The opinions expressed in this talk are those of the authors and do not necessarily represent the views of the FCC or any of its staff.

Issues in FCC Package Bidding Auction Design

FCC Wye River Conference III

Karla Hoffman

Joint work with

Melissa Dunford, Dinesh Menon, Rudy Sultana,Thomas Wilson

Decisive Analytics Corporation

(under contract with Computech, Inc.)

November 22, 2003

Outline of Talk

- Examine ISAS auction design
  - No provision for "last and best"
  - Is chosen linear pricing algorithm most appropriate?
  - Communication complexity
- Consider using ascending proxy as final "round"
  - Address computational issues
  - Design of *accelerated* proxy mechanism
- Test alternative linear pricing approaches
  - Used accelerated proxy mechanism to benchmark linear pricing algorithms
- Bidder aid tools

Positives of the ISAS Auction Design

- Price discovery
- Package creation
- No budget exposure problem (XOR)
- Linear pricing
  - Perceived as fair
  - Easy to use
  - Reduces parking problem
- Transparency

Open Issues with the ISAS Auction Design

- May require large increment size to close in reasonable time
- No provision for "last and best"
- Limited testing of linear pricing scheme
- Bidders must determine what packages to create and bid
- Rules may seem complex to bidder
- Treats every item as unique
  - Better to have quantity specification for homogeneous items
- Opportunity for gaming

Outline of Talk

- Examine ISAS auction design
  - No provision for "last and best"
  - Is chosen linear pricing algorithm most appropriate?
  - Communication complexity
- Consider using ascending proxy as final "round"
  - Address computational issues
  - Design of *accelerated* proxy mechanism
- Test alternative linear pricing approaches
  - Used accelerated proxy mechanism to benchmark linear pricing algorithms
- Bidder aid tools

Economic Characteristics of Ascending Proxy

- Guaranteed to arrive at efficient outcome
- When buyer sub-modularity property holds, mechanism arrives at VCG prices
- Even when buyer sub-modularity property does not hold, prices are in the core
- Collusion and other destructive bidding eliminated since bidders forced (through proxy) to bid straightforwardly

Ascending Proxy Mechanism

- Each bidder provides all packages of interest to proxy with valuations
- Bidder can only win one of the packages submitted (XOR among packages of bidder)
- Proxy bids for bidder in myopic best-response manner
- Auctioneer solves WDP to determine provisionally-winning bids
- If bid is non-winning, then price goes up by epsilon
- Proxy agents place bids until no bids are profitable or winning
- Auction ends when no new bids are placed in a round
- At end of auction, winning bidders pay what they bid

Proxies Place Bids

- A bidder's proxy follows a "Myopic Best Response" strategy
  - Myopic because the proxy only looks at the current prices
  - Best response refers to profit maximizing
    - *Profit = Value – Price*
- In a round, a proxy submits the bidder's most profitable package at the current price
  - If ties exist, all ties are submitted
  - If a bidder has a current provisionally winning bid the proxy does not place any new bids (since all non-winning bids of that bidder are not as profitable as the winning bid)

Proxy Rounds

- Simulation of a Proxy Auction with 6 licenses and 10 bidders
  - Most bidders entered many packages ~ 30-40 packages (out of possible 63)
  - Value of the auction ~ $3.4 million
- Results:
  - With $5000 increment, over 22,000 rounds
  - With $10 increment, over 9 million rounds!
- Auction theory requires very small increment
- But, FCC needs an auction design that can handle thousands of items

Is there a way to overcome this computational stumbling block?

Accelerated Proxy Mechanism

- Reduces substantially the number of rounds of the proxy mechanism
- Works backwards from “end result” and thereby requires far fewer iterations than proxy mechanism
- Same nice properties as Ausubel-Milgrom proxy auction

Accelerated Proxy: Methodology

STEP 1: Solve Winner Determination Problem for Efficient Outcome

(Objective function coefficients are valuations)

- **Determines winning bidders**
- **Determines winning bids of winning bidders**

STEP 2: Determine the Opening Prices for All Bids of All Bidders

a. **Opening prices of non-winning bidders' bids = valuations**

b. **Opening prices of winning bids of winning bidders = *"Safe Price"***

***Safe Price* = Max of all valuations on this package by non-winning bidders**

Opening Price (Winning Bid) = Safe Price

c. **All opening prices of all *losing bids* of winning bidder have same profitability**

Profit (Winning Bid) = Valuation (Winning Bid) - Opening Price (Winning Bid)

Opening Price (Non-Winning Bid) = Valuation (Non-Winning Bid) - Profit (Winning Bid)

STEP 3: Use Increment Scaling Method to Determine Optimum Prices

Accelerated Proxy: Increment Scaling

FIRST STAGE: Set increment size to some large increment (scale all opening prices down to the nearest increment, but not less than zero)

- Implement Proxy Mechanism until auction ends with no new bids

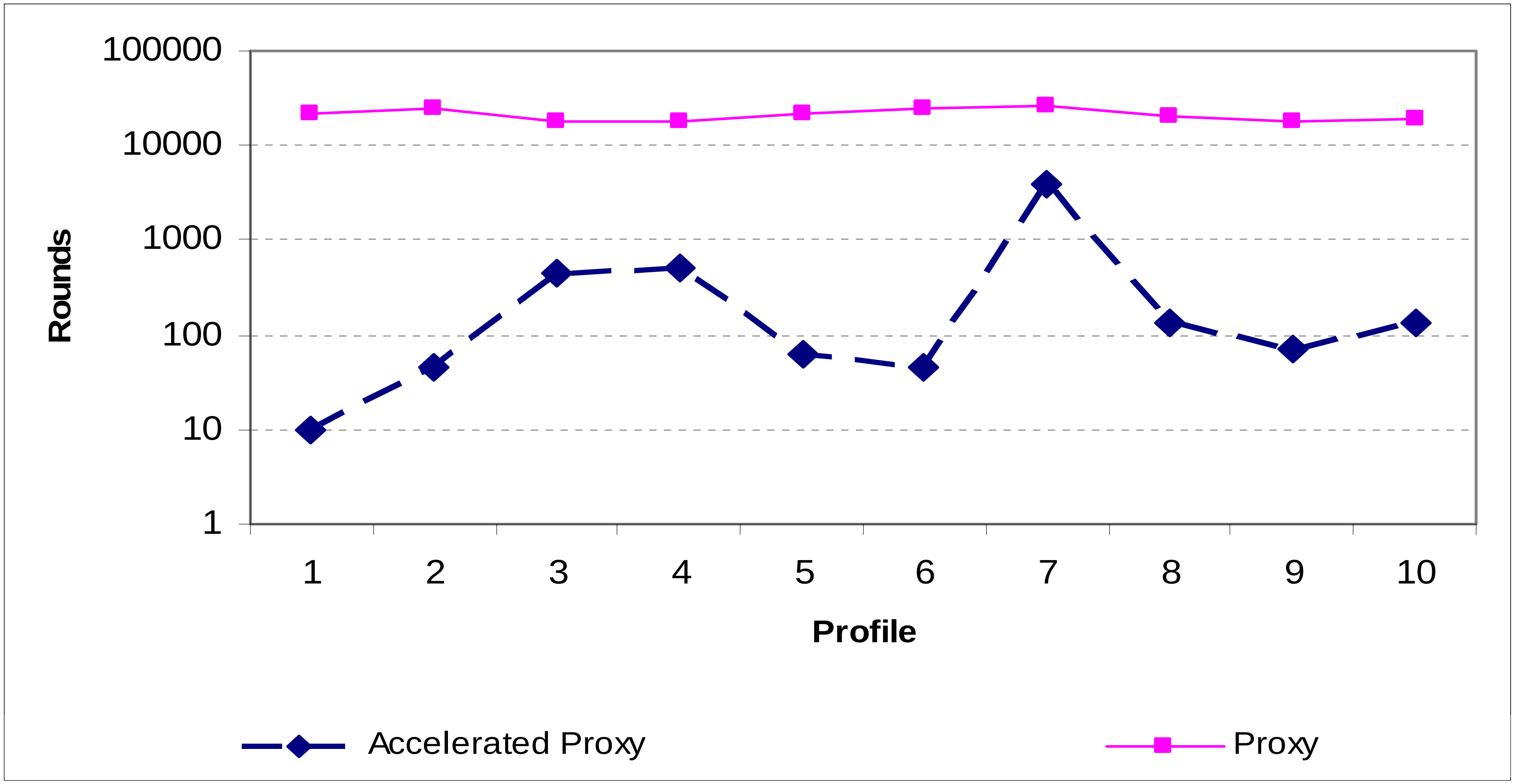
EVERY SUBSEQUENT STAGE:

- Given final outcome from prior stage, check if the current increment satisfies the *"**increment threshold**"*
  - If threshold met **STOP**, *ELSE*:
- Determine starting point for the next stage
  - Every winning agent's price vector is set equal to their final bid amounts from the previous stage less the amount of the current increment. Every non-winning agent's price vector is set equal to their prior bid amounts
- Scale down the current increment by a factor of 10 and start the next stage
- **NOTE:** May need *"Corrective Rollback"*

Properties of *Accelerated* Proxy

- Efficient Outcome
- Buyer Pareto-optimal payments by winners when the "***agents-are-substitutes***" property holds
- Buyer Pareto-optimal payments even when the "***buyer sub-modularity***" property does not hold
- Forces straight-forward bidding and therefore removes opportunity for shill bidding and collusion
- Requires far fewer integer optimizations than a direct application of the ascending proxy auction
  - Bounded by a function of number of digits of accuracy required, number of packages in the optimal allocation and number of bids by winning bidders
- ***Obtains core outcome when agents-are-substitutes property does not hold***

Rounds: Proxy vs. Accelerated Proxy



- Accelerated proxy achieves efficient outcomes with bidder payments accurate to 1 cent
- Proxy accurate to within $5,000

Outline of Talk

- Examine ISAS auction design
  - No provision for "last and best"
  - Is chosen linear pricing algorithm most appropriate?
  - Communication complexity
- Consider using ascending proxy as final "round"
  - Address computational issues
  - Design of *accelerated* proxy mechanism
- Test alternative linear pricing approaches
  - Used accelerated proxy mechanism to benchmark linear pricing algorithms
- Bidder aid tools

Testing Linear Pricing against Proxy

- Created a number of small test cases and 10 larger profiles
  - 6 items, 10 bidders, approx. $3M revenue
- Tested:
  - Ausubel-Milgrom Ascending Proxy
  - Accelerated Proxy
  - Three Linear Pricing Algorithms (with myopic best response bidding and fixed increments)
- Compare:
  - Outcomes (efficiency)
  - Payments
  - Speed of auction

Pricing Algorithms

- RAD (DeMartini, Kwasnica, Ledyard and Porter)
- Smoothed Anchoring (FCC)
- Smoothed Nucleolus
  - RAD first stage
  - Smoothing second stage

Test Case 1: Agents Are Substitutes

Agent	1	2	3	4	5
Package	AB	BC	C	C*	AB*
Value	21	35	14	20	22

Method	Increment	Rounds	Revenue	Payments by winning agents	
				A4, {C}	A5, {AB}
Accelerated Proxy	0.01	6	35	14	21
Proxy	0.1	403	36.9	15.8	21.1
Smoothed Anchoring	0.1	298	35.05	13.99	21.06
Smoothed Nucleolus	0.1	298	35.05	13.99	21.06
RAD	0.1	291	35.02	14.03	20.99
VCG	-	-	35	14	21

Buyer sub-modularity fails

Test Case 2: Agents Are Not Substitutes

Agent	1	2	3	4
Package	AB	BC*	AC	A*
Value	20	26	24	16

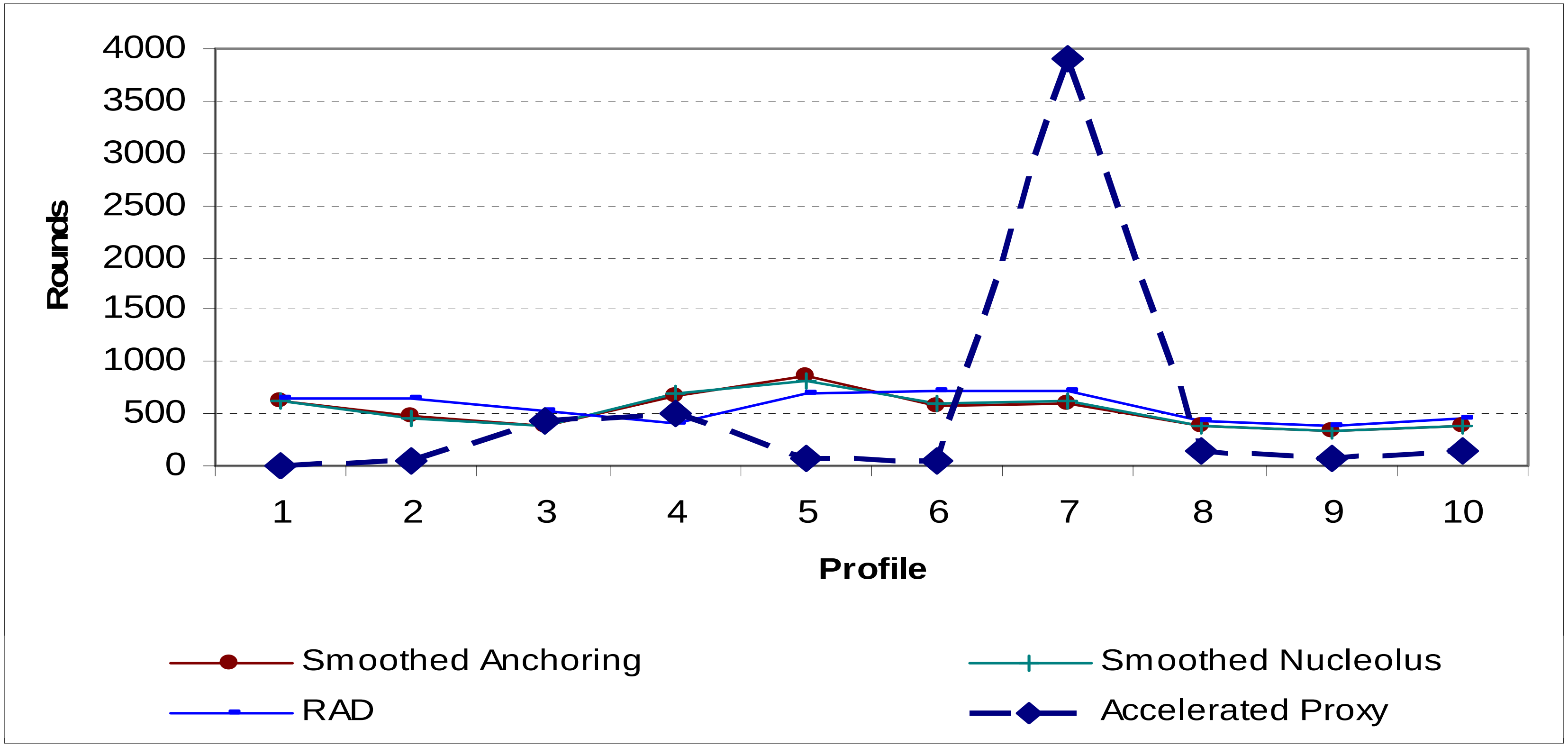
Method	Increment	Rounds	Revenue	Payments by winning agents	
				A2, {BC}	A4, {A}
Accelerated Proxy	0.01	16	24	17	7
Proxy	0.1	311	24.2	12.1	12.1
Smoothed Anchoring	0.1	234	24.33	12.19	12.14
Smoothed Nucleolus	0.1	234	24.33	12.19	12.14
RAD	0.1	257	23.95	8.3	15.65
VCG	-	-	8	8	0

Summary of 10 profiles

$5000 increment, 6 items, 10 bidders, $3M auction

Profile	Number of Winning Packages	Agents are Substitutes?	Efficient Result?	Revenue within tolerance ($5,000)			
				Proxy	RAD	Smoothed Nucleolus	Smoothed Anchoring
1	1	YES	All methods	YES	YES	YES	YES
2	2	NO	RAD only	YES	YES	YES	YES
3	4	YES	All methods	YES	$23,000	$16,000	$13,000
4	3	NO	None	YES	$7,000	YES	YES
5	2	NO	All but Proxy	YES	YES	YES	$15,000
6	2	NO	All but RAD	YES	$10,000	YES	YES
7	2	YES	All methods	$7,000	$6,000	YES	$8,000
8	3	YES	All methods	$13,000	YES	$8,000	YES
9	4	YES	RAD only	$8,000	YES	YES	YES
10	4	YES	None	YES	YES	$10,000	YES

Rounds: Accelerated Proxy vs. Linear Pricing



- Accelerated proxy achieves efficient outcomes with bidder payments accurate to 1 cent
- Linear pricing schemes use an increment of $5,000

Average Performance of the Pricing Schemes

Method	Average Number of Rounds (Increment Size: $5000)	Abs. Revenue Deviation from Accelerated Proxy Revenue ($)			Abs. Price Deviation from Accelerated Proxy Price ($)		
		Mean	Median	Max.	Mean	Median	Max.
Proxy	21,260	4,551	3,683	12,825	3,192	2,878	5,536
Smoothed Anchoring	526	4,828	2,483	14,949	4,152	3,194	16,635
Smoothed Nucleolus	527	4,539	2,161	16,330	3,283	2,170	16,561
RAD	562	5,446	2,508	22,799	2,964	2,108	16,482

Accelerated Proxy: 537 rounds on average for accuracy to **1 cent**

Conclusions of Testing

- Linear pricing arrives at outcomes similar to that of ascending proxy when increment the same, except when synergies are very large
- No linear pricing algorithm dominates all others
- With linear pricing, need some type of smoothing to overcome fluctuations
- *Accelerated* ascending proxy much faster than any other approach for same accuracy

Pros and Cons of Accelerated Proxy

- Pros:
  - Efficient
  - Core Outcome
  - No Gaming
  - Limits bidder participation burden
  - Computationally competitive for greater accuracy
  - Verifiability possible without disclosing valuations
- Cons:
  - Bidders must provide valuations
  - Language (Puts burden on bidder)
    - SOLUTION: Bidder aid tools
  - No Feedback (Price discovery)
    - SOLUTION: Hybrid designs

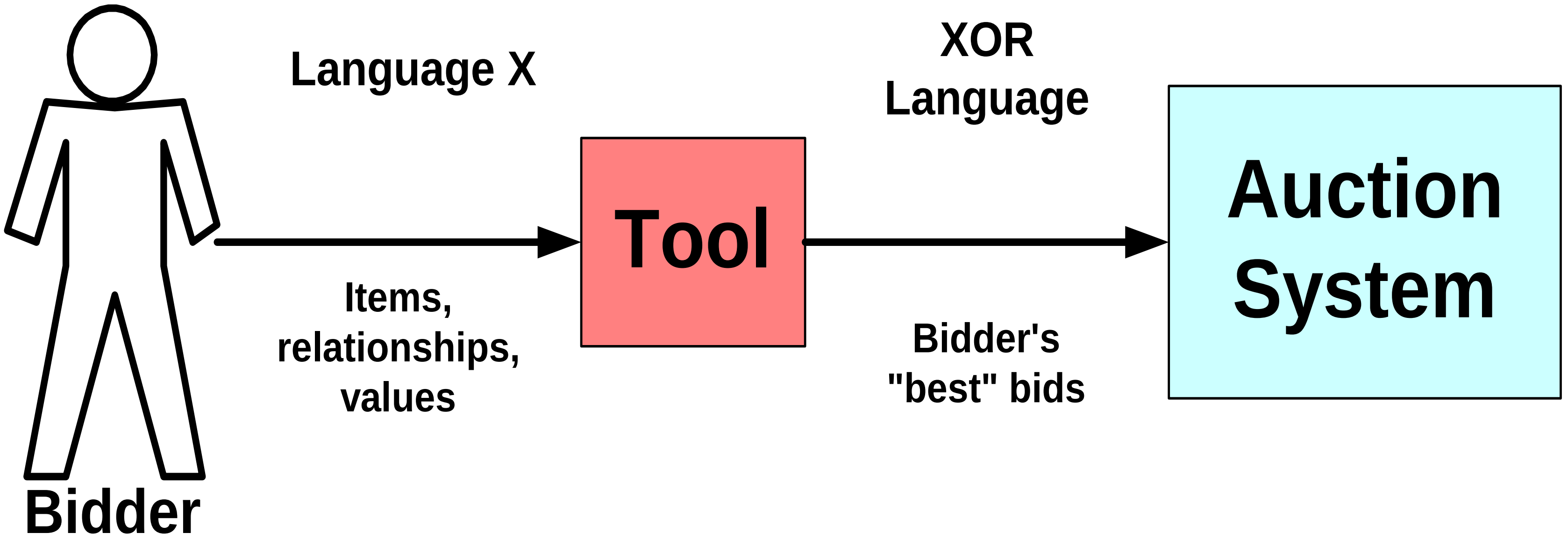
Outline of Talk

- Examine ISAS auction design
  - No provision for "last and best"
  - Is chosen linear pricing algorithm most appropriate?
  - Communication complexity
- Consider using ascending proxy as final "round"
  - Address computational issues
  - Design of *accelerated* proxy mechanism
- Test alternative linear pricing approaches
  - Used accelerated proxy mechanism to benchmark linear pricing algorithms
- Bidder aid tools

A Need for Bidder-Aid Tools

- How does the bidder express his business plans in a compact way?
- How does one create packages that reflect business needs?
- How does one alter business plans based on price discovery?

Bidder-Aid Tool Concept



Example of Bidding Language: Cramton

- **Items in a given class are in terms of $/MHz-pop**
  - May want more than one class: (e.g. Large cities, small cities, rural areas)
- **Equivalence classes**
  - A minimum amount of MHz needed
  - A value (above norm) for certain bands
  - A bonus for blocks that are contiguous
  - Incremental vales for each increment above the minimum required
- **Minimum and maximum amounts of total population needed**
- **Budget constraints (Possibly more than one)**
- **Secondary items:**
  - Contingent items (only want A if coupled with B)
  - Synergy (Want A with stand-alone value; but if with B, A gets synergy value)

The Language is translated into an optimization problem that determines the “best” packages for this bidder given budget, current prices, and activity rules

Generating Proposals: Example of Optimization

$$\textbf{Maximize}_{x} \sum_{b \in B} (v_b - p_b) x_b$$

$$Subject\ to:$$

$$\sum_{j \in B^l} x_j \leq 1 \quad for\ all\ items\ l$$

$$\sum_{i \in B^c} x_i \leq \left| B^c \right| x_c \quad for\ all\ contingent\ sets\ "c"$$

$$\sum_{c=1}^{\#Con.Sets} x_c \leq 1$$

$$\sum_{b \in B} BidAmt_b x_b \leq Budget$$

$$x_b \in \{0,1\} \quad for\ all\ bids$$

Conclusions

- Linear pricing with smoothing works well
- Further work on bidder aid tools is needed
- Other issues with ISAS design
  - Opportunity for gaming (signaling)
  - XOR bidding language forces explosion of bids for homogeneous items
  - Lots of bidder participation during auction
- Can other hybrid designs overcome these issues?
  - Clock Auction followed by Proxy
  - Iterative Proxy
- Issues with hybrid designs:
  - Activity rules
  - Information to bidders
  - What information passes between stages

Package Bidding: Bidders' Needs

- Easy to understand rules
- Easy to express needs
- Easy to interpret results
- Fair
- Reasonable completion time
- Price discovery
- Risk/Exposure not excessive
- Ability to compete effectively

Package Bidding: FCC Perspective

- *Efficiency* – Spectrum will be used
- *Transparency* – No security issues
- *Fairness* – Spectrum not held hostage to law suits
- *Speed* – Spectrum is allocated quickly
- *Participation/Competition* – Buyers come to auction

QUESTIONS?

Properties: AAS and BSM

Agents-Are-Substitutes (AAS) if:

$$w(A) - w(A \setminus K) \geq \sum_{a \in K} [w(A) - w(A \setminus a)], \quad \forall K \subset A, \ 0 \notin K$$

- VCG payoffs are supported in the core only when AAS condition is satisfied

Buyer Sub-Modularity (BSM) if:

$$w(L) - w(L \setminus K) \geq \sum_{a \in K} [w(L) - w(L \setminus a)], \quad \forall K \subset L, \ 0 \notin K,$$

$$\textit{for all} \ L \subseteq A, \ 0 \in L$$

- For all sub-coalitions, the incremental value of an additional member is decreasing in the coalition size
- BSM is a stronger condition