The Power of Two Prices: Beyond Cross-Monotonicity

Incentive-Compatible Mechanism Design

Yvonne Bleischwitz **Burkhard Monien**
Florian Schoppmann Karsten Tiemann

Department of Computer Science

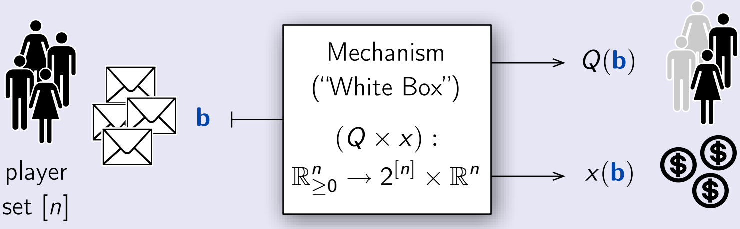
University of Paderborn

March 27, 2007

The Model

- $n \in \mathbb{N}$ players:
  - Have private valuations $v_i \in \mathbb{R}_{\geq 0}$ for service, $\mathbf{v} := (v_i)_{i \in [n]}$
  - Submit bids $b_i \in \mathbb{R}_{\geq 0}$ to service provider, $\mathbf{b} := (b_i)_{i \in [n]}$
- Service provider uses mechanism to determine outcome:

Definition (Cost-Sharing Mechanism)



- Desirable that $\mathbf{b} = \mathbf{v}$ but this cannot be a priori guaranteed

Common Assumptions for Cost-Sharing Mechanisms

Only consider mechanisms with the following properties $\forall i \in [n]$:

- NPT (No Positive Transfer) = no negative payments:

$$x_i(\mathbf{b}) \geq 0$$

- VP (Voluntary Participation) = obey bids:

$$x_i(\mathbf{b}) \leq b_i$$

- CS* (Strict Consumer Sovereignty):

$$\text{CS: } \exists b_i^+ \in \mathbb{R}_{\geq 0} : \forall \mathbf{b} \in \mathbb{R}^n_{\geq 0} : (b_i \geq b_i^+ \Longrightarrow i \in Q(\mathbf{b}))$$

$$\text{Strictness: } \forall \mathbf{b} \in \mathbb{R}^n_{\geq 0} : (b_i = 0 \Longrightarrow i \notin Q(\mathbf{b}))$$

Assume: $\mathbf{v}$ is true valuation vector, $(Q \times x)$ mechanism

- Player i's utility depends on bid vector:

$$u_i(\mathbf{b}) := \begin{cases} v_i - x_i(\mathbf{b}) & \text{if } i \in Q(\mathbf{b}) \\ 0 & \text{if } i \notin Q(\mathbf{b}) \end{cases}$$

Desirable Properties of Cost-Sharing Mechanisms

- GSP (Group-Strategyproofness):
  $\forall$ true valuations $\mathbf{v} \in \mathbb{R}^n_{\geq 0}$: $\nexists$ coalition $K \subseteq [n]$ such that
  $\exists$ cheating possibility $\mathbf{b}_K \in \mathbb{R}^K_{\geq 0}$ with
  - $u_i(\mathbf{v}_{-K}, \mathbf{b}_K) \geq u_i(\mathbf{v})$ for all $i \in K$ and
  - $u_i(\mathbf{v}_{-K}, \mathbf{b}_K) > u_i(\mathbf{v})$ for at least one $i \in K$.

  SP: Needs to hold only for coalitions K of size 1

Definition (n-Player Cost Function)

Function $C : 2^{[n]} \rightarrow \mathbb{R}_{\geq 0}$ with $C(A) = 0 \Longleftrightarrow A = \emptyset$

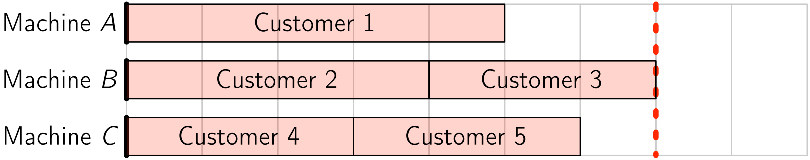
- β-BB (β-Budget-Balance, with $0 \leq \beta \leq 1$):

$$\beta \cdot C(Q(\mathbf{b})) \leq \sum_{i \in [n]} x_i(\mathbf{b}) \leq OPT(Q(\mathbf{b}))$$

A Cost-Sharing Scenario

Computing center with large cluster of parallel machines

- Offering customers (uninterrupted) processing times
- Cost proportional to makespan



$\text{Makespan}(\{1,2,3,4,5\}) = 6$

Implications of GSP

GSP is a very strong requirement:

- Even coalitions with binding agreements should have no incentive to cheat

Theorem (Moulin, 1999)

Let $(Q \times x)$ be a GSP cost-sharing mechanism, $\mathbf{b}, \mathbf{b}' \in \mathbb{R}^n_{\geq 0}$ bid vectors with $Q(\mathbf{b}) = Q(\mathbf{b}')$. Then $x_i(\mathbf{b}) = x_i(\mathbf{b}')$ for all $i \in [n]$.

Hence, GSP (with standard assumptions NPT, VP, CS*) implies:

- Payments independent of bids
- Bids only determine set of serviced players

Cost-Sharing Methods

Last theorem gives rise to:

Definition (n-Player Cost-Sharing Method)

Function $\xi : 2^{[n]} \rightarrow \mathbb{R}^n_{\geq 0}$.

ξ is cross-monotonic if $\forall A, B \subseteq [n]$ and $\forall i \in A : \xi_i(A) \geq \xi_i(A \cup B)$

Note:

- β-Budget-balance defined as before:

$$\forall A \subseteq [n] : \beta \cdot C(A) \leq \sum_{i \in [n]} \xi_i(A) \leq OPT(A)$$

Moulin Mechanisms

Algorithm $M_\xi : \mathbb{R}^n_{\geq 0} \to 2^{[n]} \times \mathbb{R}^n$ (Moulin, 1999)

Input: $\mathbf{b} \in \mathbb{R}^n_{\geq 0}$; *Output:* $Q \in 2^{[n]}$, $\mathbf{x} \in \mathbb{R}^n$

$Q := [n]$

while $\exists i \in Q\colon\ b_i < \xi_i(Q)$ **do** $Q := \{i \in Q \mid b_i \geq \xi_i(Q)\}$

$\mathbf{x} := \xi(Q)$

Theorem (Moulin, 1999)

M_ξ satisfies GSP and β-BB if ξ is cross-monotonic and β-BB.

Submodular Cost Functions

Definition (Submodular Cost-Function)

Cost function $C : 2^{[n]} \to \mathbb{R}_{\geq 0}$ where for all $A \subseteq B \subseteq [n]$ and $i \notin B$

$$C(A \cup \{i\}) - C(A) \geq C(B \cup \{i\}) - C(B).$$

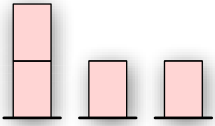
Complete characterization when C submodular:

Theorem (Moulin, 1999)

Any GSP and 1-BB mechanism has cross-monotonic cost-shares. A 1-BB cross-monotonic ξ exists. Hence, M_ξ is GSP and 1-BB.

Submodular seems natural ("marginal costs only decrease"), but:

- Example: makespan scheduling
  $C([1]) = 1$, $C([2]) = 1$,
  $C([3]) = 1$, $C([4]) = 2$



Previous Research

Good BB. Examples for cross-monotonic cost-sharing methods:

Authors	*Problem*	β^{-1}
Jain, Vazirani (2001)	MST	1
	Steiner tree, TSP	2
Pál, Tardos (2003)	Facility location	3
	Single-Source-Rent-or-Buy	15
Gupta et. al. (2003)	Single-Source-Rent-or-Buy	4.6
Könemann et. al. (2005)	Steiner forest	2
Bleischwitz, Monien (2006)	Scheduling on m links	$\frac{2m}{m+1}$

$\vdots$

A Note on Modeling Assumptions

Recall:

- CS: $\exists b_i^+ \in \mathbb{R}_{\geq 0} : \forall \mathbf{b} \in \mathbb{R}^n_{\geq 0} : (b_i \geq b_i^+ \Longrightarrow i \in Q(\mathbf{b}))$
- CS*: CS and also $\forall \mathbf{b} \in \mathbb{R}^n_{\geq 0} : (b_i = 0 \Longrightarrow i \notin Q(\mathbf{b}))$

Trivial GSP, 1-BB mechanism if only CS (Immorlica et. al., 2005):

- "Taking a fixed order, find 1st agent who can pay for the rest"

Even stronger than CS*:

- NFR (No Free Riders):

$$i \in Q(\mathbf{b}) \Longrightarrow x_i(\mathbf{b}) > 0$$

Symmetric Costs

With CS*, it is much harder to achieve GSP and good BB.

Does symmetry of costs help? That is, for $A, B \subseteq [n]$ we have

$$|A| = |B| \Longrightarrow C(A) = C(B).$$

We define $c : [n] \to \mathbb{R}_{\geq 0}$, $c(i) := C([i])$ in this case.

Our results (not discussed in this talk):

- We give a general GSP, 1-BB mechanism for 3 or less players
- There is a 4-player symmetric cost function for which no GSP, 1-BB mechanism exists

The Power of Two Prices

Bleischwitz, Monien (2006): For makespan costs (weights or machines identical), cross-monotonic methods are no better than $\frac{m+1}{2m}$-BB in general

- Is there a mechanism that is better than Moulin here? (Recall: Makespan is not submodular function)
- Is it a generic mechanism?

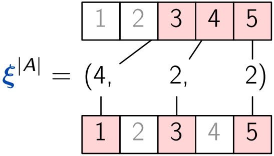
Yes,

if the cost function is symmetric.

Cost-Sharing Forms (1/2)

- Preference order. Cost vectors $\boldsymbol{\xi}^j \in \mathbb{R}^j_{\geq 0}$, $j \in [n]$, such that for $i \in [n]$, $A \subseteq [n]$:

$$\xi_i(A) := \begin{cases} \xi^{|A|}_{\mathrm{Rank}(i,A)} & \text{if } i \in A \\ 0 & \text{otherwise.} \end{cases}$$



- At most 2 different cost-shares for any set of players $A \subseteq [n]$

Definition (Cost-Sharing Form)

Consists of: Sequence $(a_k, \lambda_k)_{k \in \mathbb{N}} \subset \mathbb{R}^2_{>0}$, mappings $\sigma : \mathbb{N} \to \mathbb{N}$, $f : \mathbb{N} \to \mathbb{N}_0$

A cost-sharing form defines cost vectors $\boldsymbol{\xi}^i$, $i \in \mathbb{N}$:

$$\boldsymbol{\xi}^i = (\underbrace{\lambda_{\sigma(i)}, \ldots, \lambda_{\sigma(i)}}_{f(i) \text{ elements}}, a_{\sigma(i)}, \ldots, a_{\sigma(i)})$$

Cost-Sharing Forms (2/2)

Recall: A cost-sharing form defines cost vectors $\boldsymbol{\xi}^i$, $i \in \mathbb{N}$:

$$\boldsymbol{\xi}^i = (\underbrace{\lambda_{\sigma(i)}, \ldots, \lambda_{\sigma(i)}}_{f(i) \text{ elements}}, a_{\sigma(i)}, \ldots, a_{\sigma(i)})$$

Valid cost-sharing form:

- $\sigma(i+1) \in \{\sigma(i), \sigma(i)+1\}$
- $\sigma(i+1) = \sigma(i) + 1$
  $\Longrightarrow f(i+1) = 0$
- $f(1) = 0$
- $f(i+1) \leq f(i) + 1$
- $\lambda_k \geq a_k \geq a_{k-1}$

Example:

i	$f(i)$	$\sigma(i)$	$\boldsymbol{\xi}^i$
1	0	1	(2)
2	0	1	(2, 2)
3	1	1	(3, 2, 2)
4	2	1	(3, 3, 2, 2)
5	0	2	(1, 1, 1, 1, 1)
6	1	2	(5, 1, 1, 1, 1, 1)

σ induces segments: Ranges of cardinalities with same cost-shares!

The New Two-Prices Mechanism: Ideas

Choose correct segment k

- Find max. $j \in [n]$ such that j players bid $\geq a_{\sigma(j)}$; Set $k := \sigma(j)$
- Reject all players $i \in [n]$ with $b_i < a_k$

Cost-sharing policy when j in segment k, i.e., $\sigma(j) = k$

- $\xi^j = (\underbrace{\lambda_k, \ldots, \lambda_k}_{f(j)}, \underbrace{a_k, \ldots, a_k}_{j-f(j) \text{ players}})$; recall: $\lambda_k \geq a_k$

Serve as many players for a_k as possible

- Handling indifferent players (i.e., $b_i = a_k$) optimizes other players' utilities
- If necessary: Least preferred agents have to pay λ_k

Intuition:

- Serving least preferred player for λ_k never hurts others because $f(i+1) \leq f(i) + 1$

The New Two-Prices Mechanism: Formal Algorithm

Two-Prices Mechanism

Input: $\mathbf{b}$; *Output:* $Q \in 2^{[n]}, \mathbf{x} \in \mathbb{R}^n$

$k := \max\left\{ i \in [n] \;\middle|\; |\{j \in [n] \mid b_j \geq a_{\sigma(i)}\}| \geq i \right\} \cup \{0\}$

if $k = 0$ **then** $(Q, \mathbf{x}) := (\emptyset, 0)$; **return**

$H := \emptyset$; $L := \{i \in [n] \mid b_i \geq a_k\}$

$\nu := |\{i \in [n] \mid b_i = a_k\}|$

loop

  $q := \max\{q \in [|H| + |L|] \mid f(q) = |H|\}$

  if $q \geq |H| + |L| - \nu$ **then**

    $S := \{i \in N \mid b_i > a_k\}$

    $L := S \cup \{q - |H| - |S|$ largest elements i of L with $b_i = a_k\}$

    break

  else

    if $b_{\min L} \geq \lambda_k$ **then** $H := H \cup \{\min L\}$

    else if $b_{\min L} = a_k$ **then** $\nu := \nu - 1$

    $L := L \setminus \{\min L\}$

$Q := H \cup L$; $\mathbf{x} := \xi(Q)$

The New Two-Prices Mechanism: Example

Algorithm (for computing the Two-Prices Mechanism)

```
Find max. j ∈ [n] such that j players bid ≥ a_σ(j); Set k := σ(j)
Reject all players i ∈ [n] with b_i < a_k
loop
    If possible: Include remaining agents for a_k by rejecting in-
        different agents, then stop
    Else: Least preferred agent is included for λ_k or is rejected
```

Example for $\mathbf{b} = (\frac{5}{2}, 3, 3, 2, 0, 0)$:

- $a_k = 2$, reject agents 5, 6
- only agent 4 is indifferent
- Can't include 1,2,3 even w/o 4
- Reject agent 1 because $\frac{5}{2} = b_i < \lambda_k = 3$
- Include 2,3 by rejecting 4

i	$f(i)$	$\sigma(i)$	ξ^i
1	0	1	(2)
2	0	1	$(2, 2)$
3	1	1	$(3, 2, 2)$
4	2	1	$(3, 3, 2, 2)$
5	0	2	$(1, 1, 1, 1, 1)$
6	1	2	$(5, 1, 1, 1, 1, 1)$

Two-Prices Mechanism is GSP

Theorem

The two-prices menchanism is GSP and NFR.

Proof (Sketch). Let $\mathbf{v} \in \mathbb{R}^n_{\geq 0}$ be true valuation vector, $\mathbf{b} \in \mathbb{R}^n_{\geq 0}$ other bid vector and $K \subseteq [n]$ such that $\mathbf{b}_{-K} = \mathbf{v}_{-K}$. We show:

$$\exists i \in K : u_i(\mathbf{v}_{-K}, \mathbf{b}_K) > u_i(\mathbf{v}) \Longrightarrow \exists j \in K : u_j(\mathbf{v}_{-K}, \mathbf{b}_K) < u_j(\mathbf{v})$$

Outline of proof:

- Do not need to consider $\sigma(|Q(\mathbf{b})|) \neq \sigma(|Q(\mathbf{v})|)$
- Assumptions imply: $x_i(\mathbf{v}) \in \{0, \lambda_k\}$, but $x_i(\mathbf{b}) = a_k$
- Only two options:
  - $\exists j \in [i] : b_j \geq \lambda_k > v_j$ or
  - $\exists j \in \{i+1, \dots, n\} : b_j \leq a_k < v_j$

It follows that $j \in K$ and $u_j(\mathbf{b}) < u_j(\mathbf{v})$ □

A Two-Price Cost-Sharing Form for Subadditive Costs

C is subadditive if $\forall A, B \subseteq [n]$, $C(A \cup B) \leq C(A) + C(B)$.

Algorithm (for computing makespan cost-sharing form)

Input: $c : [n] \to \mathbb{R}_{\geq 0}$; *Output:* $(a_k, \lambda_k), \sigma : \mathbb{N} \to \mathbb{N}, f : \mathbb{N} \to \mathbb{N}_0$

$r := 0$; $a_1 := \infty$

for $i := 1, \ldots, n$ **do**

  if $\frac{c(i)}{i} \leq a_r$ **then** $r := r + 1$; $a_r := \frac{c(i)}{i}$; $f(i) := 0$

  else

    if $f(i-1) = 0$ and $i \cdot a_r < \frac{3}{4} \cdot c(i)$ **then** $\lambda_r := \frac{c(i)}{4}$

    if λ_r still undefined **then** $f(i) := 0$

    else

      $f(i) := \max\{j \in [f(i-1)+1]_0 \mid \lambda_r \cdot j + (i-j) \cdot a_r \leq c(i)\}$

  $\sigma(i) := r$

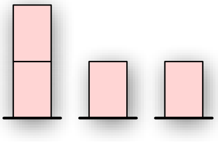
Scheduling Example

Algorithm:						Cost Vectors:
i	$c(i)$	$\sigma(i)$	$a_{\sigma(i)}$	$\lambda_{\sigma(i)}$	$f(i)$	ξ^i
1	1	1	$c(1) = 1$	–	0	(1)
2	1	2	$\frac{c(2)}{2} = \frac{1}{2}$	–	0	$(\frac{1}{2}, \frac{1}{2})$
3	1	3	$\frac{c(3)}{3} = \frac{1}{3}$	–	0	$(\frac{1}{3}, \frac{1}{3}, \frac{1}{3})$
4	2	3	–	$\frac{1}{4} \cdot c(4) = \frac{1}{2}$	1	$(\frac{1}{2}, \frac{1}{3}, \frac{1}{3}, \frac{1}{3})$

Consider $i = 4$:

- $\frac{c(4)}{4} = \frac{1}{2} > \frac{1}{3} = a_{\sigma(3)}$. Hence, $\sigma(4) = \sigma(3)$.
- Furthermore, $4 \cdot \frac{1}{3} = \frac{4}{3} < \frac{3}{4} \cdot c(4) = \frac{3}{2}$. Hence, $\lambda_{\sigma(4)} = \frac{1}{4} \cdot c(4)$

Optimal Makespan:



Budget-Balance

Theorem

The two-price cost-sharing mechanism used with a cost-sharing form computed for subadditive costs is $\frac{3}{4}$-BB and NFR.

Proof (Idea).

- GSP: Follows from before
- NFR: By the algorithm, $\forall i \in [n] : a_{\sigma(i)} > 0$
- BB: Use: c non-decreasing and subadditive

$\frac{3}{4}$ is the best to expect from any valid cost-sharing form:

Theorem

$\forall \varepsilon \in (0, \frac{1}{4}]$, *there are scheduling instances (identical jobs and machines) for which no* $(\frac{3}{4} + \varepsilon)$*-BB cost-sharing form exists.*

Conclusion and Further Research (1/2)

Motivation:

- Mechanism Design: Align players' incentives to global objective

New results presented in this talk:

- Generic GSP mechanism without free riders (symmetric costs)
- β-BB if the underlying cost-sharing form is β-BB
- Application: Makespan mechanisms (identical jobs)
  - Best-known BB improved from $\frac{m+1}{2m}$ to $\frac{3}{4}$
  - Best our new technique can yield in general
- For ≥ 4 players, symmetry of costs not sufficient for existence of 1-BB, GSP mechanism
- For ≤ 3 players, symmetry is sufficient!

Conclusion and Further Research (2/2)

Lots of open questions:

- Generalize the approach
- What is the best budget balance factor for scheduling?
- Bringing in efficiency: Trade-Offs
- Other applications than schedling

Thank you for your attention!