Computation and Incentives in Public Projects

Dave Buchfuhrer   Michael Schapira   Yaron Singer

Caltech, Yale, Berkeley

June 9, 2010

Outline

Public Projects

A combinatorial public project is a game in which the goal is to choose k items from a set of m to provide for shared use among n agents.

Public Projects

A combinatorial public project is a game in which the goal is to choose k items from a set of m to provide for shared use among n agents.

This differs from an auction in that allocated items are shared.

Definition (Social Welfare)

Suppose that each agent i gets value $v_i(S)$ for allocation S. Then the social welfare of S is

$$\sum_i v_i(S)$$

History

- Public projects were first studied by Papadimitriou, Schapira and Singer in a 2008 FOCS paper titled *On the Hardness of Being Truthful*
- Our results use techniques from this paper to achieve hardness results for approximating social welfare with maximal-in-range mechanisms
- These techniques were also used in a recent paper in SODA 2010, *Limits on the Social Welfare of Maximal-In-Range Auction Mechanisms* by Buchfuhrer, Dughmi, Fu, Kleinberg, Mossel, Papadimitriou, Schapira, Singer and Umans.

Maximal-in-Range (MIR)

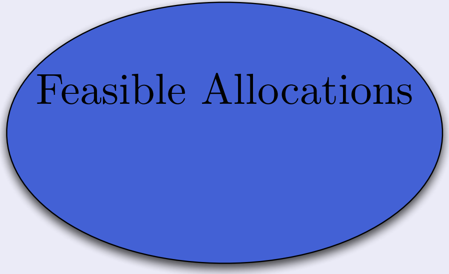
Definition (Maximal-in-Range)

An allocation algorithm is maximal-in-range if there exists some range R such that the algorithm always outputs an allocation from R that maximizes the social welfare.

Maximal-in-Range (MIR)

Definition (Maximal-in-Range)

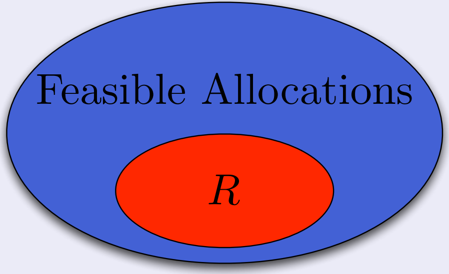
An allocation algorithm is maximal-in-range if there exists some range R such that the algorithm always outputs an allocation from R that maximizes the social welfare.



Maximal-in-Range (MIR)

Definition (Maximal-in-Range)

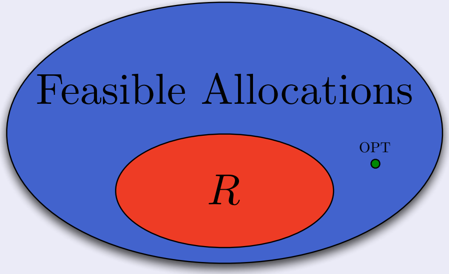
An allocation algorithm is maximal-in-range if there exists some range R such that the algorithm always outputs an allocation from R that maximizes the social welfare.



Maximal-in-Range (MIR)

Definition (Maximal-in-Range)

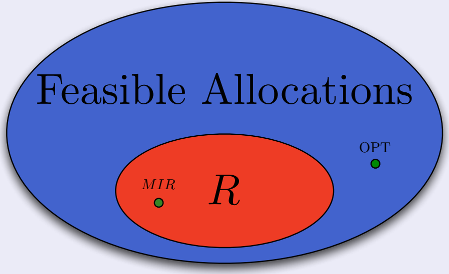
An allocation algorithm is maximal-in-range if there exists some range R such that the algorithm always outputs an allocation from R that maximizes the social welfare.



Maximal-in-Range (MIR)

Definition (Maximal-in-Range)

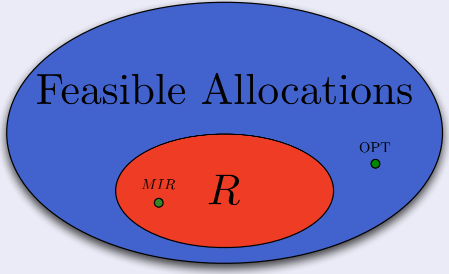
An allocation algorithm is maximal-in-range if there exists some range R such that the algorithm always outputs an allocation from R that maximizes the social welfare.



Maximal-in-Range (MIR)

Definition (Maximal-in-Range)

An allocation algorithm is maximal-in-range if there exists some range R such that the algorithm always outputs an allocation from R that maximizes the social welfare.



- An algorithm can be implemented truthfully via VCG iff it is MIR
- For sufficiently general valuations, VCG is the only truthful mechanism

Performance of MIR mechanisms

Theorem (sketch)

A maximal in range allocation algorithm for any NP-hard combinatorial public project cannot approximate the welfare with a ratio better than $\sqrt{m}$ *unless* $NP \subseteq P/poly$.

Proof scheme.

- A mechanism that gets better than a $\sqrt{m}$ ratio requires an exponential range for sufficiently expressive valuation classes (PSS 08)

□

Performance of MIR mechanisms

Theorem (sketch)

A maximal in range allocation algorithm for any NP-hard combinatorial public project cannot approximate the welfare with a ratio better than $\sqrt{m}$ unless $NP \subseteq P/poly$.

Proof scheme.

- A mechanism that gets better than a $\sqrt{m}$ ratio requires an exponential range for sufficiently expressive valuation classes (PSS 08)
- By Sauer's lemma, an exponential range must contain a polynomial-sized subset S^* of items allocated in every way

□

Performance of MIR mechanisms

Theorem (sketch)

A maximal in range allocation algorithm for any NP-hard combinatorial public project cannot approximate the welfare with a ratio better than $\sqrt{m}$ *unless* $NP \subseteq P/poly$.

Proof scheme.

- A mechanism that gets better than a $\sqrt{m}$ ratio requires an exponential range for sufficiently expressive valuation classes (PSS 08)
- By Sauer's lemma, an exponential range must contain a polynomial-sized subset S^* of items allocated in every way
- We construct instances in which it is NP-hard to determine which members of S^* should be selected. These follow fairly directly from the proofs of NP-hardness. □

Embedding NP-hard problems into S^*

Suppose we have an NP-hardness reduction to the problem

Example

Does there exist a subset $S \subseteq [m], |S| = k$ such that $v(S) = v([m])$?

Embedding NP-hard problems into S^*

Suppose we have an NP-hardness reduction to the problem

Example

Does there exist a subset $S \subseteq [m], |S| = k$ such that $v(S) = v([m])$?

We simply embed $[m]$ into S^* and set social welfare

$$v'(S = S_1 \cup S_2) = v(S_1) + \epsilon |S_2|$$

where $S_1 \subseteq S^*, S_2 \subset [m'] \backslash S^*$

Embedding NP-hard problems into S^*

Suppose we have an NP-hardness reduction to the problem

Example

Does there exist a subset $S \subseteq [m], |S| = k$ such that $v(S) = v([m])$?

We simply embed $[m]$ into S^* and set social welfare

$$v'(S = S_1 \cup S_2) = v(S_1) + \epsilon |S_2|$$

where $S_1 \subseteq S^*, S_2 \subset [m'] \backslash S^*$

Lemma

The auction after this embedding has social welfare $v([m]) + \epsilon(k' - k)$ iff there is a set $S \subseteq [m], |S| = k$ such that $v(S) = v([m])$.

Outline

Definition

Definition (Unit-Demand Valuation)

An agent with a unit-demand valuation has private values w_j for each item j, and has total value

$$v_i(S) = \max_{j \in S} w_j$$

for set S.

In auctions, unit-demand agents are trivial.

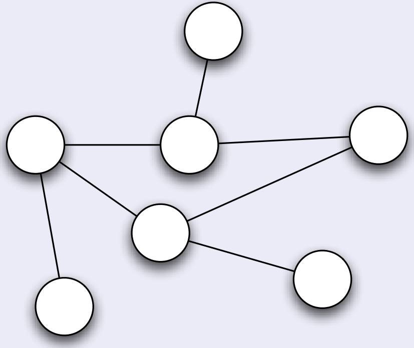
NP hardness

Theorem

The public project problem with unit-demand agents is NP-hard.

Proof by picture

Reduction from vertex cover:



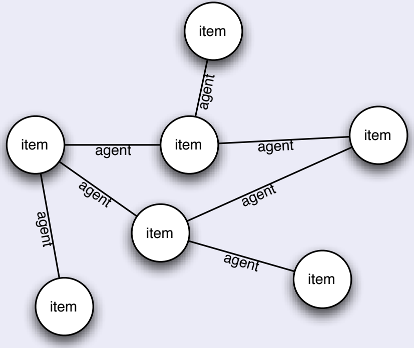
NP hardness

Theorem

The public project problem with unit-demand agents is NP-hard.

Proof by picture

Reduction from vertex cover:



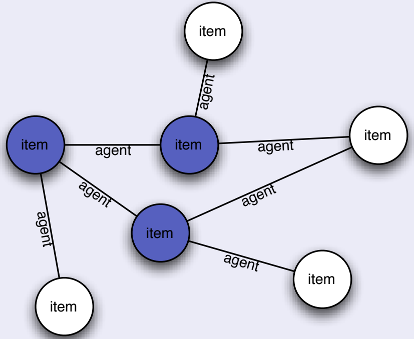
NP hardness

Theorem

The public project problem with unit-demand agents is NP-hard.

Proof by picture

Reduction from vertex cover:



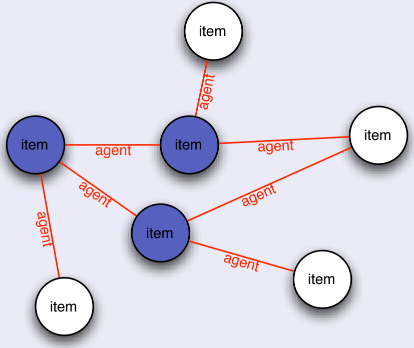
NP hardness

Theorem

The public project problem with unit-demand agents is NP-hard.

Proof by picture

Reduction from vertex cover:



2-$\{0, 1\}$ Unit-Demand

Definition (2-$\{0, 1\}$ Unit-Demand)

An agent has a 2-$\{0, 1\}$ unit-demand valuation if for some two items i, j:

$$v(S) = \begin{cases} 1 & i \in S \vee j \in S \\ 0 & \text{otherwise} \end{cases}$$

The previous proof showed hardness for 2-$\{0, 1\}$ unit-demand agents, as an agent is satisfied if one of the items chosen corresponds to one of the 2 endpoints of his edge.

Truthful Approximation

Recall

NP hardness means VCG mechanisms can't beat a $\sqrt{m}$ approximation

Truthful Approximation

Recall

NP hardness means VCG mechanisms can't beat a $\sqrt{m}$ approximation

Theorem

There exists a truthful 2-approximation 2-$\{0, 1\}$ unit-demand agents

Truthful Approximation

Recall

NP hardness means VCG mechanisms can't beat a $\sqrt{m}$ approximation

Theorem

There exists a truthful 2-approximation 2-$\{0,1\}$ *unit-demand agents*

Mechanism

Choose the k items corresponding to the vertices of highest degree

Truthful Approximation

Recall

NP hardness means VCG mechanisms can't beat a $\sqrt{m}$ approximation

Theorem

There exists a truthful 2-approximation 2-$\{0, 1\}$ *unit-demand agents*

Mechanism

Choose the k items corresponding to the vertices of highest degree

Proof.

- The number of edges covered is at least half the sum of degrees
- There's no benefit to lying

□

Outline

Definition

Definition (Coverage Valuation)

An agent with a coverage valuation associates a set T_j with each item j, and has value

$$v_i(S) = \left| \bigcup_{j \in S} T_j \right|$$

for set S.

NP hardness

Theorem

The public projects problem with a single coverage valuation agent is NP-hard.

Definition (max-k-cover)

Input: Several sets $T_1, \ldots, T_m$

Goal: Find a set $S \subseteq [m], |S| = k$ maximizing $|\bigcup_{j \in S} T_j|$

NP hardness

Theorem

The public projects problem with a single coverage valuation agent is NP-hard.

Definition (max-k-cover)

Input: Several sets $T_1, \ldots, T_m$
Goal: Find a set $S \subseteq [m], |S| = k$ maximizing $|\bigcup_{j \in S} T_j|$

Definition (Public Project with 1 Coverage Valuation Agent)

Input:
Goal:

NP hardness

Theorem

The public projects problem with a single coverage valuation agent is NP-hard.

Definition (max-k-cover)

Input: Several sets $T_1, \ldots, T_m$
Goal: Find a set $S \subseteq [m], |S| = k$ maximizing $|\bigcup_{j \in S} T_j|$

Definition (Public Project with 1 Coverage Valuation Agent)

Input: Several sets $T_1, \ldots, T_m$
Goal:

NP hardness

Theorem

The public projects problem with a single coverage valuation agent is NP-hard.

Definition (max-k-cover)

Input: Several sets $T_1, \ldots, T_m$
Goal: Find a set $S \subseteq [m], |S| = k$ maximizing $|\bigcup_{j \in S} T_j|$

Definition (Public Project with 1 Coverage Valuation Agent)

Input: Several sets $T_1, \ldots, T_m$
Goal: Find a set $S \subseteq [m], |S| = k$ maximizing $|\bigcup_{j \in S} T_j|$

Wait a second...

Theorem

No truthful poly-time mechanism for public projects can achieve better than a $\sqrt{m}$ approximation unless $NP \subseteq P/poly$.

Proof.

- Our results show hardness for VCG to do better than $\sqrt{m}$
- For a single agent, any mechanism must be maximal-in-range to be truthful, so VCG is all there is □

Outline

Summary of Results

Computational Results

valuation class	no. of agents	appx. ratio r
unit-demand	constant n	$r = 1$ $r = 1 - \frac{1}{e}$ [New]
multi-unit-demand	$1, 2$ 3 ≥ 4 ≥ 10 n	$r = 1$ [New] $2/3$ [New] $\leq r < 1$ [New] $1 - \frac{1}{e}$ [10] $\leq r < 1$ [New] $1 - \frac{1}{e}$ [10] $\leq r < 1 - \epsilon$ (no PTAS)[New] $r = 1 - \frac{1}{e}$ [New]
capped additive	1 constant ≥ 2 n	$r = 1$ $r = 1 - \epsilon$ (FPTAS) [New] $r = 1 - \frac{1}{e}$ [New]
fractionally-subadditive	constant n	$r = 1$ $\max\{\frac{1}{n}, \frac{1}{\sqrt{m}}\}$ [16] $\leq r \leq 2^{-\frac{\log^{1-\gamma} n}{6}}$ [New]

Truthful Results

<table>
<tr><th>valuation class</th><th>no. of agents</th><th>Truthful appx. ratio r</th><th>VCG-based appx. ratio r</th></tr>
<tr><td>2-{0,1} unit-demand</td><td>n</td><td>$1/2 \leq r < 1$ [New]</td><td rowspan="7">$r = \frac{1}{\sqrt{m}}$ [New]</td></tr>
<tr><td>unit-demand</td><td>n</td><td>?</td></tr>
<tr><td rowspan="2">multi-unit-demand</td><td>3</td><td>$2/3 \leq r < 1$ [New]</td></tr>
<tr><td>n</td><td>?</td></tr>
<tr><td>capped-additive</td><td>≥ 2</td><td>?</td></tr>
<tr><td>coverage</td><td>1</td><td>$r = \frac{1}{\sqrt{m}}$ [New]</td></tr>
<tr><td>fractionally-subadditive</td><td>n</td><td>?</td></tr>
</table>

Conclusions and Open Problems

- Public projects are hard even for simple classes of valuations, allowing for mechanism design to be explored on simpler problems than in auctions
- Can we improve upon the VCG mechanism in simple public projects?
- The requirement of truth can be too much even for a single agent
- Can we define a satisfying substitute for truth in these situations?