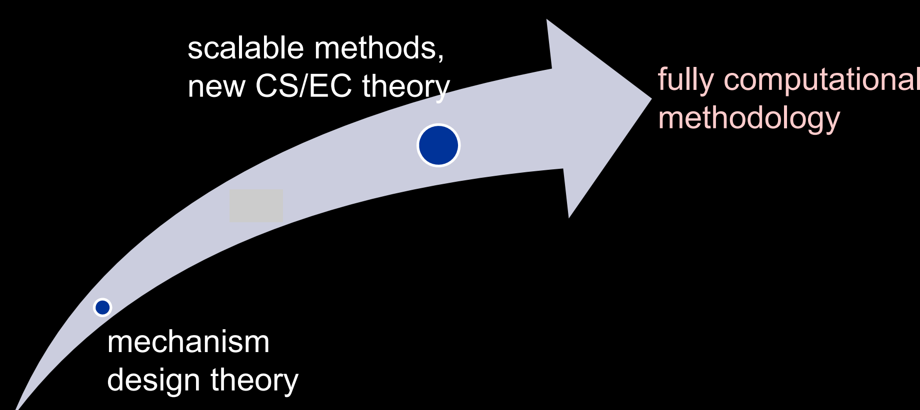
- Automated MD
  - discretize and solve via search (Conitzer & Sandholm’02)
  - not-applicable to combinatorial problems
- Parameterized MD
  - e.g., linear-affine weighted VCG (Likhodedov & Sandholm‘05)
  - doesn’t generate truly new mechanisms

“a problem of CMD is solved”

“a problem of CMD is solved”

=

can take a state-of-the-art computational method for solving the cooperative problem and apply- “with small modification” -to solve the MD problem



Case study: Dynamic Auctions

- Airline tickets
  - Jet Blue (Sep 8-14, 2008)
- Entertainment tickets
  - TicketMaster
- Sponsored search
  - Google, Microsoft, Yahoo!

- Many possibilities
  - cloud computing, wireless spectrum, logistics, taxi dispatch, airline ground-hold programs, ...

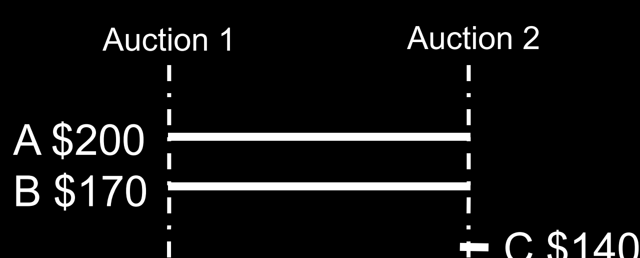
Simple Dynamic Auction Example

(Hajiaghayi et al. '05)

- Agent: arrival, departure, value. Can "lurk"
- Policy: sell to highest unallocated agent
- Payment: ~~second-highest bid~~

critical value

monotonic!



A wins in period 1, pays $140 upon departure

A More Complex Problem

- Multiple items to allocate (either each period, or by some deadline)
- Multi-unit demand, valuation r_i if allocated q_i units in some period $t \in \{a_i, ..., d_i\}$
- Probabilistic model of arrivals
- Goal: max. expected total value, or revenue

Analytic Bottleneck in Practice

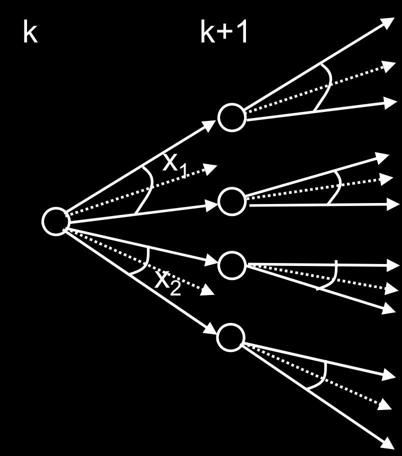
- Underlying (cooperative) problem has effective computational methods but no closed-form, analytic solution
- Provably optimal mechanisms have been designed only for simple special cases:
  - unit-demand (PV’08); unit-demand, renewal process (G’06); unit-demand, AR(k) process (PST’08)
  - impatient bidders, Poisson distr., simple type (GM’08)

Solve = take state-of-the-art method for stochastic optimization and with a slight perturbation use it for clearing (and pricing) dynamic auctions

Computational Problem

- Can model as a Markov Decision Process
- State space:
  - value, patience and demand of current agents
  - history of past allocation decisions
- Transition function:
  - depends on decision & probabilistic arrival model
- Action: allocate resource(s) to agents
- Reward: total value received by agents

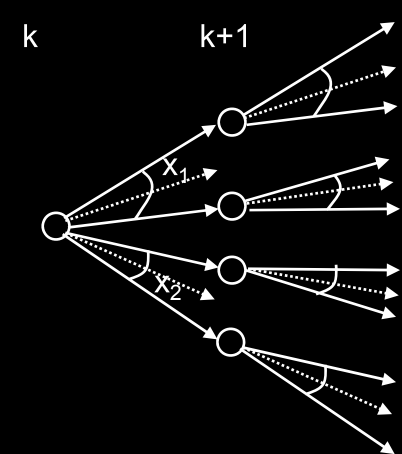
Solving MDPs

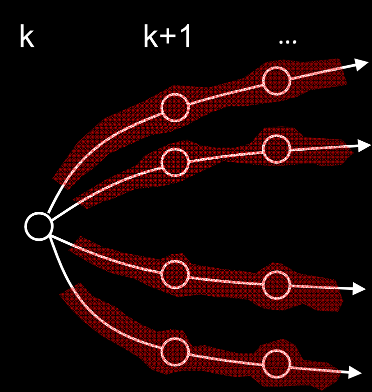


tree-sampling(ε)

(Kearns et al. '99)

Solving MDPs





tree-sampling(ε)

(Kearns et al. '99)

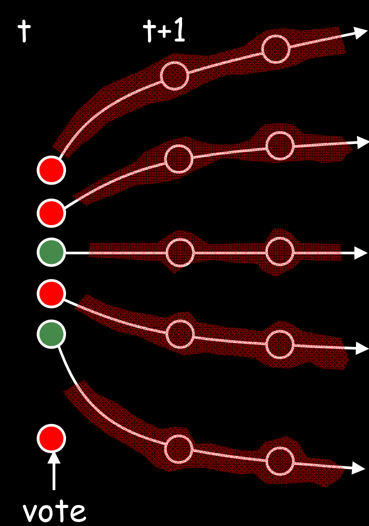
Online Stoch. Comb. Optim.

Exogoneous uncertainty

$Pr(v^k | v^{1..k-1}, x^{1..k-1}) = Pr(v^k | v^{1..k-1})$

(van Hentenryck and Bent'04)

Example: Consensus Algorithm
(van Hentenryck and Bent'04)



Solve = take Consensus algorithm for stochastic optimization and with a slight perturbation use it for clearing (and pricing) dynamic auctions

Perturb (as necessary) to make IC

- Policy π is IC if and only if it is “monotonic”
- Type: arrival, departure, value, quantity

(Hajiaghayi et al. ‘05)

Will policies determined via stochastic optimization be monotonic?

- 3 units to sell. 2 decision periods.

period 1	period 2
A_1: (1,1,q=1,\$5)	Pr $1-\epsilon$ (2,2,q=3,\$1000)
A_2: (1,2,q=2,\$500)	Pr ϵ (2,2,q=1,\$5000)
g^1:={}	A_3: (2,2,q=1,\$5000) arrives
	g^2:={A_2, A_3}

- 3 units to sell. 2 decision periods.

period 1	period 2
A_1: (1,1,q=1,\$5)	Pr $1-\epsilon$ (2,2,q=3,\$1000)
A_2: (1,2,q=2,~~\$500~~)	Pr ϵ (2,2,q=1,\$5000)
~~g^1:={}~~	A_3: (2,2,q=1,\$5000) arrives
	~~g^2:={A_2, A_3}~~

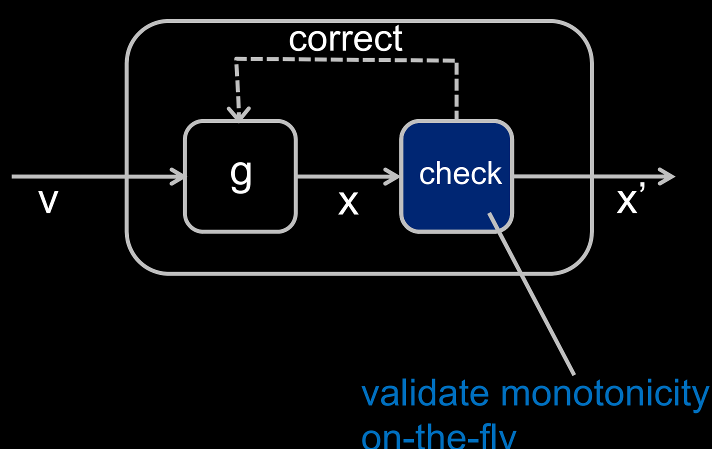
\$1000

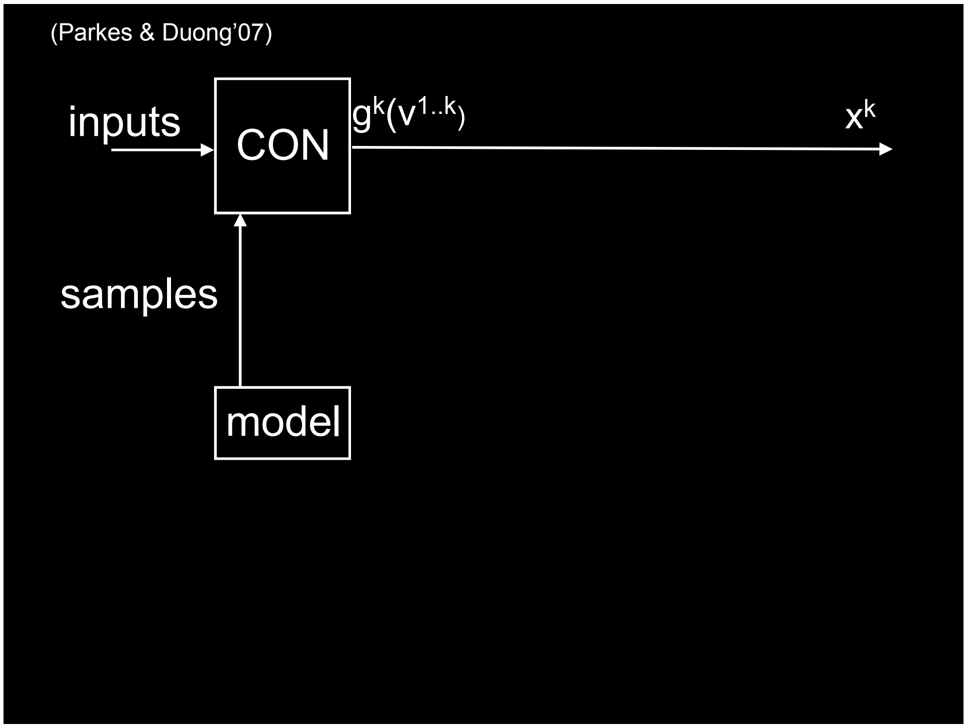
g^1={A_1} g^2={A_3}

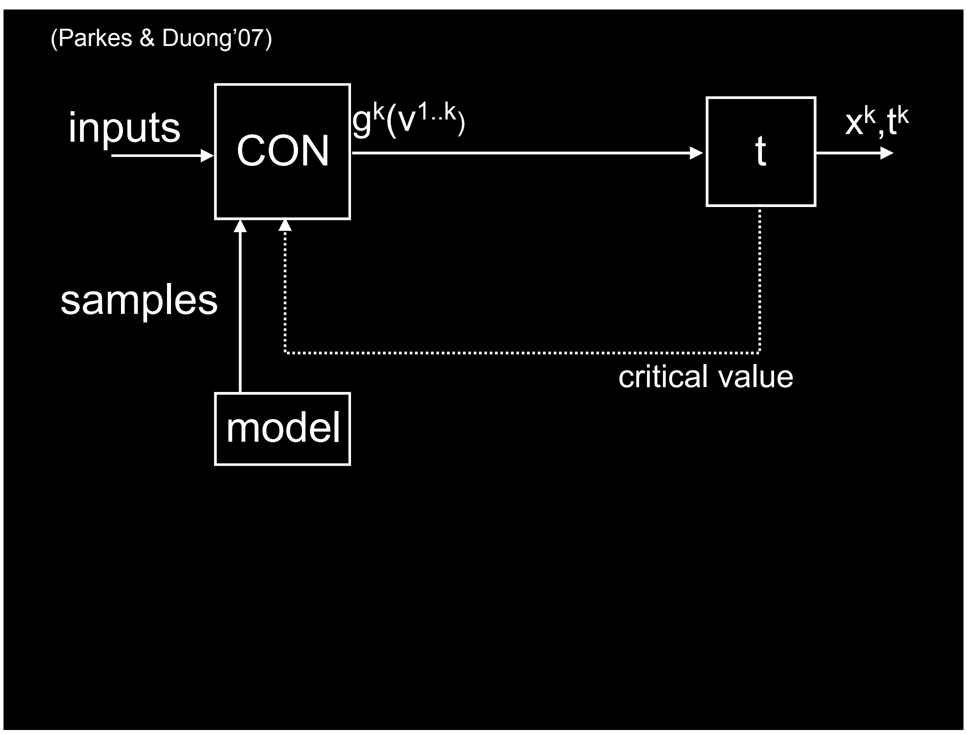
Solve = take Consensus algorithm for stochastic optimization and with a slight perturbation make it monotonic

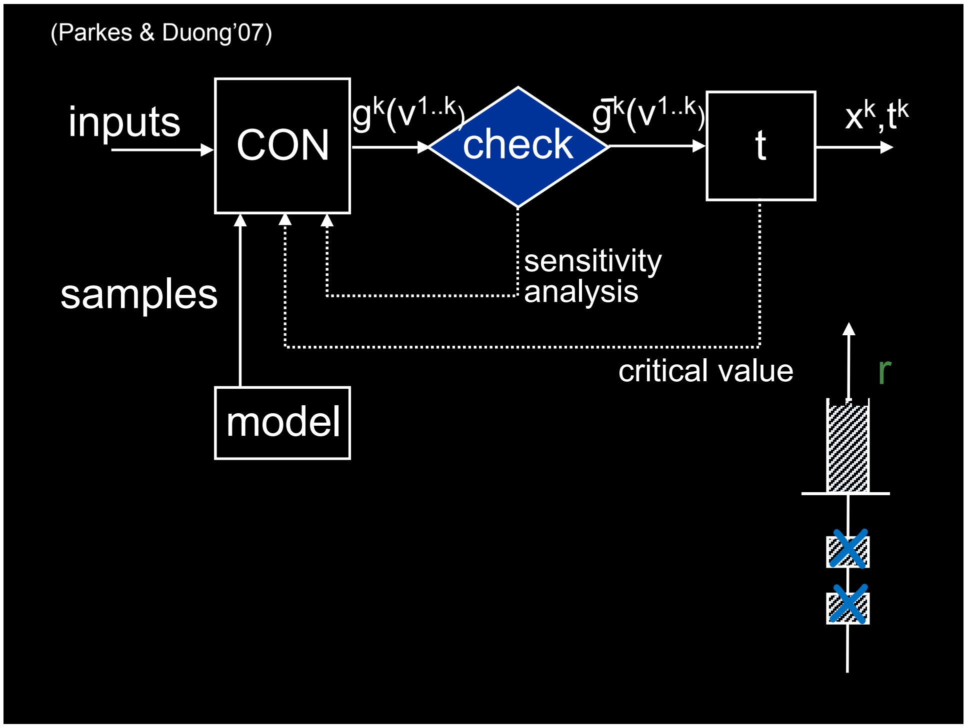
Self-correcting MD

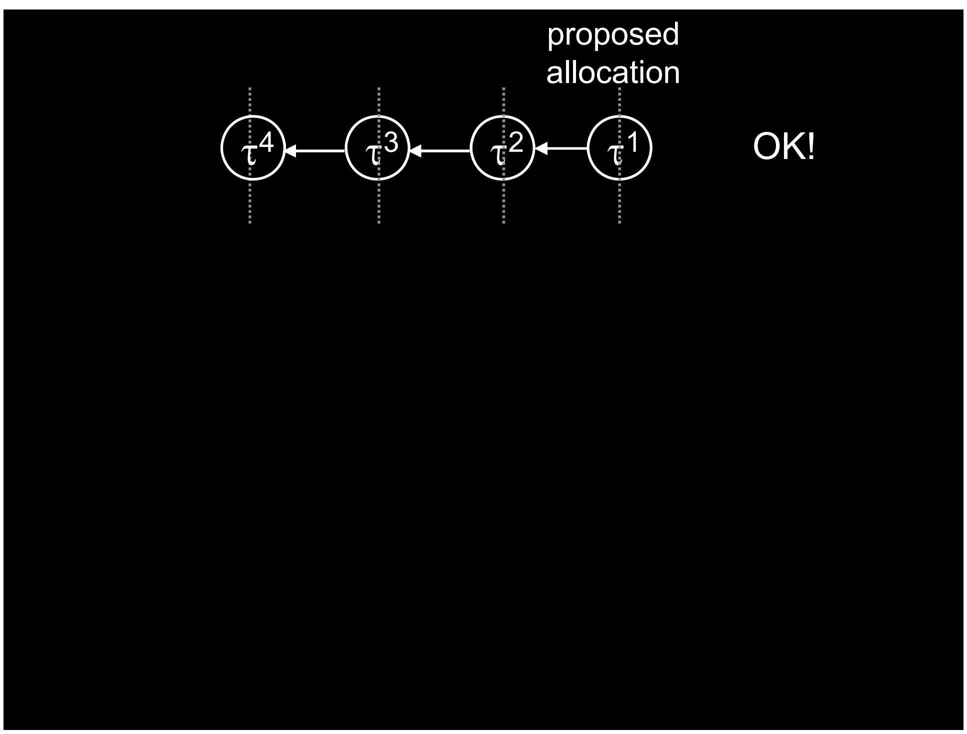
(Parkes & Duong'07)

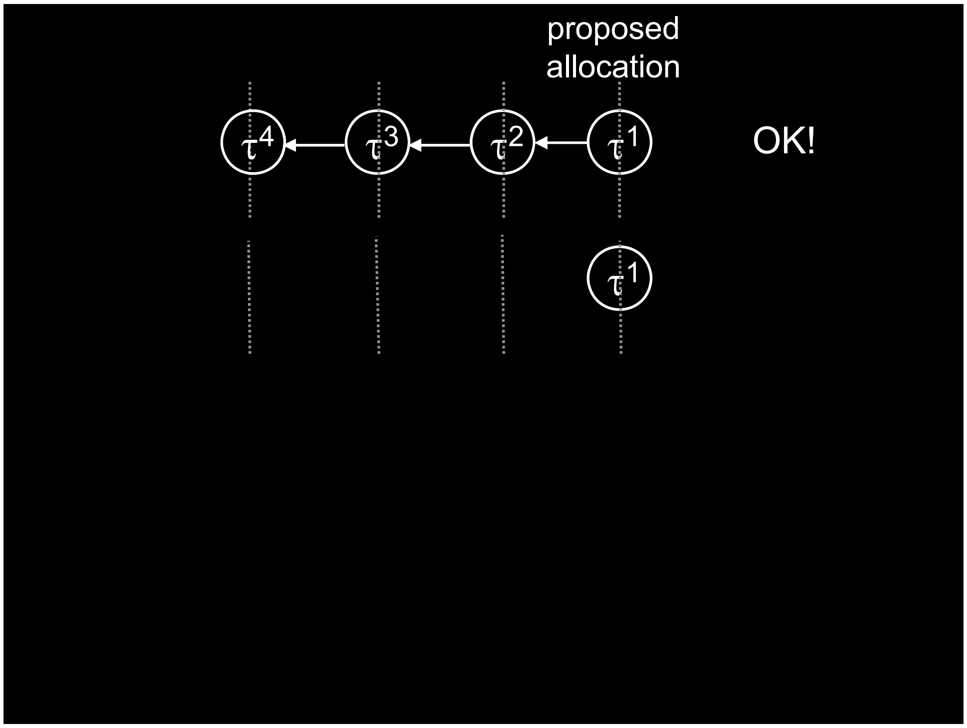


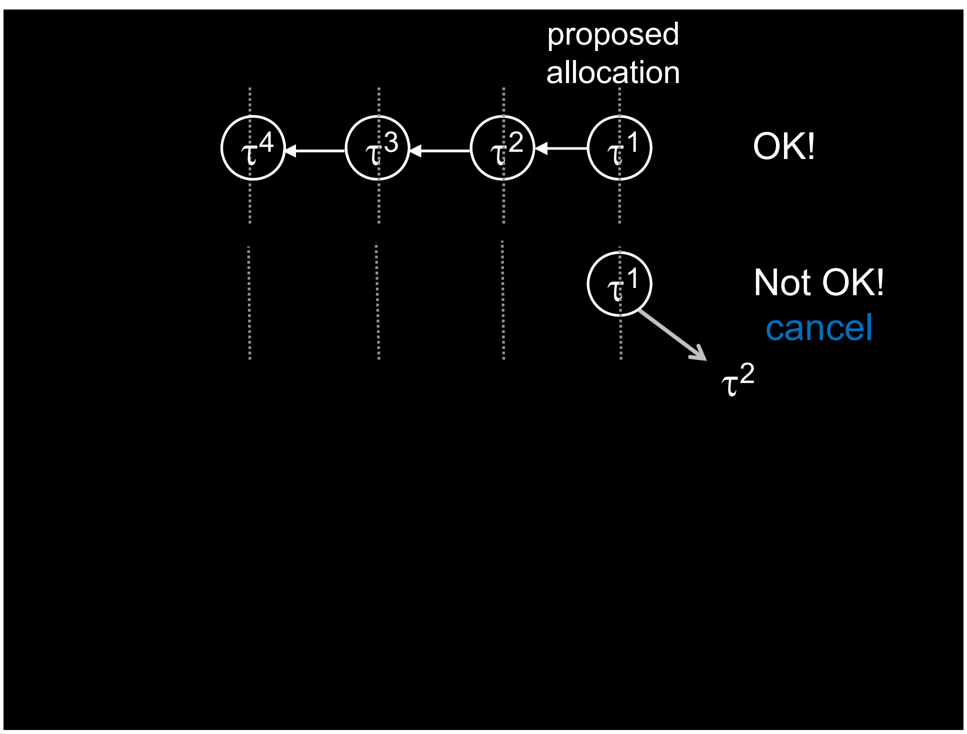

(Parkes & Duong'07)
inputs
CON
$g^k(v^{1..k})$
x^k
samples
model

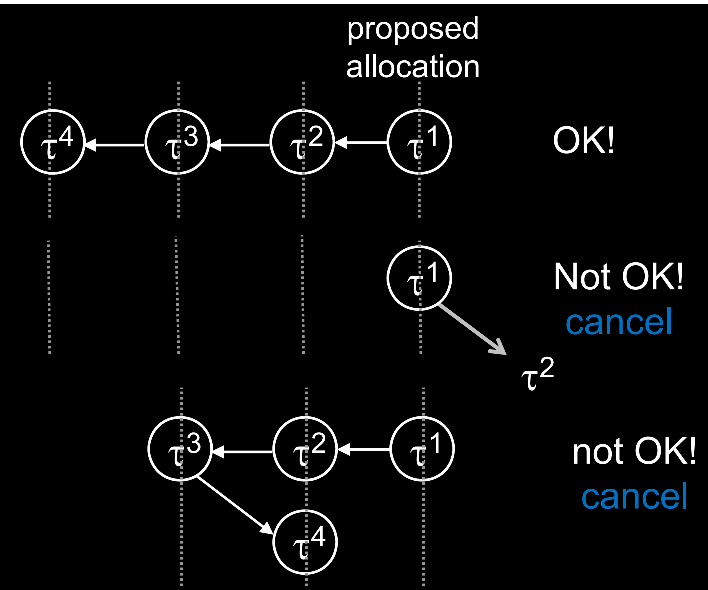

(Parkes & Duong'07)
inputs
CON
$g^k(v^{1..k})$
t
x^k,t^k
samples
critical value
model


(Parkes & Duong'07)
inputs
CON
$g^k(v^{1..k})$
check
$\bar{g}^k(v^{1..k})$
t
x^k,t^k
samples
sensitivity analysis
critical value
r
model


proposed allocation
τ^4
τ^3
τ^2
τ^1
OK!


proposed
allocation
τ4
τ3
τ2
τ1
OK!
τ1

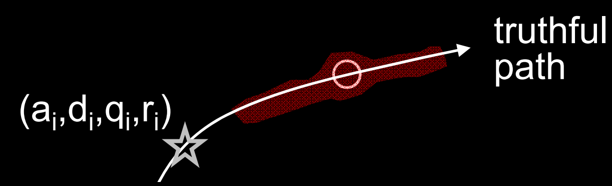

proposed
allocation
τ4
τ3
τ2
τ1
OK!
τ1
Not OK!
cancel
τ2

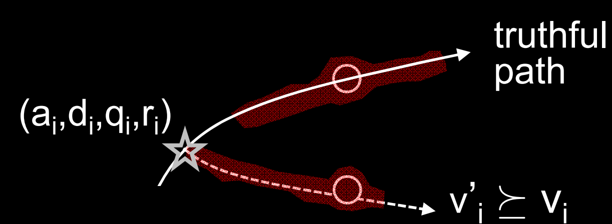


Lemma: the "corrected" policy is monotonic

(Parkes & Duong'07)

- Theorem. The Consensus algorithm, coupled with 'self-correction' and critical-value payments is incentive compatible



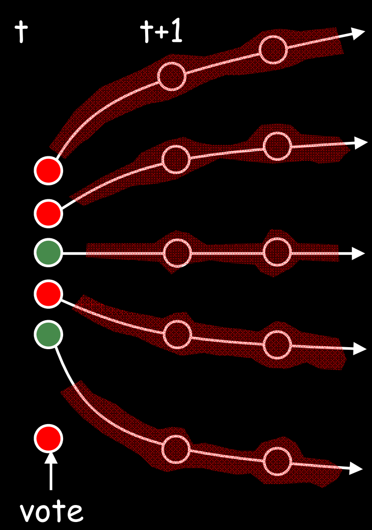


Need exogeneous uncertainty

$$Pr(v^k|\ v^{1..k-1},x^{1..k-1})=Pr(v^k|v^{1..k-1})$$

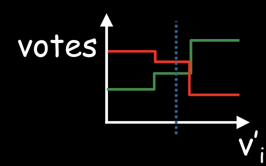
Computation (I)

(Parkes & Duong'07)



* vote changes small number of times as v'_i varies

* can compute these "breakpoints" by a combinatorial argument



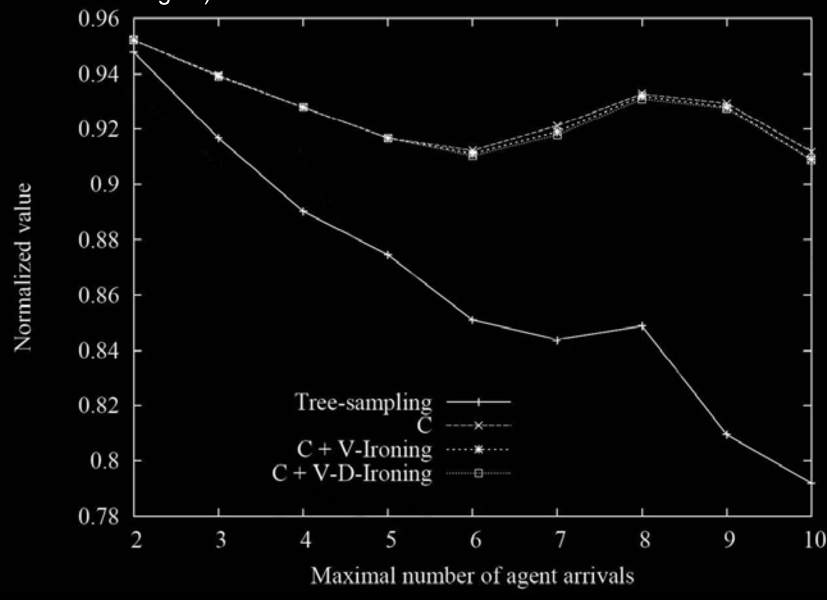
Computation (II)

(Constantin & Parkes'08)

- Naïve algorithm is quadratic in number of larger types
- Linear algorithm:
- Check that the allocation is "monotonic earlier" for all $v_i'' \in v_i'{}^{++}$, for all $v'_i \succeq v_i$.

Example: Scheduling Domain

(Parkes & Duong'07)



Example: Allocation Domain

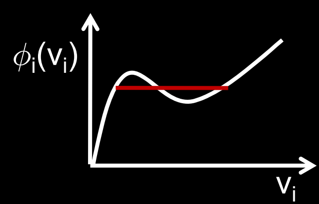
(Constantin & Parkes'08)

- 10 item auction
- Bidders U[1,5] demand, U[1,5] patience
- Exponential value distribution

Ironing	NowWait	OnlyDep	HROrRew	IgnoDep	Opt
No	0.946	0.937	0.840	0.840	1
Yes	0.908	0.569	0.834	0.834	1

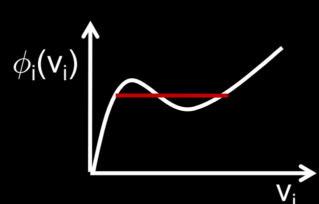
Ironing (two kinds)

Myerson
(input ironing)



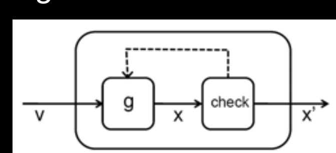
Ironing (two kinds)

Myerson
(input ironing)



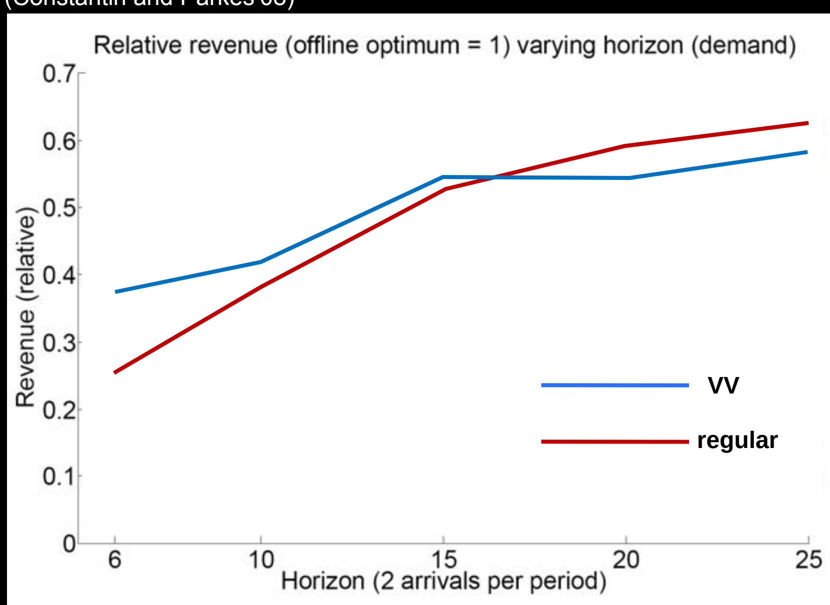
Output ironing

- "Iron" out the non-monotonicity of algorithmic rule



Can also combine with input ironing
and boost revenue

(Constantin and Parkes'08)



Solve = take Consensus algorithm for stochastic optimization and with output ironing use it for clearing (and pricing) dynamic auctions

Not doing away with analysis ☺

- Use analysis for game-theoretic purposes, to characterize incentive-compatible policies. (Rochet'87, Lavi et al. '03, Saks& Yu'05, Gui et al.'04, Archer & Kleinberg'08)
- Enables a heuristic ("fully computational") approach for mechanism design

Future Directions

- Combine this "perturbation" approach with other behavioral models
  - spiteful agents
  - non-equilibrium play (e.g. promote good best-response dynamics)
  - ...
- Adapt output ironing to other algorithmic methods, including offline mechanisms.

Summary

- Pragmatic, computational agenda for mechanism design
- "Solved" = small modification to (heuristic) algorithm to make incentive compatible
- Case study: stochastic optimization for dynamic mechanisms

www.eecs.harvard.edu/econcs