Auctions for Structured Procurement

Matt Cary,
Google, Inc.

Abraham Flaxman,
Microsoft Research

Jason Hartline,
Northwestern U.

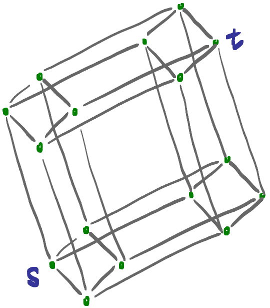
Anna Karlin,
U. Washington

January 20, 2008

Structured Procurement Auction Example

Example—(s, t)-path procurement auction

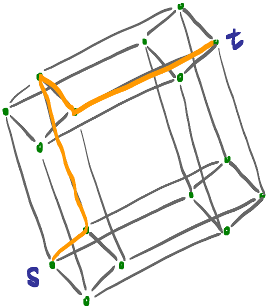
- Communication network represented by a graph, $G = (V, E)$
- Node $s \in V$ wants to receive a message from node $t \in V$



Structured Procurement Auction Example

Example—(s, t)-path procurement auction

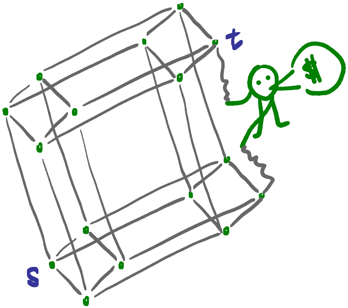
- Each edge of the network is controlled by a utility maximizing agent
- Node s can pay edges to transmit the message



Structured Procurement Auction Example

Example—(s, t)-path procurement auction

- Each edge of the network is controlled by a utility maximizing agent
- Node s can pay edges to transmit the message



Simple Procurement Auction Example

Example—2nd price auction, *procurement version*

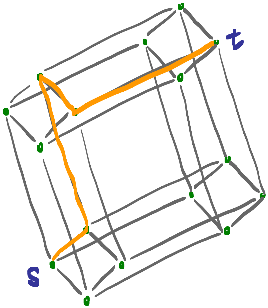
- For the public good, we must hire a Pokémon

  (Any Pokémon will do the job just as well as any other.)
- Each Pokémon has private value for doing the job.
- Hire the cheapest monster, pay it the second cheapest price.

Structured Procurement Auction Example

Example—(s, t)-path procurement auction

- Each edge of the network is controlled by a utility maximizing agent
- Node s can pay edges to transmit the message



Structured Procurement Auction Example

VCG mechanism for (s, t)-path procurement

- Auctioneer s asks all edges what they will charge to transmit a message
- Each edge e replies with a bid b_e
- Auctioneer selects the cheapest (s, t)-path, and pays

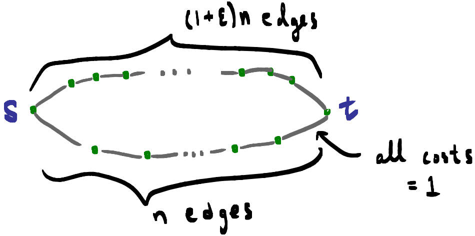
$$p_e = \text{dist}_{-e}(s, t) - \text{dist}(s, t) + b_e$$

to each edge on this path.

Frugality in combinatorial auctions

Negative results for path auctions:

- Frugality may be large.



An Alternative Benchmark

We draw inspiration from digital goods auctions.

Many things, you don't want to do just once.

- In digital goods auctions, can approximately maximize profits by tricky choice of how many items to sell.
- Let's reformulate procurement auction so we can decide how many items to buy.

Multiple Procurement Auction Example

Example—Multiple Path Procurement Auction

- We can buy as many (s, t)-paths as we desire.
- Each path is worth v (so k paths are worth $k \cdot v$).
- Now we have some flexibility; we can pick k.
- $VCG_k =$ cost of procuring the cheapest k disjoint paths via the VCG (generalized 2nd price) mechanism.
- Can we design mechanism which compares well with the benchmark value $\max_k \{k \cdot v - VCG_k\}$?

Outline of Present Paper

- General definition of this Multiple Item Structured Procurement Auction framework.
- Reduction from optimization problem to decision problem.
- Investigate when decision version of problem has solution.
- In cases where decision version is always truthful (Matroid Procurement), compare random sampling auction to Multiple Procurement Benchmark.

General Formulation

Multiple Item Structured Procurement Auction:

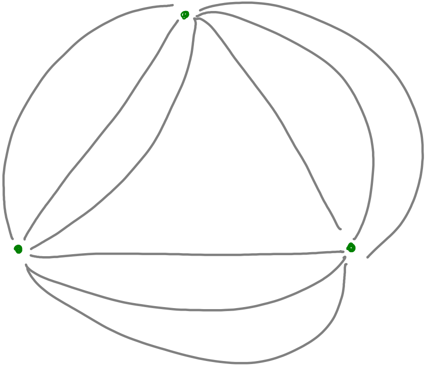
- Agents correspond to elements of $E = \{1, \ldots, N\}$, feasible sets $\mathcal{F} \subseteq 2^E$.
- Each set is worth v to auctioneer, so k disjoint sets are worth $k \cdot v$.
- VCG_k is the cost of procuring the cheapest k disjoint sets in $\mathcal{F}$ via the VCG (generalized 2nd price) mechanism.

- Benchmark we will compare against is *Multiple Procurement Benchmark*

$$OPT = \max_k \{k \cdot v - VCG_k\}.$$

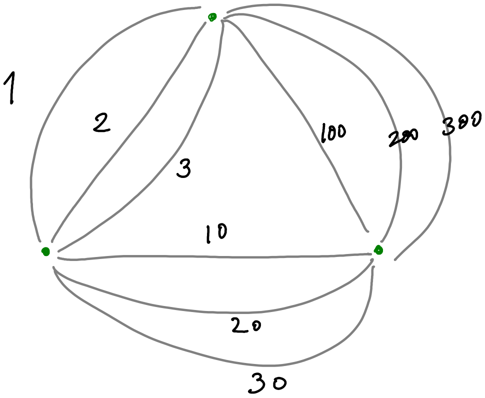
Multiple Spanning Tree Procurement Example

E = edges, $\mathcal{F}$ = spanning trees



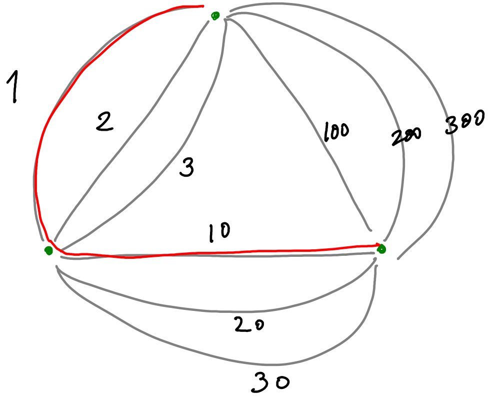
Multiple Spanning Tree Procurement Example

$v = 100$



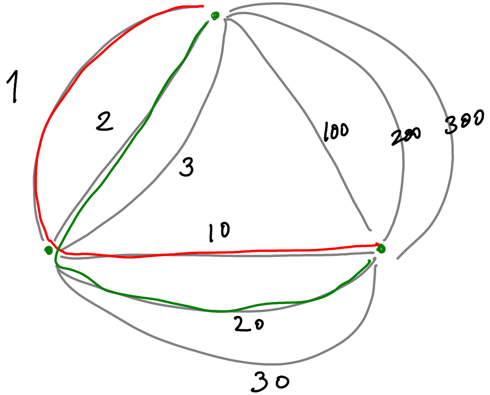
Multiple Spanning Tree Procurement Example

$$1 \cdot v - VCG_1 = 100 - (2 + 20) = 78$$



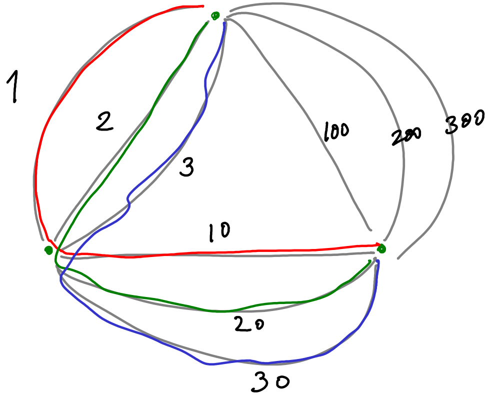
Multiple Spanning Tree Procurement Example

$$2 \cdot v - VCG_2 = 200 - (3 + 3 + 30 + 30) = 134$$



Multiple Spanning Tree Procurement Example

$$3 \cdot v - VCG_3 = 300 - (100 + 100 + 100 + 100 + 100 + 100) = -300$$



Reduction to Decision Problem

Definition

The *Profit Extraction Mechanism* with target revenue R, procurement utility v and set system $(E, \mathcal{F})$ works as follows:

1. Find the largest k such that the VCG_k satisfy $v \cdot k - VCG_k \geq R$.
2. If such a k exists, procure cheapest k disjoint sets in $\mathcal{F}$ with the VCG_k payments.
3. Otherwise, procure $\emptyset$ with 0 payments for all.

Theorems on Profit Extracting

Theorem

The Profit Extraction Mechanism is truthful for matroid set systems.

Theorem

The Profit Extraction Mechanism is not truthful for non-matroid set systems; For any non-matroid $\mathcal{F}$, there is a set of private values c and a choice of R for which the profit extractor is not truthful.

Random Sampling

Definition (RSPE)

The *Random Sampling Profit Extraction* auction (RSPE) on E:

1. Randomly partition the agents E into two parts E' and E''.
2. Compute the optimal benchmark on each part: $R' = OPT(E')$ and $R'' = OPT(E'')$.
3. Run the profit extractor with R'' on E' and likewise with R' on E''.

Random Sampling

Theorem

Let $(E, \mathcal{F})$ be a set system whose feasible sets are the bases of a matroid M.

Let $k^\star = \text{argmax}_k \{v \cdot k - VCG_k\}$ and $OPT = v \cdot k^\star - VCG_{k^\star}$.

If $k^\star \geq \frac{8}{\epsilon^2} \log(\text{rank}(M))$ then, for any $\epsilon > 0$, the RSPE procurement mechanism obtains profit $\geq \frac{1-\epsilon}{2} OPT$ with probability $1 - \frac{2}{\text{rank}(M)}$.

Conclusion And Open question

- Define multiple procurement benchmark for structured procurement auctions.
- Reduce optimization problem to decision problem, show this is truthful for Matroid Procurement
- *Show random sampling and decision problem solution combine to give constant approximate optimal solution for Matroid Procurement*.
- Open Questions:
  - Does something work for non-matroids? Especially for path auctions?
  - For what set systems does decision problem have truthful solution?