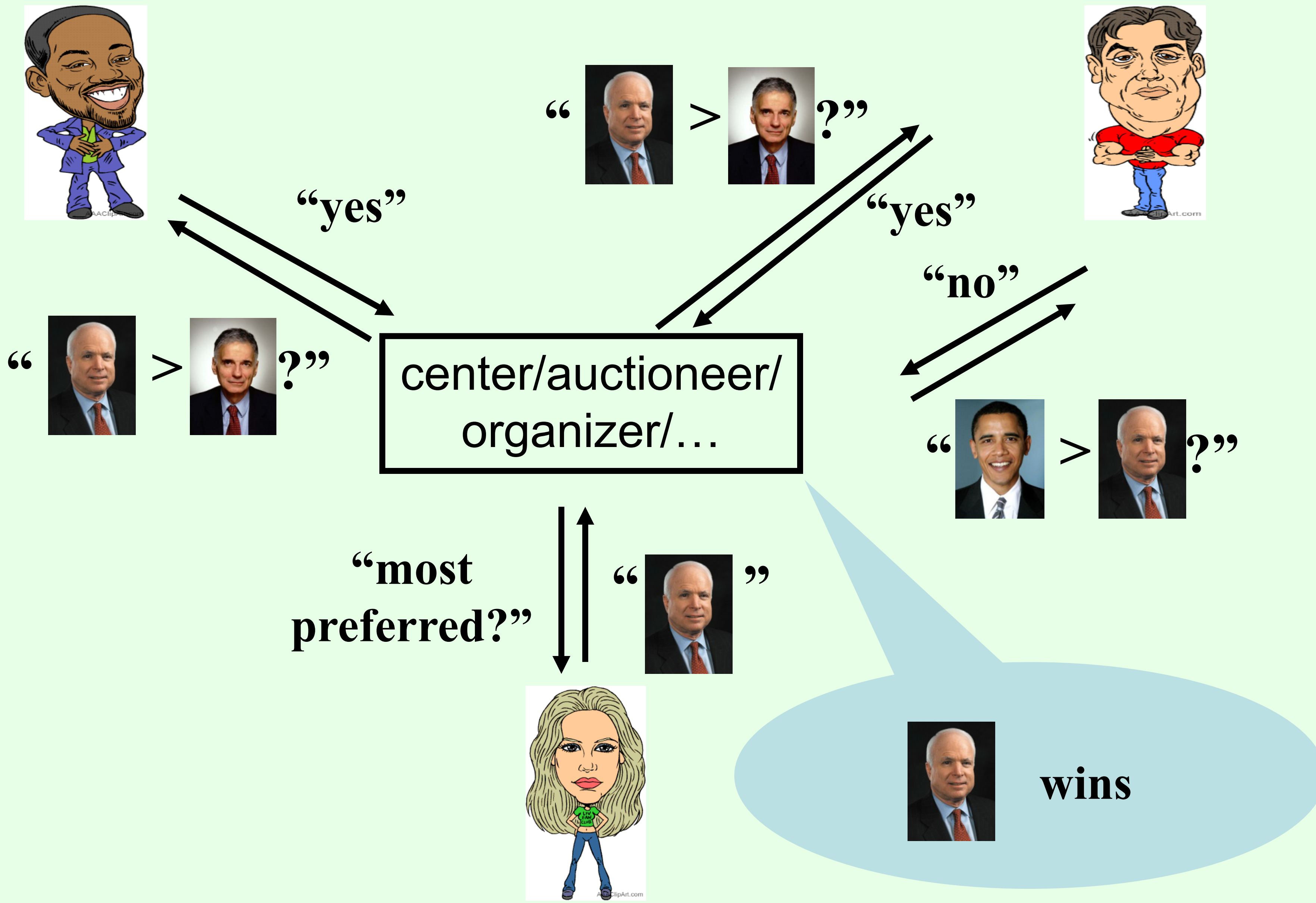
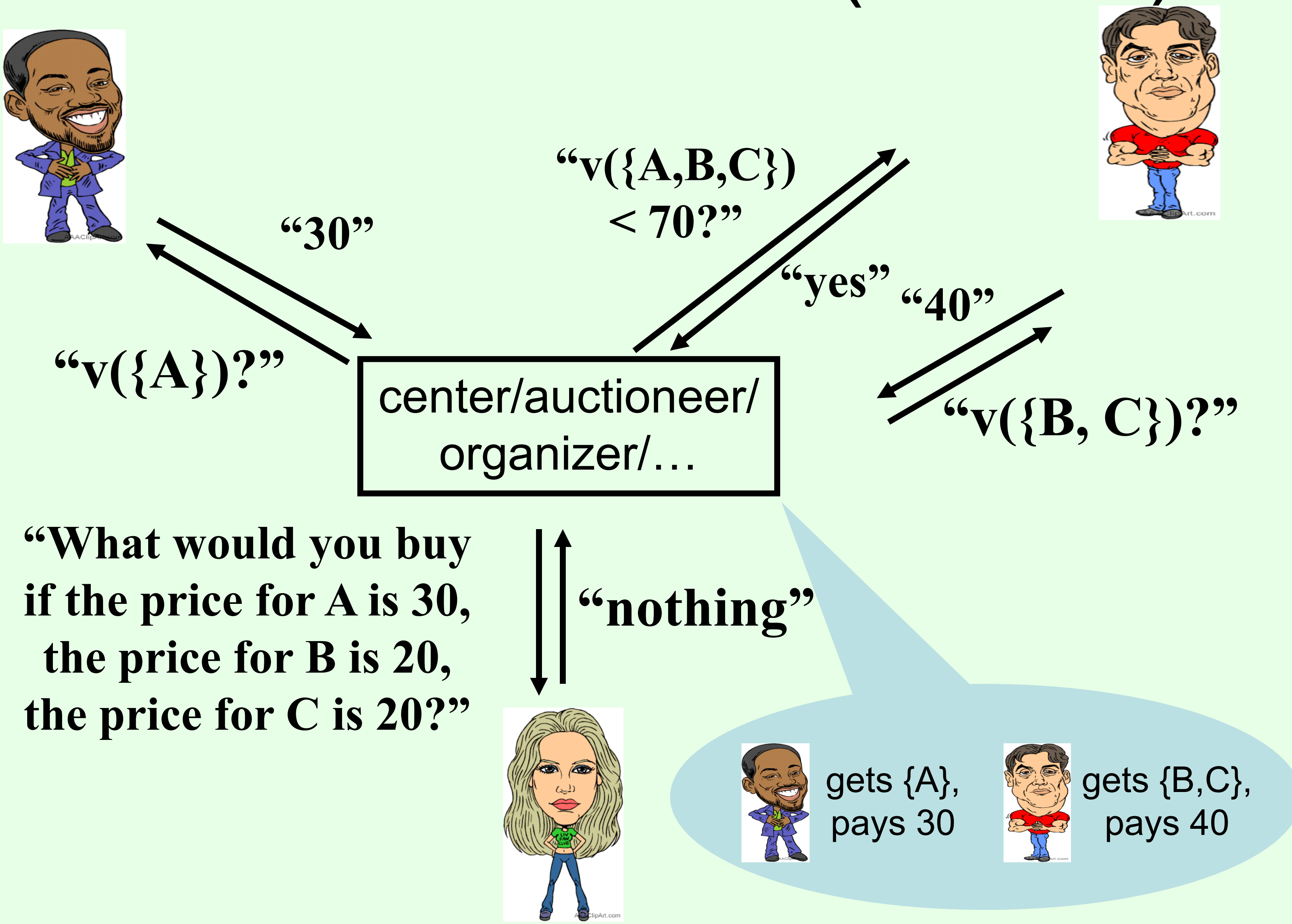
CPS 296.1
Preference elicitation/ iterative mechanisms

Vincent Conitzer
conitzer@cs.duke.edu

Preference elicitation (elections)



Preference elicitation (auction)



Unnecessary communication

- We have seen that mechanisms often force agents to communicate large amounts of information
  - E.g., in combinatorial auctions, should in principle communicate a value for every single bundle!
- Much of this information will be irrelevant, e.g.:
  - Suppose each item has already received a bid >$1
  - Bidder 1 values the grand bundle of all items at $v_1(I) = \$1$
  - To find the optimal allocation, we need not know anything more about 1's valuation function (assuming free disposal)
  - We may still need more detail on 1's valuation function to compute Clarke payments…
  - … but not if each item has received two bids >$1
- Can we spare bidder 1 the burden of communicating (and figuring out) her whole valuation function?

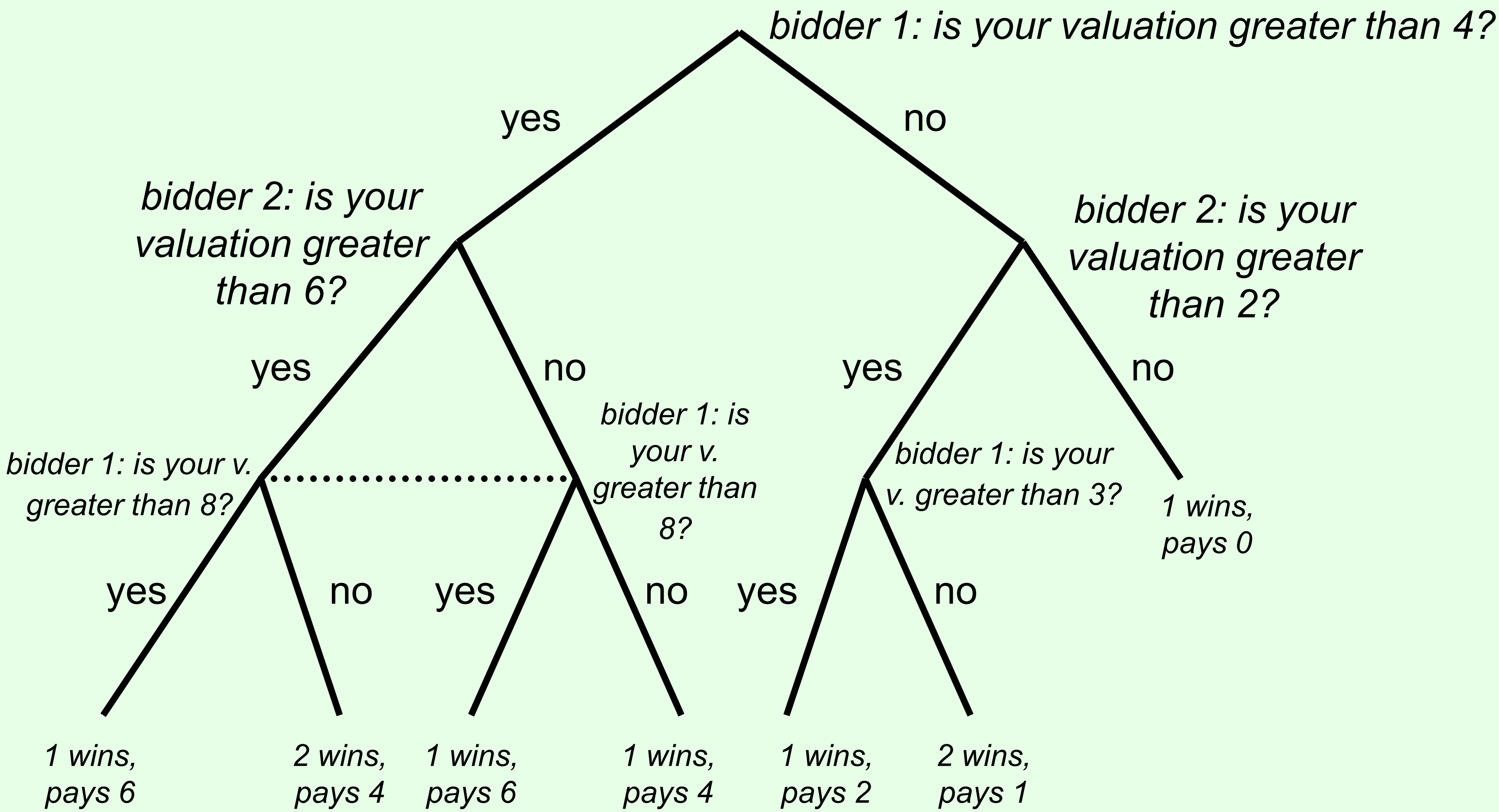
Single-stage mechanisms

- If all agents must report their valuations (types) at the same time (e.g., sealed-bid), then almost no communication can be saved
  - E.g., if we do not know that other bidders have already placed high bids on items, we may need to know more about bidder 1's valuation function
  - Can only save communication of information that is irrelevant regardless of what other agents report
    - E.g. if a bidder's valuation is below the reserve price, it does not matter exactly where below the reserve price it is
    - E.g. a voter's second-highest candidate under plurality rule
- Could still try to design the mechanism so that most information is (unconditionally) irrelevant
  - E.g. [Hyafil & Boutilier IJCAI 07]

Multistage mechanisms

- In a multistage (or iterative) mechanism,
  - bidders communicate something,
  - then find out something about what others communicated,
  - then communicate again, etc.
- After enough information has been communicated, the mechanism declares an outcome
- What multistage mechanisms have we seen already?

A (strange) example multistage auction



- Can choose to hide information from agents, but only insofar as it is not implied by queries we ask of them

Converting single-stage to multistage

- One possibility: start with a single-stage mechanism (mapping o from Θ_1 x Θ_2 x … x Θ_n to O)
- Center asks the agents queries about their types
  - E.g., "Is your valuation greater than v?"
  - May or may not (explicitly) reveal results of queries to others
- Until center knows enough about $\theta_1, \theta_2, \ldots, \theta_n$ to determine $o(\theta_1, \theta_2, \ldots, \theta_n)$
- The center's strategy for asking queries is an elicitation algorithm for computing o
- E.g., Japanese auction is an elicitation algorithm for the second-price auction

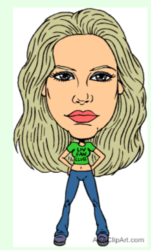
Elicitation algorithms

- Suppose agents always answer truthfully
- Design elicitation algorithm to minimize queries for given rule
- What is a good elicitation algorithm for STV?
- What about Bucklin?

An elicitation algorithm for the Bucklin voting rule based on binary search

[Conitzer & Sandholm 05]

- Alternatives: A B C D E F G H

- Top 4? {A B C D} {A B F G} {A C E H}
- Top 2? {A D} {B F} {C H}
- Top 3? {A C D} {B F G} {C E H}

Total communication is nm + nm/2 + nm/4 + … ≤ 2nm bits
(n number of voters, m number of candidates)

Funky strategic phenomena in multistage mechanisms

- Suppose we sell two items A and B in parallel English auctions to bidders 1 and 2
  - Minimum bid increment of 1
- No complementarity/substitutability
- $v_1(A) = 30$, $v_1(B) = 20$, $v_2(A) = 20$, $v_2(B) = 30$, all of this is common knowledge
- 1's strategy: "I will bid 1 on B and 0 on A, unless 2 starts bidding on B, in which case I will bid up to my true valuations for both."
- 2's strategy: "I will bid 1 on A and 0 on B, unless 1 starts bidding on A, in which case I will bid up to my true valuations for both."
- This is an equilibrium!
  - Inefficient allocation
  - Self-enforcing collusion
  - Bidding truthfully (up to true valuation) is not a dominant strategy

Ex-post equilibrium

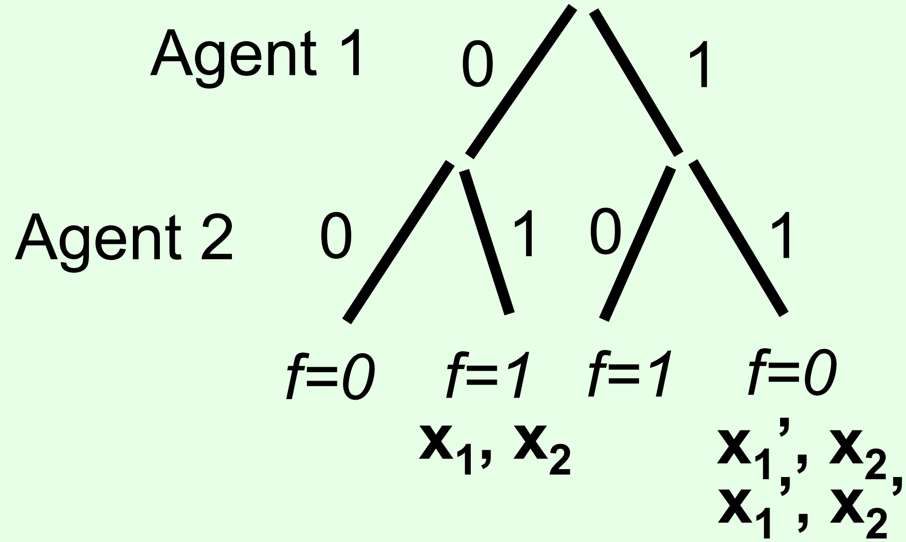
- In a Bayesian game, a profile of strategies is an ex-post equilibrium if for each agent, following the strategy is optimal for every vector of types (given the others' strategies)
  - That is, even if you are told what everyone's type was after the fact, you never regret what you did
  - Stronger than Bayes-Nash equilibrium
  - Weaker than dominant-strategies equilibrium
    - Although, single-stage mechanisms are ex-post incentive compatible if and only if they are dominant-strategies incentive compatible
- If a single-stage mechanism is dominant-strategies incentive-compatible, then any elicitation protocol for it (any corresponding multistage mechanism) will be ex-post incentive compatible
- E.g., if we elicit enough information to determine the Clarke payments, telling the truth will be an ex-post equilibrium (but not dominant strategies)

Lower bounds on communication

- Communication complexity theory can be used to show lower bounds
  - "Any elicitation algorithm for rule r requires communication of at least N bits (in the worst case)"
- Voting [Conitzer & Sandholm 05]
  - Bucklin requires at least on the order of nm bits
  - STV requires at least on the order of n log m bits
    - Natural algorithm uses on the order of $n(\log m)^2$ bits
- Combinatorial auction winner determination requires exponentially many bits [Nisan & Segal 06]
  - … unless only a limited set of valuation functions is allowed

How do we know that we have found the best elicitation protocol for a mechanism?

- Communication complexity theory: agent i holds input x_i, agents must communicate enough information to compute some $f(x_1, x_2, \ldots, x_n)$
- Consider the tree of all possible communications:
- Every input vector goes to some leaf



- If $x_1, \ldots, x_n$ goes to same leaf as $x_1', \ldots, x_n'$ then so must any mix of them (e.g., $x_1, x_2', x_3, \ldots, x_n'$)
- Only possible if f is same in all 2^n cases

Example on board: finding which valuation is higher (or tie)

- Suppose we have a fooling set of t input vectors that all give the same function value f_0, but for any two of them, there is a mix that gives a different value
- Then all vectors must go to different leaves ⇒ tree depth must be ≥ log(t)
- Also lower bound on nondeterministic communication complexity
  - With false positives or negatives allowed, depending on f_0

Combinatorial auction WDP requires exponential communication [Nisan & Segal JET 06]

- … even with two bidders!
- Let us construct a fooling set
- Consider valuation functions with
  - $v(S) = 0$ for $|S| < m/2$
  - $v(S) = 1$ for $|S| > m/2$
  - $v(S) = 0$ or 1 for $|S| = m/2$
- If m is even, there are 2^(m choose m/2) such valuation functions (doubly exponential)
- In the fooling set, bidder 1 will have one such valuation function, and bidder 2 will have the dual such valuation function, that is, $v_2(S) = 1 - v_1(I \setminus S)$
- Best allocation gives total value of 1
- However, now suppose we take distinct (v_1, v_2), (v_1', v_2')
- WLOG there must be some set S such that $v_1(S) = 1$ and $v_1'(S) = 0$ (hence $v_2'(I \setminus S) = 1$)
- So on (v_1, v_2') we can get a total allocation value of 2!

iBundle: an ascending CA [Parkes & Ungar 00]

- Each round, each bidder i faces separate price $p_i(S)$ for each bundle S
  - Note: different bidders may face different prices for the same bundle
  - Prices start at 0
- A bidder (is assumed to) bid $p_i(S)$ on the bundle(s) S that maximize(s) her utility given the current prices, i.e., that maximize(s) $v_i(S) - p_i(S)$ (straightforward bidding)
  - Bidder drops out if all bundles would give negative utility
- Winner determination problem is solved with these bids
- If some (active) bidder i did not win anything, that bidder's prices are increased by ε on each of the bundles that she bid on (and supersets thereof), and we go to the next round
- Otherwise, we terminate with this allocation & these prices

Restricted valuations

- For (e.g.) combinatorial auctions, if we know that agents' valuation functions lie in a restricted class of functions, then they may be easy to elicit
- E.g. if we know that an agent's valuation function is an OR of bundles of size at most 2, then all we need to ask a bidder for is his value of each bundle of size at most 2, to know the entire function
  - $O(m^2)$ queries
  - So-called value queries
- Which classes of valuations can we elicit using only polynomially many queries?
  - … and what types of queries do we need?

- Closely related to query learning in machine learning

Restricted valuations…

- Various restricted classes can be elicited using polynomially many value queries
  - Read-once & toolbox valuations [Zinkevich, Blum, Sandholm EC 03]
  - Valuations with limited item interdependency [Conitzer, Sandholm, Santi AAAI 05]
- Other classes inherently require other types of query
- E.g., demand query: “Which bundle would you buy given prices p(S) on bundles?”
  - Could also just have prices on items
  - Compare iBundle ascending CA
- A value query can be simulated using polynomially many demand queries (even just with item prices), but not vice versa [Blumrosen & Nisan EC 05]
- Using (bundle-price) demand queries, XOR valuations can be elicited using O(m^2 #terms) queries [Lahaie & Parkes EC 04]
- … but if only item-price demand queries (and value queries) are allowed, exponentially many queries are required [Blum et al. JMLR 04]