Towards Iterative Combinatorial Exchanges

FCC Combinatorial Auction Workshop,
November 23rd, 2003

David C. Parkes
Harvard University.

Motivation

- Highly **fragmented** spectrum (frequency, control, and geography)
  - result of administrative allocation
- 2.5-2.7 GHz Spectrum
  - more total spectrum than cellular and PCS
  - 19@ 6MHz Instructional TV
  - 12 @ 6MHz MDS (wireless cable)
  - 493 Basic Trading Areas
- A “**big bang**” exchange:
  - make large amounts of spectrum (assigned & unassigned) available
  - improve allocative efficiency, take advantage of new technologies
Combinatorial Exchanges

- **Multiple** buyers and sellers, with **expressive bids**
  - e.g. “Buy {NYC, PHL, BOS} and sell {DC} for $1 million”

- FCC can also participate, **actively**:
  - e.g. “Convert ITFS licenses into wireless phone licenses”

- and **passively** (define aggregations):
  - e.g. “all contiguous 6 MHz blocks of spectrum in a BTA are equivalent”
  - may help computationally, and mitigate hold-out problem

Stylized Example.

Efficient trade: 1 and 2 sell, 3 buys.
Surplus $51.
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
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 $\pi^t$ to agent $i$ with $V^t(N) - V^t(N \setminus i) > 0$
- adjust $V^{t+1}(N)$, $V^{t+1}(N \setminus i)$ to $v^{t+1} = \max(0, v^t - \pi^t)$
- 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 $V(N \setminus i)$, 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**

Payoffs $\pi_i \geq 0$ to solve:

$$\min_{\pi} f(\pi)$$

s.t. $\sum_i \pi_i \leq V(N)$ (BB)

$$\pi_i \leq V(N) - V(N\setminus i), \quad \forall i$$ (*)

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

Note 1: $\pi_{VCG,i} = V(N) - V(N\setminus i)$

Note 2: (BB) and $\sum_{j \neq i} \pi_j \geq V(N\setminus i)$ (1-core) is equivalent to (*)

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

---

**VCG-Based Schemes**

Somewhat natural to consider VCG-based schemes, that divide surplus according to *ex post* $\pi_{VCG}$ ("payoff left on table")

- **Threshold.** Minimize worst-case $(\pi_{VCG,i} - \pi_i)$
- **Fractional.** Each agent gets $\pi_i = \mu \pi_{VCG,i}$
- **Large.** Allocate payoff in order $\pi_1, \pi_2, \pi_3, \ldots$
- **Reverse.**

**Threshold scheme** minimizes the maximal ex post regret across all agents.
Threshold Rule

- Implements a slight variation of 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 i.i.d. Uniform [0,1] values and costs
  - linear-strategy equilibrium; with $\hat{v} = (2/3)v + 1/12$ and $\hat{c} = (2/3)c + 1/4$ (Myerson & Satterthwaite, 83)
Experiments: ex ante BNE

- Consider a very 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:
  \[ \alpha^* = \arg \max_{\alpha} E_i E_{-i} [v_i(x^*(b)) - p_i(b)] \]

where \( x^*(b) \) is allocation given bids \( b \), price \( p_i(b) \) is payment by agent \( i \), and the expectation is taken w.r.t. distribution over types of agents.

Naive Approach

- Enumerate a payoff matrix, compute ex ante BNE

\[
\begin{array}{cccccccc}
\alpha_i & -0.5 & -0.48 & \ldots & 0 & \ldots & 0.12 & 0.14 & \ldots & 0.98 & 1.0 \\
\alpha_j & 1.5 & 1.4 & \ldots & 1.3 & \ldots & \\
\end{array}
\]

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

Choose a small set of strategies $A^0 = (\alpha_1, \ldots, \alpha_M) \in [a, b]^0$, for some initial search space.
Assume all agents except agent 1 play $\alpha_t$; initialize to $\alpha^0$.
Compute the BR, $\alpha^* \in A_t$, given average payoff to agent 1 and $\alpha^t$.
Move $\alpha^{t+1}$ towards $\alpha^*$; refine $[a, b]^{t+1}$ to focus search, select new $A^{t+1}$.

Details.

- new center: $\alpha^{t+1} = 1/3 \alpha^* + 2/3 \alpha^t$
- $|A| = 7$
- terminate when $\alpha^{t+1}$ is within 0.01 of $\alpha^t$
- new range: $[a, b]^{t+1}$ centered on $\alpha^{t+1}$
  - $(b-a)^{t+1} = 3/4 (b-a)^t$, if $\alpha^{t+1} \in [a, b]^t$
  - $(b-a)^{t+1} = 4(\alpha^{t+1} - \alpha^*)$, otherwise.
- Final validation step:
  - discretize $[a, b]^0$ to grid level 0.01, and check $\alpha^*$ is a BR to $\alpha^*_{-1}$.
Experimental Results

- 5 buyers, 5 sellers, 20 goods
- 10 bundles/agent.
  - Generate using Uniform (Sandholm'99); with value $U[0,1]$, same # items (drawn at random) in each bundle.
  - Assume XOR logic between bundles
- Winner determination performed by commercial MIP software.
- Evaluate over 500 instances.

- Speed-up: 1% accuracy in 2.5 CPU hours (c/f 2.5 days for enumeration)

Example 1- VCG payments
Example 2- No Discount

Example 3- Large
Example: Validity

Validating ex post Nash of Threshold rule

Main Results

Balanced allocation

Equilibrium

Efficiency
Large rule

- $\alpha^* \approx -0.08$
- Optimal strategy is to **overbid**, and win
  $$\pi_i = \pi_{VCG,i}$$
  - Implies that at least one participant has negative ex post payoff in BNE
- Unstable to high bids:
  - A buyer can always benefit from overstating her value if she knows she will win

NB. d’AGVA “expected Groves” mechanism is BB and **ex post** EFF, but only **ex ante** IR. (also needs an informed designer) (Arrow’79,d’Aspremont & Gerard-Varet’79)

Threshold Scheme

- Threshold is most efficient after Large (95% vs. 99%)
- Threshold rule is **more stable**:
  - Buyers cannot benefit from overstating values (quite general assumptions)
  - A buyer, $i$, receiving payoff division $\pi_i$, can only benefit from decreasing its bid (by less than $\pi_i$) when there is some $V(\bar{N} \setminus j), j \neq i$, without $i$.

- Also, reduces nicely to existing double auctions in special cases.
FCC: A Special Player

Can also apply core constraints for the FCC
\[ \pi_{\text{FCC}} + \sum_{i \in \mathcal{L}} \pi_i \geq V(\text{FCC} \cup \mathcal{L}), \forall \mathcal{L} \subseteq (\mathcal{N} \setminus \text{FCC}) \]

Nice alternative to involving the FCC as an active bidder.

⇒ the FCC cannot make more revenue simply by looking at the outcome

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

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
Preference Elicitation

- One-sided auctions
  - ascending-price auctions (linear prices, non-linear prices, non-linear & anonymous prices)
  - proxy agents (decouple best-response queries from user queries)
  - direct elicitation approaches (query-based, e.g. value queries, ordinal queries, etc.)

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
  - not present in one-sided auctions when VCG outcome in core.
High-Level Approach.

- **Proxied**
  - users make direct but incremental statements about valuations for different bundles.

- **Threshold-based.**
  - solve WD to maximize reported surplus, and implement the Threshold payoff-division rule

- **Activity Rules.**
  - consistency: incremental value information must not contradict with earlier information.
  - require “progress” across stages.

- **Staged w/ Final Round.**
  - price-based feedback

---

Information Required.

Consider information:
\[ v'(S) = \max(0, v(S) - \pi) \]

Can compute VCG, if:
1. Complete info from all losers
2. high enough bids from winner i to place them in \( V'(N\setminus j) \), for all \( j \neq i \), or complete info if not possible for any \( \pi > 0 \).

For Threshold, also need:
3. high enough bids from winner i to gain some payoff division, or complete info if not possible for any \( \pi > 0 \)
Proxy Information: Upper & Lower Bounds

- Agents provide bounds on values of bundles
  - \( v_i(S) \in [b_i(S), \bar{b}_i(S)] \)
  - Can be via a compact bidding language ("upper-bid" and "lower-bid")
  - Maintain consistency, \( b(S') \geq b(S), \forall S' \supseteq S; b(S') \leq b(S), \forall S' \subseteq S \)
- Refine bounds between stages, and introduce new bundles
  - "Relaxed→Tight" information allows early price discovery

Activity Rule (Flavors).

- Consistency:
  - can refine bounds on existing bundles
  - new bounds within existing bounds
- 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 **Threshold outcome** w/ high bids and low asks
  - “high” outcome
  - provides feedback in early stages

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

Price-Feedback/ Buy-side

- Compute **high bid prices** $\bar{p}_{bid,j}$ for items $j$ based on high bids $\bar{b}_i(S)$

\[
\begin{align*}
\min_{p,0} & \delta \\
\text{s.t.} & \quad \bar{b}(S') \geq \sum_{j \in S} \bar{p}_{bid,j} \quad \forall \text{ winner } i, \text{ winner } S' \\
& \quad \bar{b}_i(S) \leq [\bar{b}(S') - \sum_{j \in S} \bar{p}_{bid,j}] + \delta + \sum_{j \in S} \bar{p}_{bid,j} \quad \forall \text{ winner } i, \text{ loser } S \\
& \quad \bar{b}_i(S) \leq \delta + \sum_{j \in S} \bar{p}_{bid,j} \quad \forall \text{ loser } i
\end{align*}
\]

(assumes an XOR bidding language, use Kohlberg (72) iterative scheme to define unique solution, might also want to do smoothing across stages.)

- Provide accurate **winner feedback**, suggest how far can drop price and still win.
Price-Feedback/ Buy-side

- Compute **low bid prices** $p_{bid,j}$ for items $j$ based on low bids $b_i(S)$

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

s.t. $b_i(S) \leq \sum_{j \in S} p_{bid,j} \quad \forall \text{ loser } i$

$b_i(S') \geq \sum_{j \in S} p_{bid,j} - \delta \quad \forall \text{ winner } i, \text{ winner } S'$

$b_i(S) \leq [b_i(S') - \sum_{j \in S} p_{bid,j}] + \delta + \sum_{j \in S} p_{bid,j} \quad \forall \text{ winner } i, \text{ loser } S$

- Provide accurate **loser feedback**, suggest how far must increase price to win.

11/23/2003

Combinatorial Exchanges

---

Price Feedback/ Sell-side

- Compute **low ask prices** $p_{ask,j}$ to give winner feedback, suggest how far can increase price and still win
  - make these prices accurate for winners, with $b_i(S') \leq \sum_{j \in S} p_{ask,j} \quad \forall \text{ winners } (i, S')$

- Compute **high ask prices** $p_{ask,j}$ to give loser feedback, suggest how must drop price to win
  - make these prices accurate for losers, with $\bar{b}_i(S) \geq \sum_{j \in S} p_{bid,j} \quad \forall \text{ losers } i$

11/23/2003

Combinatorial Exchanges
Item Discovery.

- Also need buy-side prices for items offered on sell-side
  - perhaps $0.5(p_{ask,j} + p_{ask,j})$ is a good signal?

- Also need sell-side prices for items requested on buy-side
  - perhaps $0.5(p_{buy,j} + p_{buy,j})$ is a good signal?

Next steps.

1. Put together a computer-based simulation of this system.
2. Implement simple bidding agents, check for bad behaviors, refine.
3. Implement more sophisticated bidding agents, check for bad behaviors, refine.
4. Work on computational properties, provide scalability.
5. Run in an Experimental Economics Lab?
Conclusions.

- A combinatorial exchange can facilitate a “big bang” spectrum auction; allow incumbents 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!