The Complexity of Simple and Optimal Deterministic Mechanisms for an Additive Buyer

Xi Chen, George Matikas, Dimitris Paparas, Mihalis Yannakakis
The Set-up

Seller has n items for sale
The Set-up

Seller has $n$ items for sale

Buyer has (private) value for each item

\[
\begin{array}{cccccc}
$50 & $25 & $78 & $135 & $53 \\
\end{array}
\]

Probability distribution of value for each item, known to seller

\[
F_1 \times F_2 \times F_3 \times F_4 \times F_5
\]

Valuation of buyer drawn randomly from \( F = F_1 \times F_2 \times \ldots \times F_n \)
The Set-up

Seller has $n$ items for sale

Buyer has (private) value for each item

<table>
<thead>
<tr>
<th>Item</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pants</td>
<td>$50</td>
</tr>
<tr>
<td>Shirt</td>
<td>$25</td>
</tr>
<tr>
<td>Shoes</td>
<td>$78</td>
</tr>
<tr>
<td>Jacket</td>
<td>$135</td>
</tr>
<tr>
<td>Hat</td>
<td>$53</td>
</tr>
</tbody>
</table>

Additive buyer:
Value of a subset $S$ of items = sum of values of items in $S$
The Set-up

- Seller can assign a price to each subset
  
  🤸‍♂️: $45  🧸: $30  💰
  🤸‍♂️ 🧸: $70  💰

  💰

- or offers a menu of only some subsets (bundles)

- **Buyer’s Utility** for a subset S: $u(S) = \text{value}(S) - \text{price}(S)$

- **Buyer buys subset S with maximum utility, if $u(S) \geq 0$**

  (break ties say by highest value rule)
Optimal Pricing Problem

- Optimal Pricing (Revenue Maximization) Problem

Find pricing that maximizes the expected revenue

\[
\max E[\text{Revenue}] = \sum_{v \sim F} \text{Pr}(v) \cdot \text{price}(S_v)
\]

where \(S_v = \text{bundle bought by buyer with valuation } v\)
Single Item Pricing Scheme

• Set a price for each item

\begin{align*}
\text{Price for each subset } S : \sum \{ \text{price}(i) \mid i \in S \} \\
\end{align*}

\begin{itemize}
\item Optimal price for each item \( i \) :
\end{itemize}

value \( p^*_i \) that maximizes \( p^*_i \cdot \Pr[\text{value}(i) \geq p^*_i] \)

[Myerson ‘81]
Grand Bundle Pricing Scheme

• Can only buy the set of all items (the “grand bundle”) for a given price, or nothing at all.

• There are examples where it gets more revenue than single item pricing:
  2 iid items with values \{1, 2\} with probability \(\frac{1}{2}\) each
  - Single item pricing: opt revenue 2 (eg. price 1 for each)
  - Grand bundle pricing: opt revenue 9/4
    price 3 for the grand bundle
Partition Pricing Scheme

• Partition the items into groups and assign price to each group in partition.

  | Item 1 | $85  |
  | Item 2 | $60  |
  | Item 3 | $170 |

• Can buy any set of groups for sum of their prices

• Includes single item and grand bundle pricing as special cases

• Can get more revenue than both in some examples
Randomized Schemes (Lottery Pricing)

- Lottery = vector \((q_1, \ldots, q_n)\) of probabilities for the items. If buyer buys the lottery then she gets each item \(i\) with probability \(q_i\).

- Lottery pricing: Menu = set of (lottery, price) pairs.

<table>
<thead>
<tr>
<th>Item</th>
<th>Probability</th>
<th>Price</th>
</tr>
</thead>
<tbody>
<tr>
<td>Jeans</td>
<td>0.5</td>
<td>$120</td>
</tr>
<tr>
<td>Shirt</td>
<td>0.2</td>
<td></td>
</tr>
<tr>
<td>Shoes</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>Jacket</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>Hat</td>
<td>0.4</td>
<td></td>
</tr>
</tbody>
</table>

- Buyer buys lottery with maximum expected utility.

- There are examples where lottery pricing gives more revenue than the optimal deterministic pricing.
Pricing schemes ↔ Mechanism design

• Buyer submits a bid for each item

• Mechanism determines allocation the buyer receives and the price she pays
  Mechanism must be incentive compatible and individually rational

• Bundle pricings ↔ deterministic mechanisms

• Lottery pricings ↔ randomized mechanisms
Past Work

• Lots of work both in economic theory and in computer science
• 1 item: well-understood (also for many buyers) Myerson’81; randomization does not help
• 2 items: much more complicated; randomization can help

Work on
- Simple pricing schemes and their power/limitations
- Approximation of revenue
- Complexity

- Other models, e.g. unit-demand buyers, many buyers, correlated distributions
Past Work: Approximation

• Single item pricing: \( \Theta(\log n) \) approximation to optimal revenue [Hart-Nisan’12, Li-Yao’13]

• Grand bundle: \( O(1) \) approximation for IID distributions [LY13]

• Better of single item/grand bundle: 6-approximation for any (independent) distributions [Babaioff et al’15]

• Approximation schemes for subclasses of distributions [Daskalakis et al ’12, Cai-Huang’13]

• Reduction of many buyers to one, and \( O(1) \) approximation [Yao’15]
Past Work: Complexity

- Grand Bundle: Computing the best price for the grand bundle is \#P-hard [Daskalakis et al ’12]

- Partition pricing: Computing the best partition and prices is NP-hard. But PTAS for best revenue achievable by any partition mechanism [Rubinstein ’16]

- Randomized mechanisms: \#P-hard to compute the optimal solution/revenue [Daskalakis et al ’14]
Questions

• Is there an efficient algorithm that finds an optimal (deterministic) pricing?

• Is there such an algorithm when the instance has a “simple” optimal pricing?

• Is there a simple (i.e. easy to check) characterization of when single item pricing is optimal?

• For grand bundle pricing?
Results

• The optimal deterministic pricing problem is #P-hard, even if all distributions have support 2, and if the optimal is guaranteed to have a very simple form (we call it “discounted item pricing”): single item prices & price for grand bundle. Buyer can buy any subset for sum of its item prices or the grand bundle at its price
  - Also #P-hard to compute the optimal revenue.

• It is #P-hard to determine for a given instance
  - if single item pricing is optimal,
  - if grand bundle pricing is optimal
Results

- For IID distributions of support 2, the optimal revenue (even among randomized solutions) can be achieved by a discounted item pricing (i.e., single item prices & price for grand bundle), and it can be computed in polynomial time.

- For constant number of items (and any independent distributions), the problem can be also solved in polynomial time.
Integer Linear Program

- Let $D_i = \text{support of } F_i$ and $D = D_1 \times \ldots \times D_n$ (exponential size)
- Variables: $x_{v,1}, \ldots, x_{v,n} \in \{0,1\}$, $\pi_v$, $\forall v \in D$
- $(x_{v,1}, \ldots, x_{v,n}) = \text{characteristic vector of bundle bought for valuation } v$, $\pi_v$ its price

$$\max \sum_{v \in D} \pi_v \cdot \Pr[v]$$

Subject to

1. $\forall v \in D : x_{v,i} \in \{0,1\}$
2. $\forall v \in D : \sum_{i \in [n]} v_i \cdot x_{v,i} - \pi_v \geq 0$
3. $\forall w, v \in D : \sum_{i \in [n]} w_i \cdot x_{w,i} - \pi_w \geq \sum_{i \in [n]} w_i \cdot x_{v,i} - \pi_v$

($w$ does not envy the bundle of $v$)

- The LP $(x_{v,i} \in [0,1])$ models the optimal lottery problem
IID with support size 2

• Can assume wlog that support=\{1,b\} with b>1
  (If support=\{0,b\} then trivial: price all items at b. Otherwise rescale.)

• Let p = \Pr(\text{value}=b), 1-p = \Pr(\text{value}=1)

• Let \[ Q_i = \binom{n}{i} \cdot p^i (1-p)^{n-i} = \Pr(\text{i items at value } b) \]

• Lemma: There exists an integer \( k \in [0:n] \) such that
  \[ (n-i)Q_i - (b-1)(Q_{i+1} + ... + Q_n) \]
  is \(< 0\) for all \( i : 0 \leq i \leq k \), and is \( \geq 0 \) for all \( i : k \leq i \leq n \).

• Optimal Pricing \( S^* \): Price every item at \( b \), and offer the grand bundle at price \( kb+n-k \)
Proof Sketch

- Expected revenue of $S^*$ is $R^* = \sum_{1\leq i<k} bi \cdot Q_i + (kb + n - k) \sum_{k\leq i\leq n} Q_i$

- Since IID, the LP for the optimal lottery has a symmetric optimal solution [DW12], and the LP can be simplified to a more compact symmetric LP.

- Variables: $x_i, i=1,\ldots,n$ : probability of getting a value $b$ item when the valuation has $i$ items at $b$
  
  $y_i, i=0,\ldots,n-1$ : probability of getting a value 1 item when the valuation has $i$ items at $b$

  $\pi_i, i=0,\ldots,n$ : price of lottery for a valuation with $i$ items at $b$
Proof Sketch ctd.

The symmetric LP maximizes \( \sum_{i=0}^{n} \pi_i \cdot Q_i \)

Relax the LP by keeping only some of the constraints

1. \( 0 \leq x_i \leq 1 \) and \( 0 \leq y_i \leq 1 \) for all \( i \)

2. \( \pi_0 \leq n y_0 \) (the utility of the all-1 valuation is \( \geq 0 \))

3. For each \( i \in [n] \), the valuation \( w \) with \( w_j = b \) for \( j \leq i \) and \( w_j = 1 \) for \( j > i \) does not envy the lottery of the valuation \( v \) with \( v_j = b \) for \( j \leq i - 1 \) and \( w_j = 1 \) for \( j > i - 1 \)

\[
bix_i + (n - i) y_i - \pi_i \geq b(i - 1)x_{i-1} + (n - i + b)y_{i-1} - \pi_{i-1}
\]

• Combine the inequalities to upper bound every \( \pi_i \) in terms of the \( x, y \) variables
Proof Sketch ctd.

\[ \pi_0 \leq ny_0 \]
\[ \pi_i \leq bix_i + (n - i)y_i - (b - 1)(y_{i-1} + y_{i-2} + \ldots + y_1 + y_0) \]

Replacing in the objective function every \( \pi_i \) by its upper bound

\[ \rightarrow \text{linear form in } x_i, y_i \text{ that upper bounds optimal value} \]

- Coefficient of \( x_i \) is \( biQ_i > 0 \) \( \Rightarrow \) expression maximized if \( x_i = 1 \)
- Coefficient of \( y_i \) is \( (n-i)Q_i - (b-1)(Q_{i+1} + \ldots + Q_n) \), which is
  - \( < 0 \) if \( i < k \), and \( \geq 0 \) if \( i \geq k \) \( \Rightarrow \) expression maximized if we set \( y_i = 0 \) for all \( i < k \) and \( y_i = 1 \) for all \( i \geq k \)

- Substituting these values in the expression that upper bounds the objective function gives precisely \( R^* \)
#P-Hardness

- Reduction from the following problem, COMP.
  
  **Input:**
  1. Set $B$ of integers $0 < b_1 \leq b_2 \leq \ldots \leq b_n \leq 2^n$
  2. Subset $W \subseteq [n]$ of size $|W| = n/2$. Let $w = \sum_{i \in W} b_i$
  3. Integer $t$

  **Question:** Is the number of subsets $S \subseteq [n]$ of size $|S| = n/2$ such that $\sum_{i \in S} b_i \geq w$ at least $t$?
Construction

• n+1 items: n items ↔ b_i’s + special item

• First n items: almost iid with support {1, big}
  Item i: value 1 with probability p=1/2(h+1), where h=2^{2n}
  value h+1+b_i\delta , where \delta=1/2^{3n}, with probability 1-p

• Item n+1: support {\sigma, \sigma+\alpha}, where \sigma=1/p^n, \alpha=(n/2)h+w\delta (<<\sigma)
  value \sigma with probability (\alpha/(\sigma+\alpha))+\varepsilon for some \varepsilon=\varepsilon(t)=o(1/\sigma)
  value \sigma+\alpha with probability (\sigma/(\sigma+\alpha))-\varepsilon (=almost 1)
Two Candidate Solutions

- **Solution 1**: Grand bundle at price \( n + \sigma = \text{sum of low values} \)
  Equivalently, single item pricing with all prices = low values

- **Solution 2**: Discounted item pricing where all item prices = high values, and grand bundle price = \( n + \sigma + \alpha \)

Theorem: One of these two solutions is the unique optimal solution. \#P-hard to tell which one of the two.

Solution 1 is optimal if the answer to the COMP question is No
\( ( | \{ S \subset [n] \text{ of size } |S|=n/2 \text{ such that } \sum_{i \in S} b_i \geq w \} | < t ) \)

Solution 2 is optimal if the answer to the COMP question is Yes
\( ( | \{ S \subset [n] \text{ of size } |S|=n/2 \text{ such that } \sum_{i \in S} b_i \geq w \} | \geq t ) \)
Two Candidate Solutions

• **Solution 1:** Grand bundle at price $n+\sigma = \text{sum of low values}
  Equivalently, single item pricing with all prices= low values

• **Solution 2:** Discounted item pricing where all item
  prices=high values, and grand bundle price = $n + \sigma + \alpha$

Theorem: One of these two solutions is the unique optimal solution. #P-hard to tell which one of the two.

Corollaries:
1. #P-hard to tell if single item pricing is optimal
2. #P-hard to tell if grand bundle pricing is optimal
Proof Sketchy Outline

• Integer Linear Program, using the allocation variables $x_{v,i}$ and utility variables $u_v$ instead of price variables $\pi_v$

$$u_v = \sum_{i \in [n]} v_i \cdot x_{v,i} - \pi_v$$

• Denote a valuation by $(S, \sigma)$ (or $(S, \sigma+\alpha)$), for $S \subseteq [n]$ if $S$ = set of first $n$ items that have high value and $n+1$th item has value $\sigma$ (or $\sigma+\alpha$)

• In solution 1, all variables $x_{v,i} = 1$

  For $v = (S, \sigma)$, $u_v = \sum_{i \in S} h_i$

  For $v = (S, \sigma + \alpha)$, $u_v = \alpha + \sum_{i \in S} h_i$
Proof Sketchy Outline ctd.

• In Solution 2:
  1. If \( v = (S, \sigma + \alpha) \), all \( x_{v,i} = 1 \), \( u_v = \sum_{i \in S} h_i \)
  2. If \( v = (S, \sigma) \) and \( \sum_{i \in S} h_i \geq \alpha \) then all \( x_{v,i} = 1 \), \( u_v = \sum_{i \in S} h_i - \alpha \)
  3. If \( v = (S, \sigma) \) and \( \sum_{i \in S} h_i < \alpha \) then \( x_{v,i} = 1 \) for all \( i \in S \),
     \( x_{v,i} = 0 \) for all \( i \notin S \) and for \( i = n + 1 \), and \( u_v = 0 \)

- Every S with \( |S| > n/2 \) satisfies case 2,
- every S with \( |S| < n/2 \) satisfies case 3,
- a set S with \( |S| = n/2 \) satisfies case 2 if \( \sum_{i \in S} b_i \geq \nu \)
  and case 3 otherwise
Proof Sketchy Outline ctd.

- Relaxed ILP – keep only a subset of the envy constraints
  - $(S, \sigma+\alpha)$ does not envy $(\emptyset, \sigma+\alpha)$, for all $S \neq \emptyset$
  - $(\emptyset, \sigma+\alpha)$ does not envy $(S, \sigma)$, and vice-versa, for all $S \subseteq [n]$
  - for all $T \subset S \subseteq [n]$, $(S, \sigma)$ does not envy $(T, \sigma)$

- Long sequence of lemmas shows that the optimal solution to the relaxed ILP must be either solution 1 or solution 2
  - For $v = (\emptyset, \sigma+\alpha)$, if $x_{v,n+1} = 0$ then it must be Solution 1,
    - if $x_{v,n+1} = 1$ then it must be Solution 2
Constant Number of Items

• #items = k = constant, support size m for each item
• V = set of $m^k$ possible valuation vectors (polynomial)
• d=2$^k$ possible bundles (constant)

• Space $R_+^d$ of possible price vectors $p$ for the bundles partitioned by hyperplanes into cells such that \( \forall \text{cell } C \) \( \forall \text{ valuation } v \) buys the same bundle for all $p \in C$

• Hyperplanes:

\[
\forall v \in V \ \forall j \in [d]: \sum_{l \in B_j} v_l - p_j = 0
\]

\[
\forall v \in V \ \forall j, j' \in [d]: \sum_{l \in B_j} v_l - p_j = \sum_{l \in B_{j'}} v_l - p_{j'}
\]

\[
\forall j, j' \in [d]: p_j = p_{j'}
\]
Constant Number of Items

• The supremum revenue for price vectors in C is given by an LP, and is achieved at a vertex of C.

⇒ Optimum overall is achieved at a vertex of the subdivision

• Polynomial number of hyperplanes, constant dimension d
  ⇒ polynomial number of vertices.

• Try them all and pick best.
Conclusions

• Showed that the optimal (deterministic) pricing problem is hard, and this holds even when the optimal solution is very simple: single item pricing + discount for grand bundle

• Can we find a polynomial time approximation scheme, or can we rule it out?
  When there is a simple optimal solution?

• IID case?
  Is there a PTAS?