Towards Iterative Combinatorial Exchanges

FCC Combinatorial Auction Workshop,
November 23nd, 2003

David C. Parkes
Harvard University.

Motivation

- Highly fragmented spectrum (frequency, control, and geography) $\Rightarrow$ proposal for "big-bang" exchange
- Assumptions:
  - forced relocation of spectrum to alternative bands still leaves a substantial bargaining problem, and high transaction costs
  - efficient reallocation is the main goal (although "reasonable" FCC revenue important)

Combinatorial Exchanges

- Multiple buyers and sellers, w/ expressive bids
  - e.g. "Buy 10MHz in NYC counties A, B, C and D for $1million", "Sell 78-84 MHz in counties A and D for $300,000"

- FCC can also participate, actively:
  - e.g. the *only* agent able to buy ITFS licenses and convert into flexible-use licenses"

  and passively (define aggregations):
  - e.g. "all contiguous 6 MHz blocks of spectrum in a BTA are equivalent"

Main Challenges

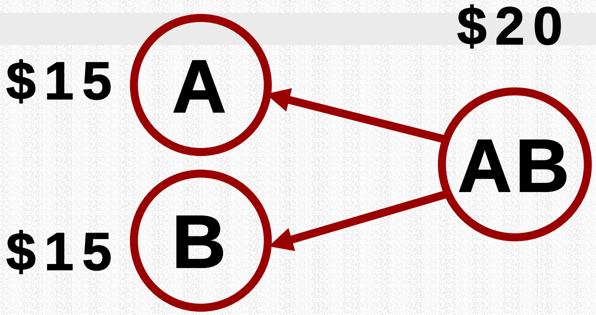
- Winner-determination
  - likely to be harder than one-sided auctions (Sandholm's talk)
- Economic
  - mitigating the bargaining or "hold-out" problem
- Preference elicitation
  - hard valuation problems
  - iterative designs likely important to guide elicitation

Main Challenges

- Winner-determination
  - likely to be harder than one-sided auctions
- Economic
  - mitigating the bargaining or “hold-out” problem
- Preference elicitation
  - hard valuation problems
  - iterative designs likely important to guide elicitation

Bargaining Problem.

Example:



- Many *ex post* Nash equilibrium:
  - ($5,$15,$20); ($10,$10,$20); ($15,$15,$30)...
  - presents an efficiency problem, because agents need to select an equilibrium.

- Construct *ex post* Nash:
  - allocate π_i to some agent i, with $V(N)-V(N\backslash i)>0$
  - adjust values, and repeat.

A One-Shot Design

(Parkes, Kalagnanam and Eso, 2001)

- Collect bids
- Compute $V(N)$, value of surplus-maximizing trade given all bids
  - implement this outcome
- Compute $V(N \backslash i)$, value of surplus-maximizing trade without bids from i
- Divide surplus $\sum_i \pi_i = V(N)$ across participants
  - try to mitigate bargaining problem

Surplus Division

Allocate payoffs, $\pi_i \geq 0$, to satisfy:

$$\textstyle\sum_i \pi_i \leq V(N) \qquad \text{(BB)}$$

$$\pi_i \leq V(N)-V(N\backslash i),\ \forall\ i \qquad (*)$$

$$\pi_i \geq 0,\ \forall\ i \qquad \text{(P)}$$

$(*)$ is just $(\sum_i \pi_i = V(N)$ and $\sum_{j\neq i} \pi_j \geq V(N\backslash i))$

Lemma. Any mechanism that implements V(N) and satisfies (BB), (*), and (P) has *ex post regret* $\pi_{VCG,i}$ for agent i, given bids of other agents.

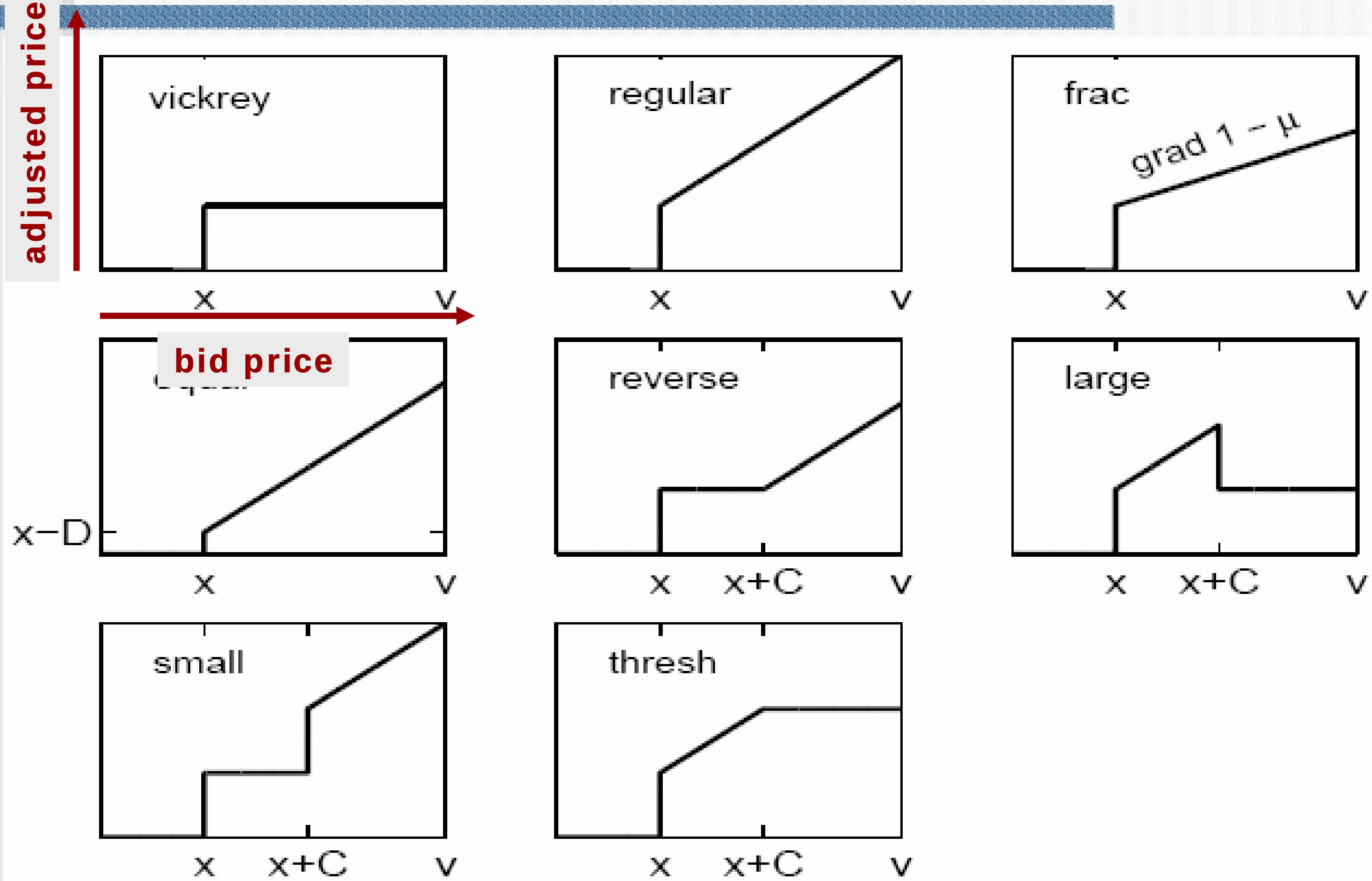
VCG-Based Schemes

Consider VCG-based schemes. Set objective to min $D(\pi, \pi_{VCG})$, for distance $D(.,.)$.

- Threshold. Minimize worst-case $(\pi_{VCG,i} - \pi_i)$
  - i.e. minimize the maximal ex post regret.

- Fractional. Each agent gets $\pi_i = \mu\ \pi_{VCG,i}$
- Large. Allocate payoff in order $\pi_1, \pi_2, \pi_3 \ldots$
- Reverse….

Stylized Representations



Threshold: Special Cases

- Implements the k-DA uniform price, double auction with $k=0.5$ (Wilson'85)
  - Threshold payoff division implemented with price $p^*=0.5(\min(a_{k+1},b_k)+\max(b_{k+1},a_k))$, asks $a_1<a_2<\ldots<a_m$, bids $b_1>b_2>\ldots b_m$, k items trade
- Second-best (for efficiency) for the standard single item bargaining problem, for iid and Uniform [0,1] values and costs (Myerson & Satterthwaite, 83)

Experimental Validation

- Limited strategy space:
  - $b_i(S) = (1-\alpha)\ v_i(S),\ \forall\ S$, if buyer
  - $b_i(S) = (1+\alpha) v_i(S),\ \forall\ S$, if seller

- Compute a symmetric *ex ante* BNE:

$$\alpha^* = \arg\max_\alpha E_i E_{-i} \left[v_i(\mathbf{x}(\alpha,\alpha^*)) - \mathbf{p}_i(\alpha,\alpha^*) \right]$$

$x^*(\alpha,\alpha^*)$ is the allocation,

$p_i(\alpha,\alpha^*)$ is payment to agent i.

Naive Approach

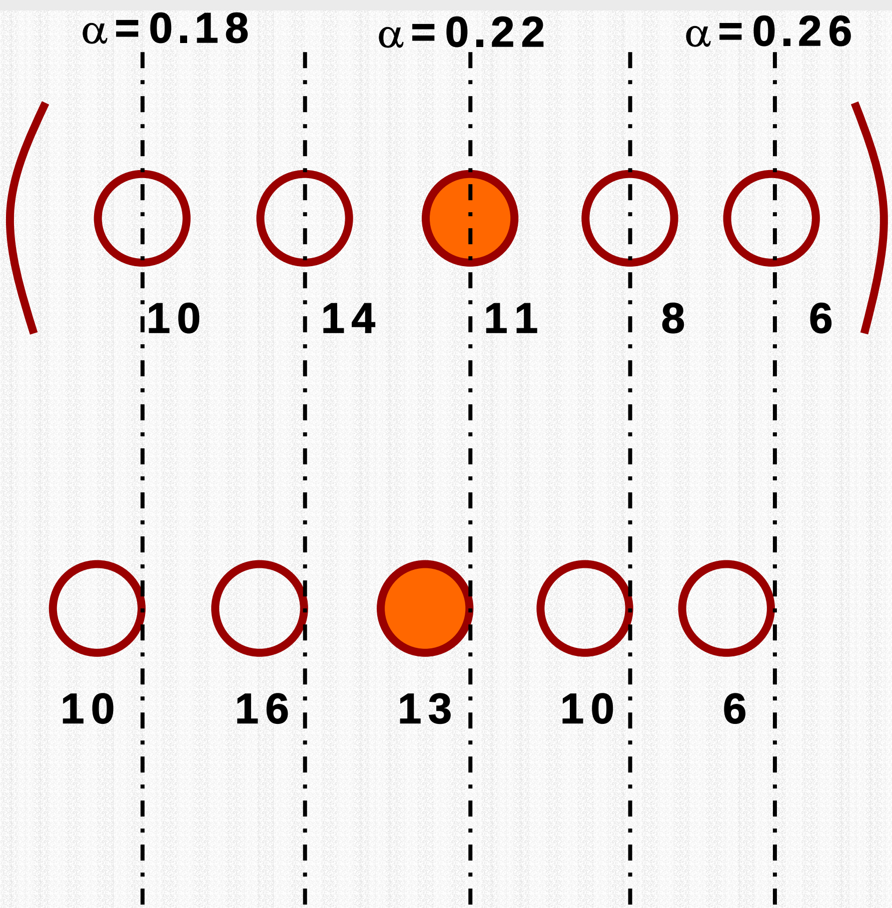
- Enumerate a payoff matrix, compute *ex ante* BNE

α_{-i}

α_i	-0.5	-0.48	…	0	…	0.12	0.14	…	0.98	1.0
-0.5	1.5	1.4								
-0.48	1.3									
…										
0										
…										
0.12										
0.14										
…										
0.98										
1.0										

Took 2.5 days, for a grid size of 0.01, 500 instances, 5 buyers, 5 sellers, 20 goods, 10 bids/asks per agent.

Iterative Approach.



Algorithm

(w/ David Kyrch)

- Choose a small set of strategies $A^t = (\alpha^t_1, \ldots, \alpha^t_M)$.
- Assume all agents except agent 1 play $\alpha^t \in A^t$
- Compute the BR, $\alpha^* \in A^t$, given α^t
- Move α^{t+1} towards α^*
- Refine A^t to focus search.

Details.

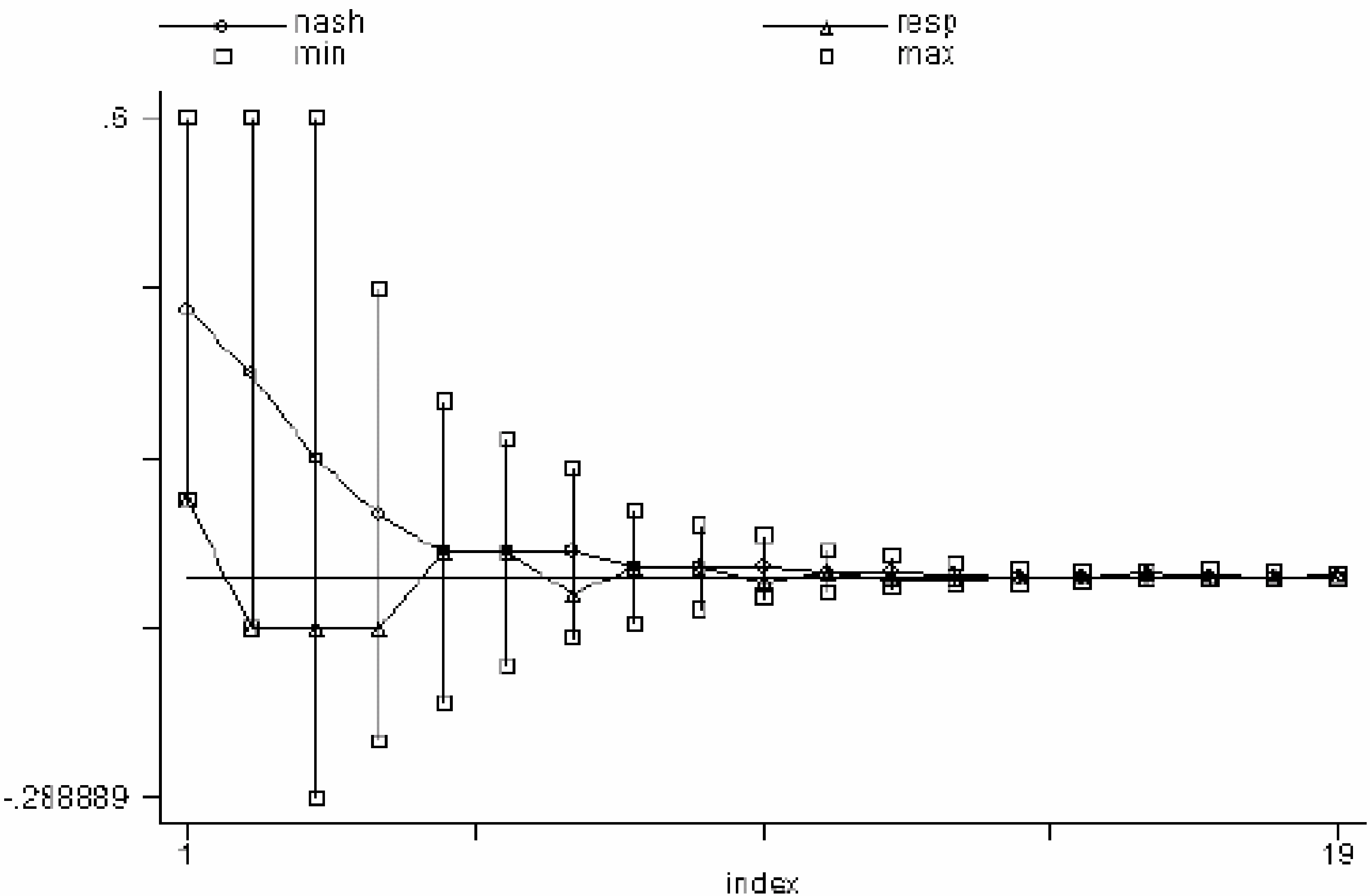
- Select 7 points in A
- New center: $\alpha^{t+1}=1/3\alpha^t + 2/3\ \alpha^*$
- Select a new range A^{t+1}, centered on α^{t+1}
  - $|A^{t+1}|= 3/4\ |A^t|$, if α^{t+1} within current range
  - $|A^{t+1}|=4(\alpha^{t+1}-\alpha^t)$, otherwse.

- Terminate when α^* is within 0.01 of α^t
  - finally validate that α^* is a BR to α^* over entire range [-0.5,1.0]

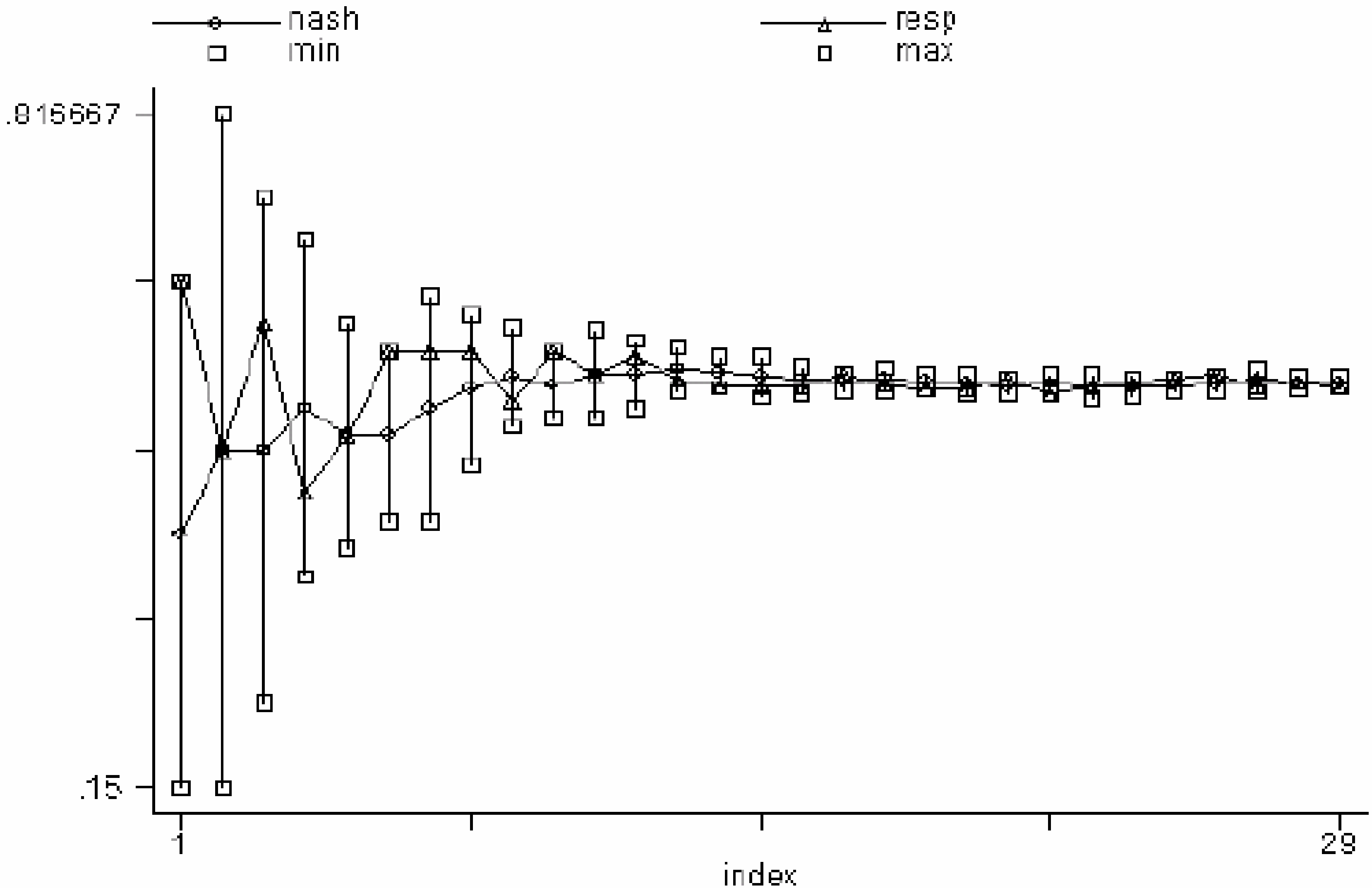
Experimental Results

- 5 buyers, 5 sellers, 20 goods
- 10 bundles/agent.
  - Uniform (Sandholm'99), XOR valuations.
- 500 instances
- Compute 1% accuracy in 2.5 CPU hours.

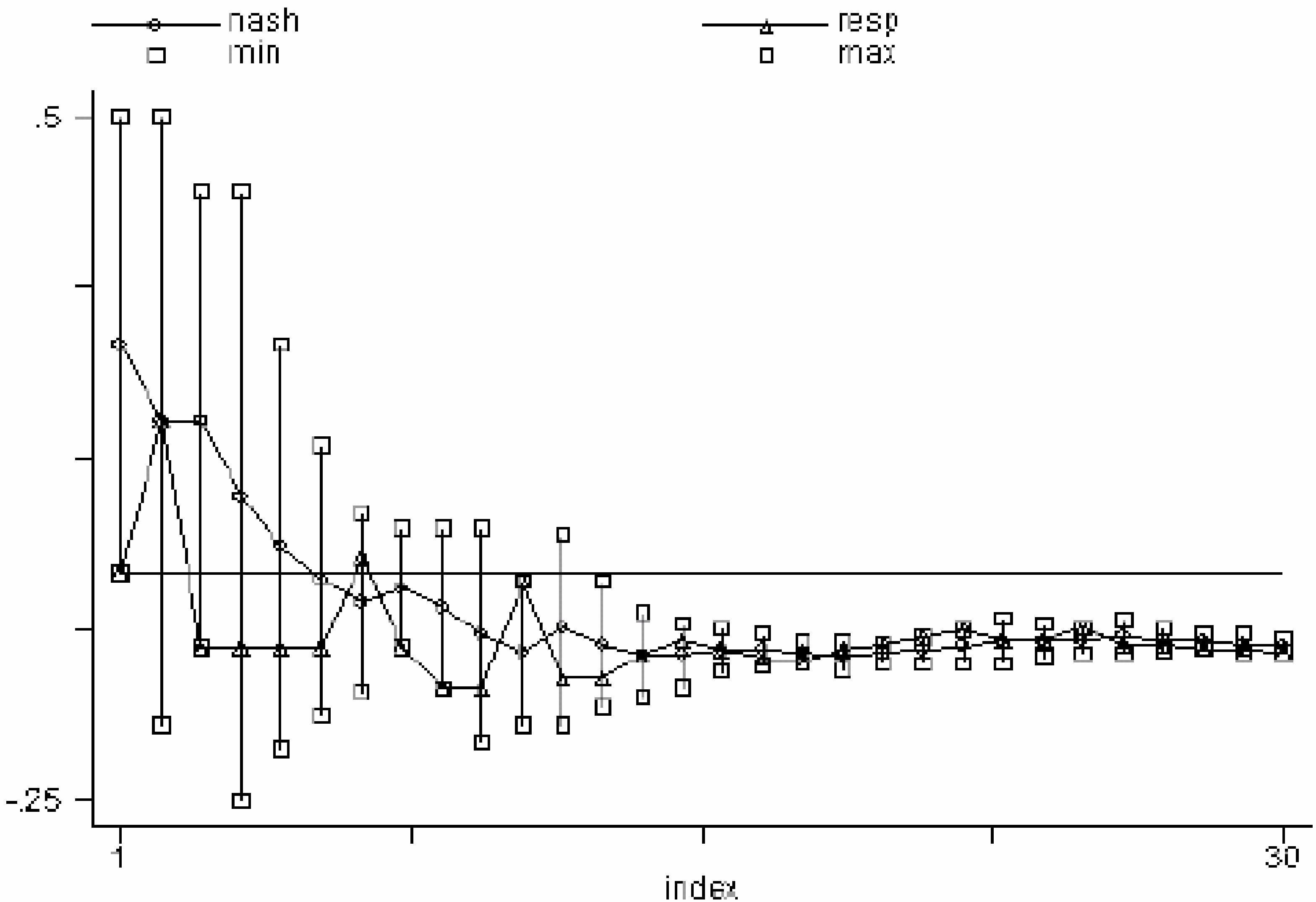
Example 1- VCG payments



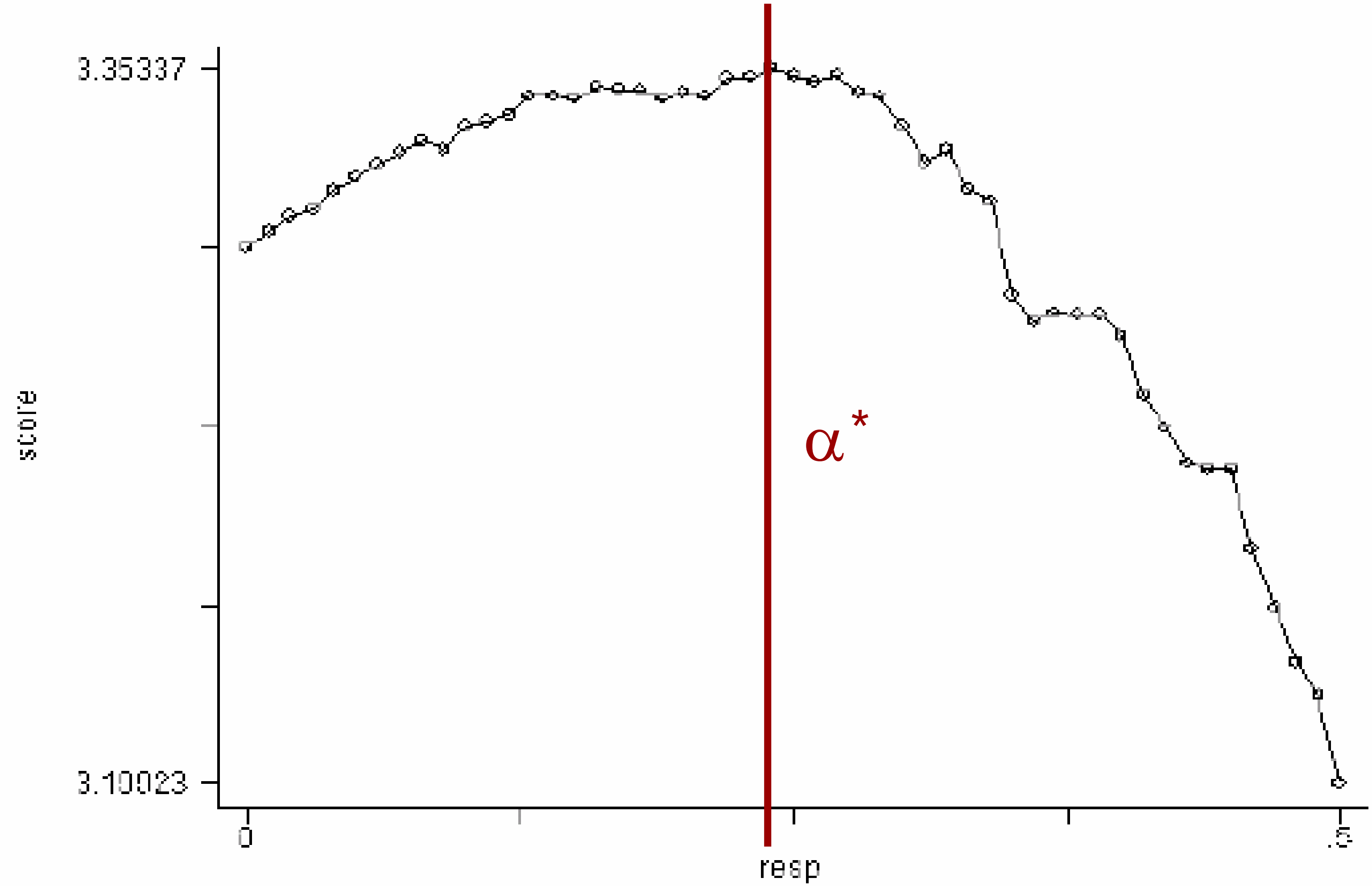
Example 2- No Discount



Example 3- Large

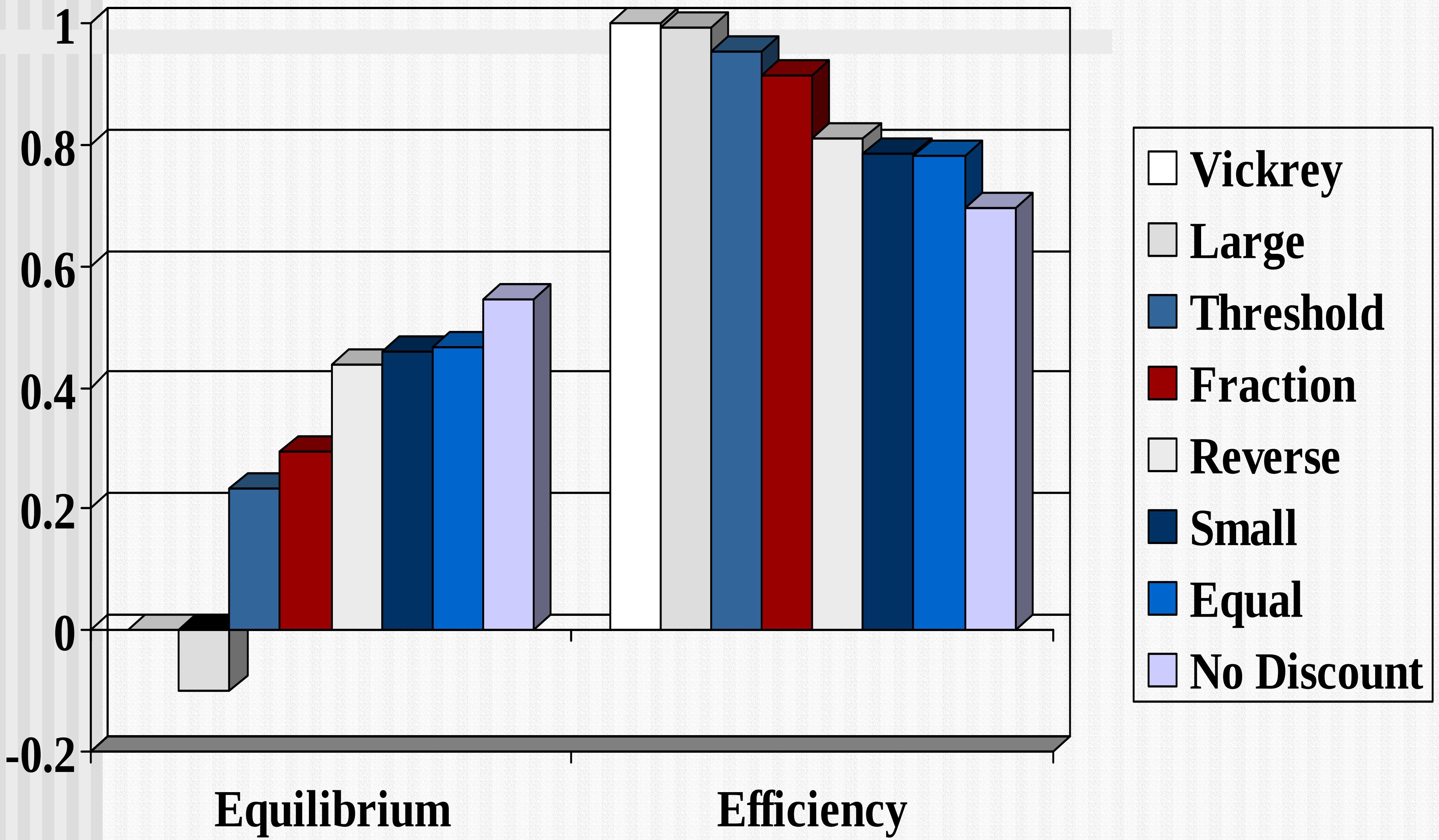


Example: Validity



Validating *ex post* Nash of Threshold rule

Main Results



(99% efficient)

Threshold vs. Large

(95% efficient)

- Optimal strategy in Large is to overbid
  - at least one participant has negative ex post payoff in BNE
  - an agent in efficient allocation can bid $v+\Delta$, large Δ, and ensure $\pi_{VCG,i}$
- Buyers in Threshold can only benefit by decreasing their bid, and then only if
  - their bid is adjusted by more than their Threshold payoff, or
  - there is some $V(N\backslash j)$, $j \neq i$, without i.

FCC: A Special Player (Milgrom)

- Can also apply core constraints for the FCC

$$\pi_{FCC} + \sum_{i \in L} \pi_i \geq V(FCC \cup L),\ \forall L \subseteq (N \setminus FCC)$$

- FCC cannot propose an alternative with more revenue that a subset of participants will all prefer (based on their reports).

- Helps to prevent “give aways.”

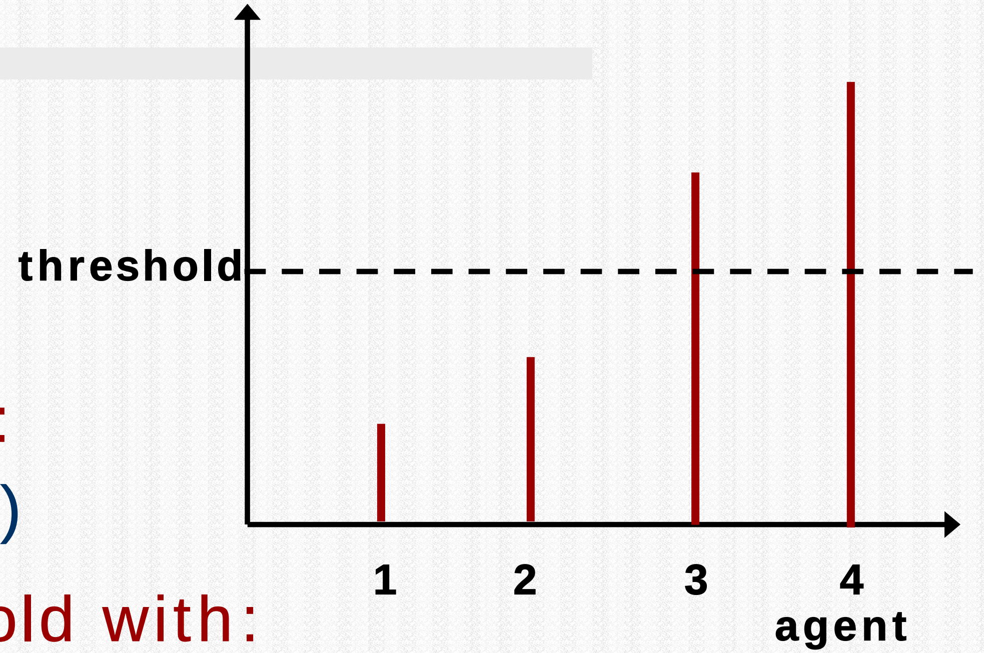
Main Challenges

- Winner-determination
  - likely to be harder than one-sided auctions
- Economic
  - mitigating the bargaining or “hold-out” problem
- Preference elicitation
  - hard valuation problems
  - iterative designs likely important to guide elicitation

Elicitation for Exchanges: Key Problems.

- Item discovery
  - scope of exchange may not be initially known
- Price discovery
  - may be *no trade* in initial stages
- Bargaining
  - the bargaining problem is omnipresent

Threshold Information Requirements.



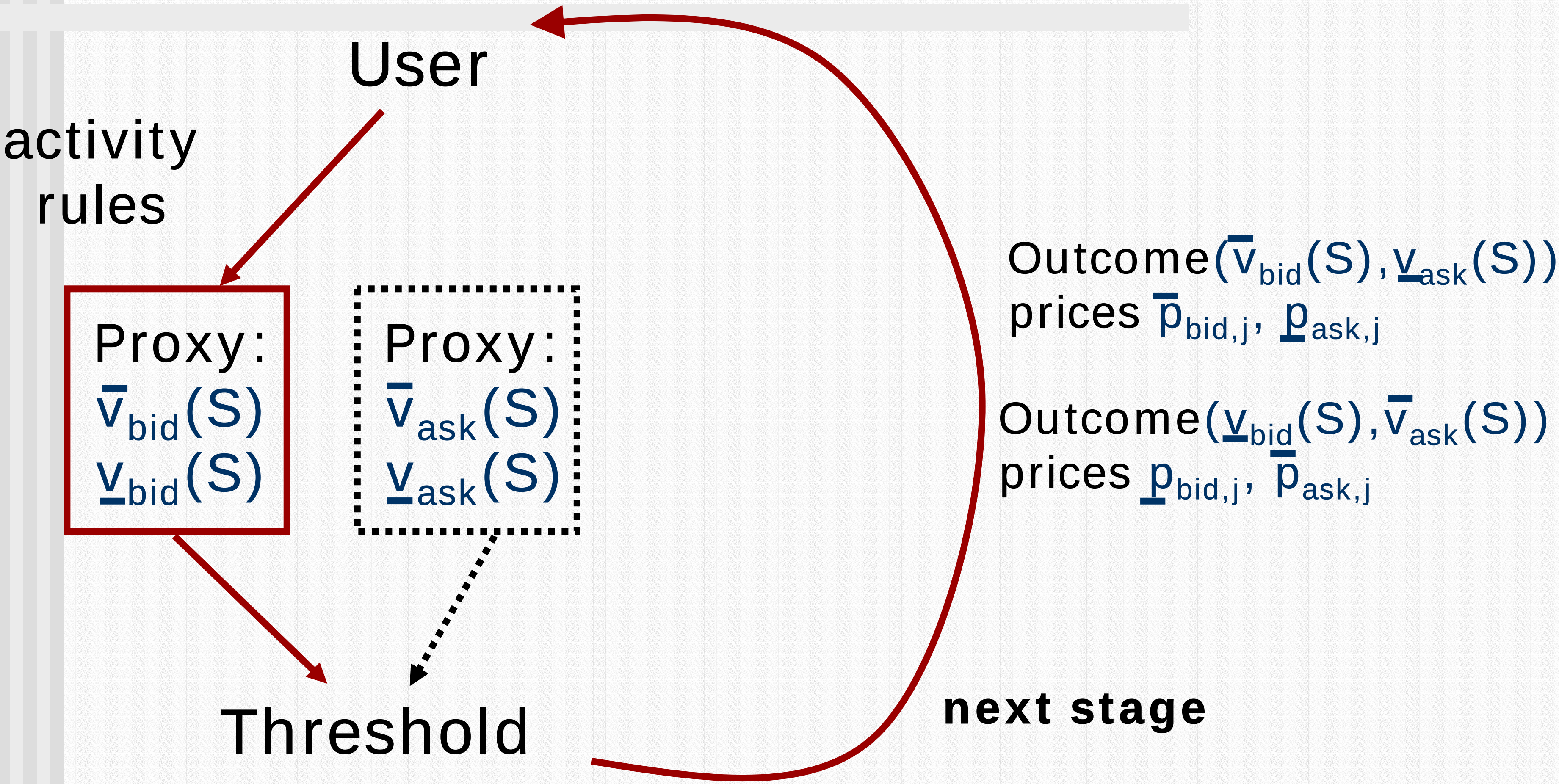
Consider information:

$v'_i(S)=\max(0,v_i(S)-\Delta_i)$

Can compute Threshold with:

1. Complete info from all losers
2. Winner i in V(N\ j) for all $j \neq i$, or bids $\Delta_i=0$
3. Winner i receives $\pi_i>0$ from Threshold, or bids $\Delta_i=0$

Staged Approach.



High-Level Approach.

- Proxied:
  - *direct* but *incremental* value information.
- Threshold:
  - implement the Threshold rule in each stage
- Activity Rules:
  - consistent bounds across stages (relax by α?)
  - require progress across stages
- Staged w/ Final Round.
  - price-based feedback

Proxy Information

- Lower and upper valuation functions provided w/ appropriate bidding language
  - Maintain consistency, w/ $\overline{v}(S') \geq \overline{v}(S)$, $\forall S' \supseteq S$; $\underline{v}(S') \leq \underline{v}(S)$, $\forall S' \subseteq S$

- Incremental tightening of information allows early price discovery

Activity Rules.

- Consistency.
  - Can refine bounds on existing bundles
  - Can introduce new bundles (w/ bounds to respect free disposal)
- Progress:
  - tighten limits on allowed slack between bounds in later stages
  - limit # of additional bundles that can introduced in later stages
- At some point, move to a final stage.

In Each Stage...

- Compute “high” Threshold outcome w/ high bids and low asks
  - provides feedback in early stages
- Compute “low” Threshold outcome w/ low bids and high asks
  - provides feedback in later stages
- Finally implement this outcome

Buy-side: High item prices

- Compute high bid prices $\bar{p}_{bid,j}$ for items j based on high bids $\bar{v}_i(S)$
- Provide accurate winner feedback, suggest how far can drop price and still win.

$$\min_{p,\delta} \delta$$

$$\text{s.t. } \bar{v}_i(S') \geq \sum_{j \in S'} \bar{p}_{bid,j} \quad \forall \text{ winner } i, \text{ winner } S'$$

$$\bar{v}_i(S) \leq [\bar{v}_i(S') - \sum_{j \in S'} \bar{p}_{bid,j}] + \delta + \sum_{j \in S} \bar{p}_{bid,j} \quad \forall \text{ winner } i, \text{ loser } S$$

$$\bar{v}_i(S) \leq \delta + \sum_{j \in S} \bar{p}_{bid,j}, \quad \forall \text{ loser } i$$

(assumes an XOR bidding language, might also want to do *smoothing* across stages.)

Buy-side: Low item prices

- Compute low bid prices $\underline{p}_{bid,j}$ for items j based on low bids $\underline{v}_i(S)$
- Provide accurate loser feedback, suggest how far must increase price to win.

$$\min_{p,\delta} \delta$$

$$\text{s.t. } \underline{v}_i(S) \leq \sum_{j\in S} \underline{p}_{bid,j} \quad \forall \text{ loser } i$$

$$\underline{v}_i(S') \geq \sum_{j\in S'} \underline{p}_{bid,j} - \delta \quad \forall \text{ winner } i, \text{ winner } S'$$

$$\underline{v}_i(S) \leq [\underline{v}_i(S') - \sum_{j\in S'} \underline{p}_{bid,j}] + \delta + \sum_{j\in S} \underline{p}_{bid,j} \quad \forall \text{ winner } i, \text{ loser } S$$

Sell-side: Item prices

- Compute low ask prices $\underline{p}_{ask,j}$ to give winner feedback, suggest how far can increase price and still win
  - make these prices accurate for winners, with $\underline{v}_i(S') \leq \sum_{j \in S'} \underline{p}_{ask,j}$, $\forall$ winners (i, S')
- Compute high ask prices $\overline{p}_{ask,j}$ to give loser feedback, suggest how must drop price to win
  - make these prices accurate for losers, with $\overline{v}_i(S) \geq \sum_{j \in S} \overline{p}_{ask,j}$, $\forall$ losers i

Item Discovery.

- Also need buy-side prices for items offered on sell-side
  - perhaps $0.5(\underline{p}_{ask,j}+\overline{p}_{ask,j})$ is a good signal?
- Also need sell-side prices for items requested on buy-side
  - perhaps $0.5(\underline{p}_{buy,j}+\overline{p}_{buy,j})$ is a good signal?

Next steps (practical).

1

- Put together a computer-based simulation of this system.
- Implement simple bidding agents, check for bad behaviors, refine.

2

- Implement more sophisticated bidding agents, check for bad behaviors, refine.
- Work on computational properties, provide scalability.

3

- Run in an Experimental Economics Lab?

Conclusions.

- A combinatorial exchange can facilitate a "big bang" spectrum auction; allow incumbnents and new entrants to trade
- Key issues are:
  - computational
  - economic (bargaining problem)
  - preference elicitation
- Proposed a straw-model design, lots of interesting questions going forward!