Multi-dimensional Mechanism Design via Random Order Contention Resolution Schemes

MAREK ADAMCZYK
University of Warsaw
and
MICHAŁ WŁODARCZYK
University of Warsaw

Already in 1981 Myerson gave a characterization of the optimal mechanism for a single parameter Bayesian mechanism design. However, till today we have no idea for how such a characterization for the multi-dimensional setting could even look like. Moreover, it wasn't until no that long time ago that we could not develop mechanisms for such setting with any reasonable and provable performance guarantees. The seminal work of [Chawla et al. 2009] on sequential posted pricing mechanisms gave us an approach for approximately solving the Bayesian multi-parameter unit-demand mechanism design problem (BMUMD). The paper left open the question on how to obtain a constant approximation for the matroid setting. Two mathematically beautiful results from combinatorial optimization under uncertainty where devised in order to answer this question. First, Kleinberg and Weinberg in 2011 extended the classical Prophet Inequality result into the matroid setting to give a 2-approximation for BMUMD for a single matroid setting. Second, Feldman, Svensson and Zenklusen in 2016 adapted the Contention Resolution Scheme framework for online settings. We add to this line of work by considering the Contention Resolution Scheme framework in the random order setting. The most impressive implication of this research are the new algorithms for BMUMD which improve the previous results in the multimatroid setting. Although the range of implications of the CR Scheme framework in the random order is reasonably wide, we shall focus in this letter on presenting only the single matroid setting and how it is connected to BMUMD.

Categories and Subject Descriptors: F.2.2 [**ANALYSIS OF ALGORITHMS AND PROBLEM COMPLEXITY**]: Nonnumerical Algorithms and Problems—*Computations on discrete structures*

General Terms: Algorithms, Theory, Economics

Additional Key Words and Phrases: contention resolution schemes, stochastic optimization, approximation algorithms

1. INTRODUCTION

Contention resolution schemes have proven to be an incredibly powerful concept which allows to tackle a broad class of problems. The framework has been initially designed to handle submodular optimization under various types of constraints, that is, intersections of matroids (and more generally exchange systems), knapsacks, and unsplittable flows on trees. Later on, it turned out that this framework perfectly extends to optimization under uncertainty, like stochastic probing and online selection problems, which further can be applied to mechanism design.

We add to this line of work by showing how to create contention resolution

Authors' addresses: `marek.adamczyk@mimuw.edu.pl`, `michal.wlodarczyk@mimuw.edu.pl`

schemes for intersection of matroids and knapsacks when we work in the random order setting. More precisely, we do know the whole universe of elements in advance, but they appear in an order given by a random permutation. Upon arrival we need to irrevocably decide whether to take an element or not. The main application of our framework is a $k+4+\varepsilon$ approximation ratio for the Bayesian multi-parameter unit-demand mechanism design under the constraint of k matroids intersection, which improves upon the previous bounds of $4k-2$ and $e(k+1)$.

Contention resolution schemes

Let us start with an illustrative problem. Consider a matroid $\mathcal{M} = (E, \mathcal{I})$ and a fractional solution x from its polytope. Suppose we are given a weight vector $w : E \mapsto \mathbb{R}_+$, and we look for an algorithm that returns an independent set $S \in \mathcal{I}$ such that $\sum_{e \in S} w_e \geq c \cdot \sum_{e \in E} w_e x_e$ for some constant $c < 1$. The idea is to settle for a randomized algorithm and demand that every element is taken into S with probability at least $c \cdot x_e$. Such a property would immediately entail the desired guarantee.

How to design an algorithm returning S such that $\mathbb{P}[e \in S] \geq c \cdot x_e$? Chekuri et al. [Chekuri et al. 2014] presented a framework of contention resolution schemes (CR schemes) which address this problem, among other applications. The idea is to first draw a random set $R(x)$ such that $\mathbb{P}[e \in R(x)] = x_e$ for each $e \in E$ independently, and afterwards – since $R(x)$ is most likely not an independent set in $\mathcal{I}$ – to drop some elements from $R(x)$ to meet the feasibility constraint, that is, to resolve the contention between the elements.

Our contribution. Simply speaking, we show that the above problem can be solved also if we work in a random order model, i.e., when elements of E appear to us according to a uniformly random permutation, and upon arrival we need to make an irrevocable decision of whether to take an element or not.

In its full generality Chekuri et al. were dealing not only with matroids but arbitrary intersections of matroids, knapsacks, exchange systems, and unsplittable flow on trees. They were also maximizing not only linear functions, but non-negative submodular functions as well. We do so as well, restricted to intersections of matroid and knapsack constraints. For a single matroid and a linear objective, Chekuri et al. obtained an approximation (the constant c) of $1-\frac{1}{e}$, while we get $\frac{1}{2}$. However, for intersection of k matroids, starting with $k \geq 2$, we obtain a better bound of $\frac{1}{k+1}$, improving upon theirs $\frac{1}{e \cdot k + o(k)}$, even though we work in a more restrictive model. The following is the most important implication of our framework.

Theorem 1.1. *There exists a random-order CR scheme for intersection of k matroids with $c = \frac{1}{k+1}$.*

And thanks to it we can obtain the result for the BMUMD problem.

Bayesian multi-parameter unit-demand mechanism design

Consider the following mechanism design problem. There are n agents and a single seller providing a set of services. The agent i is interested in buying the i-th service and values its as v_i, which is drawn independently from a distribution D_i. Such a setting is called single-parameter. The valuation v_i is private, but the distribution D_i is known in advance. The seller can provide only a subset of services, that belongs to a system $\mathcal{I} \in 2^{[n]}$, which is specified by feasibility constraints. A mechanism accepts bids of agents, decides on subset of agents to serve, and sets individual prices for the service. A mechanism is called truthful if agents are motivated to bid their true valuations. Myerson's theory of virtual valuations yields truthful mechanisms that maximize the expected revenue of a seller [Myerson 1981], although they sometimes might be impractical [Ausubel and Milgrom 2006]. On the other hand, practical mechanisms are often non-truthful [Ausubel and Milgrom 2006]. The Sequential Posted Pricing Mechanism (SPM) introduced by Chawla et al. [Chawla et al. 2010] gives a nice trade-off – it is truthful, simple to implement, and gives near-optimal revenue. An SPM offers each agent a 'take-it-or-leave-it' price for a service. After refusal the service shall not be provided, so it is easy to see that an SPM is indeed a truthful mechanism.

The paragraph above concerns only the single-parameter setup. In the Bayesian multi-parameter unit-demand mechanism design (BMUMD for short), we have n buyers and one seller. The seller offers a number of different services indexed by set $\mathcal{J}$. The set $\mathcal{J}$ is partitioned into groups $\mathcal{J}_i$, with the services in $\mathcal{J}_i$ being targeted by agent i. Each agent i is interested in getting any one of the services in $\mathcal{J}_i$, i.e., agents are unit-demand. Agent i has value v_j for service $j \in \mathcal{J}_i$. Value v_j is independent of all other values and is drawn from distribution D_j. Once again the seller faces a feasibility constraint specified by a set system $\mathcal{I} \subseteq 2^J$.

Unlike single-parameter setup, this problem is not solvable efficiently by the well-established Myerson's approach. The paper of Chawla et al. [Chawla et al. 2010] launched a line of work in obtaining approximate results for the multi-parameter setup, by suggesting a possible avenue of a solution via the so-called Oblivious Posted Price mechanisms. One would have to first embed the multi-parameter problem into a single-parameter one, and later to ensure that the algorithm would work if the items are presented in an adversary order. Kleinberg and Weinberg [Kleinberg and Weinberg 2012] solved the BMUMD problem for matroid environments with approximation of $4k - 2$ for intersection of k matroids (with 2-approximation for a single matroid), but they have not used the Oblivious Posted Price mechanisms. Feldman et al. [Feldman et al. 2016] devised the first Oblivious Posted Price mechanisms and obtained an $ek + o(k)$ approximation for the intersection of k matroids.

Our contribution. We observe that the Oblivious Posted Price is an overly demanding notion, and we need to handle the oblivious order only when looking at the items of a given client, but there is no need to restrict the order of clients. In our algorithm we randomly shuffle clients, but cannot make assumption on the client's choice. This hybrid approach is what allows us to obtain improved bounds. For $k = 2$ we match up to ε the 6-approximation of Kleinberg and Weinberg [Kleinberg and Weinberg 2012], but starting from $k \geq 3$ our ratios are better; for $k = 3$ we get $7 + \varepsilon$ improving over 9.48 of Feldman et al. [Feldman et al. 2016].

THEOREM 1.2. *Bayesian multi-parameter unit-demand mechanism design over k matroid constraints admits a $(k + 4 + \varepsilon)$ approximation for any $\varepsilon > 0$.*

2. CONTENTION RESOLUTION SCHEME FOR A SINGLE MATROID

Consider a uniform matroid $\mathcal{M} = \left(E, \mathcal{I} \subseteq \binom{E}{k}\right)$ of rank k. We consider optimization over the *matroid polytope* $\mathcal{P}(\mathcal{M}) = \left\{x \in \mathbb{R}^E_{\geq 0} \;\middle|\; \sum_{e \in E} x_e \leq k\right\}$.

We shall use the following two properties. Both facts hold for any matroid.

Fact 2.1. *We can represent any $x \in \mathcal{P}(\mathcal{M})$ as $x = \sum_{i=1}^m \beta_i \cdot \mathbf{1}_{B_i}$, where $B_1, \ldots, B_m \in \mathcal{I}$ and $\beta_1, \ldots, \beta_m$ are non-negative weights such that $\sum_{i=1}^m \beta_i = 1$. in $\mathcal{P}(\mathcal{M})$.*

The following holds for any matroid, although in the case of a uniform matroid this statement is almost trivial. It is a delicate generalization of the Basis Exchange lemma [Schrijver 2003].

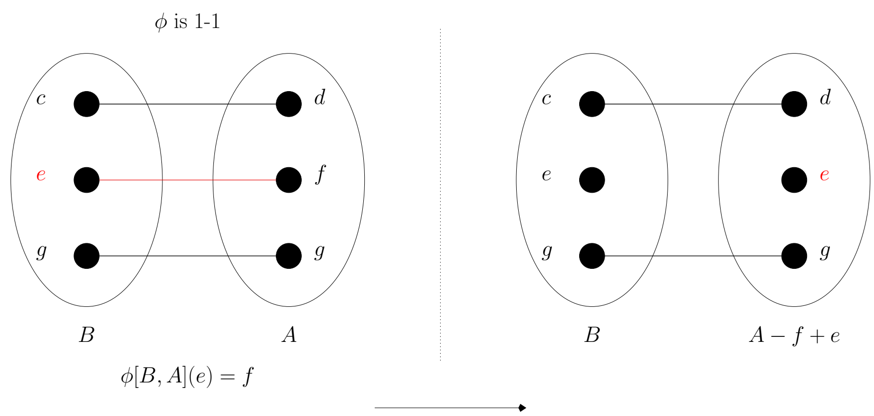

Fig. 1: A swap guided by an exchange mapping. A was an independent set, and after a swap $A - f + e$ is still independent.

Fact 2.2. *Let $A, B \in \mathcal{I}$ be two independent sets of matroid $\mathcal{M} = \left(E, \mathcal{I} \subseteq \binom{E}{k}\right)$. We can find an assignment $\phi[A, B] : A \mapsto B \cup \{\perp\}$ such that:*

(1) $\phi[A, B](e) = e$ for every $e \in A \cap B$,

(2) for each $f \in B$ there exists at most one $e \in A$ for which $\phi[A, B](e) = f$,

(3) for $e \in A \setminus B$, if $\phi[A, B](e) = \perp$, then $B + e \in \mathcal{I}$, otherwise $B - \phi[A, B](e) + e \in \mathcal{I}$.

The procedure. The procedure is shown in the Algorithm 1. What shall be important in the analysis of the approximation is that we do not assume that we know the whole set $R(x)$ in advance, but rather we reveal the set $R(x)$ after each step in we get to know what elements belongs to $R(x)$ only in line 7. We do it just to ease the analysis by using the principle of deferred decisions.

Remember that the input of the algorithm is a point $x \in \mathbb{P}[\mathcal{M}]$ and a random set $R(x)$. The algorithm's idea is quite simple.

(1) We start with the convex decomposition of x into $\sum_i \beta_i \cdot B_i$.

(2) In each step we take a random element e from elements we did not consider yet. If the element is not *blocked* yet (whatever it means at the moment), we take it into the solution S which we gradually construct.

(3) We go over each set B_i of the convex decomposition, and we insert e into B_i while removing some other element f from B_i. We do it according to the mapping from Fact 2.2, which ensures that after every such swap B_i is still an independent set. The mapping has to be calculated from scratch at every iteration. This yields the correctness of the procedure, because the solution S always belongs to each B_i, which means that S is independent as well.

However, we have to say something about how we block elements, because this is what affects the approximation guarantee. For this, given an element e we consider all sets B_i to which e belongs. Since $\sum_{i:e\in B_i} \beta_i = x_e$, we choose randomly exactly one B_i according to probability $\frac{\beta_i}{x_e}$. We call this set a *critical set* and denote it by C_e.

Now the blocking event is the removal of e from C_e. If such an event happens, then we shall discard the element, call it *blocked*, and not take it into the solution.

The whole procedure is presented in Algorithm 1

Algorithm 1 Random order contention resolution scheme for a single matroid

Given: matroid $\mathcal{M}$, $x \in \mathbb{P}[\mathcal{M}]$, and a random set $R(x)$ such that $\mathbb{P}[e \in R(x)] = x_e$ for each $e \in E$ independently
decompose x into $\sum_i \beta_i \cdot B_i$
for each element e choose a set $B_i : e \in B_i$ with probability $\frac{\beta_i}{x_e}$; call it a *critical* set, and denote it by C_e
$S \leftarrow \emptyset$
for each $e \in E$ in a random order
    if $e \notin R(x)$ **then**
        continue
    if e is not blocked (i.e., still $e \in C_e$) **then**
        $S \leftarrow S \cup \{e\}$
        for each set B_i such that $e \notin B_i$ **do**
            $B_i \leftarrow B_i + e - \phi[C_e, B_i](e)$
return S

Approximation guarantee. Suppose we are after t steps. Consider an element e. Suppose that 1) $e \in R(x)$, and 2) suppose that it is still not blocked. If so, then the probability that we shall take e into the solution S in step $t+1$ is just equal to the probability of picking e among all remaining elements, and this is just $\frac{1}{n-t}$.

What is the probability that we shall block e (meaning e will be removed from its critical set C_e)?

Let us look more precisely at the probability that a given set B_i will cause a removal of e from C_e. For a given B_i there exists at most one $f \in B_i$ such that $\phi[B_i, C_e](f) = e$. This f (if it exists) in step $t+1$:

– is chosen with probability $\frac{1}{n-t}$,

– it belongs to $R(x)$ with probability x_f,
– it has chosen B_i as its critical set with probability β_i/x_f.

Hence the probability that B_i is the cause of removing e from C_e is at most $\frac{1}{n-t}x_f \cdot \frac{\beta_i}{x_f} = \frac{\beta_i}{n-t}$. Summing over all B_i we get that the probability of removing e from C_e in step $t+1$ is at most $\sum_{i:e\notin B_i} \frac{\beta_i}{n-t} \leq \frac{1}{n-t}$. Hence

$$\mathbb{P}\left[e \text{ blocked in step } t+1\right] \leq \frac{1}{n-t} = \mathbb{P}\left[e \text{ taken in step } t+1\right].$$

This one step inequality implies a global inequality saying that at the end of the whole process $\mathbb{P}[e \text{ blocked}] \leq \mathbb{P}[e \text{ taken}]$ (although this needs a bit more formalization using the martingale framework, which we shall omit).

Since the fate of e is either to be taken or to be blocked, one of the two has to happen at some point, i.e., $\mathbb{P}[e \text{ taken}] + \mathbb{P}[e \text{ blocked}] = 1$. And so this finally implies that that $\mathbb{P}[e \text{ taken}] \geq \frac{1}{2}$.

3. MULTI-PARAMETER MECHANISM DESIGN

Recall that each client $i \in \mathcal{I}$ is interested in purchasing one service from $\mathcal{J}_i$ and their valuation of an item $c \in \mathcal{J}_i$ is modeled by a random variable v_c, independent of other valuations, with a known distribution D_c. One can think that the distribution D_c is always discrete.

Bounding by auction with copies. Imagine a setting where for each item $c \in \mathcal{J}_i$ we create an independent copy-client c interested solely in this item. The new instance has the same constraint system as the original one plus additional partition matroid. We rely on the crucial lemma from [Chawla et al. 2010], saying that the optimal revenue in the new instance can be only greater because the competition increases.

This observation allows us to obtain an LP upperbound for the true OPT. The linear program BMUMD-LP [Gupta and Nagarajan 2013] models the auction with copy-clients, which is single-parameter. C denotes the set of copy-clients, which is equivalent to the set of items, and $\mathcal{P}$ is the polytope of the constraint system.

$$\begin{aligned} \max \quad & \textstyle\sum_{c\in C}\sum_p x_{c,p}\cdot p\cdot \mathbb{P}\left[v_c \geq p\right] \qquad (\text{BMUMD-LP}) \\ \text{s.t.} \quad & \left(\textstyle\sum_p x_{c,p}\cdot \mathbb{P}\left[v_c \geq p\right]\right)_{c\in C} \in \mathcal{P} \\ & \textstyle\sum_p x_{c,p} \leq 1 \qquad \forall c \in C \\ & \textstyle\sum_{c\in \mathcal{J}_i}\sum_p x_{c,p}\cdot \mathbb{P}\left[v_c \geq p\right] \leq 1 \quad \forall i \in \mathcal{I}. \end{aligned}$$

THEOREM 3.1. *The optimal value of* BMUMD-LP *is an upper bound for the maximal revenue in the multi-parameter auction.*

To give a grip with the previous result: the object $\left(\sum_p x_{c,p}\cdot \mathbb{P}\left[v_c \geq p\right]\right)_{c\in C} \in \mathcal{P}$ is now the vector which we decompose into a convex combination of independent sets.

Single client menu. The algorithm scans clients in random order, and presents a price menu to each client, from which the client picks one item which gives him the highest utility, or resigns from choosing if all utilities are negative. Such a procedure clearly yields a truthful mechanism. Of course the menu presents only the items which are still not blocked at the moment of approaching a client.

We omit the details of how the subroutine which constructs the menu is implemented, but we shall state the most important property of it. Suppose that item c is still not blocked when we approach the client and we want a guarantee on the probability that c will be sold at price p. Then for any $\varepsilon > 0$ we can construct a menu with the following property (for all p, c).

$$\frac{1}{4} \cdot x_{c,p} \cdot \mathbb{P}\left[v_c \geq p\right] \leq \mathbb{P}\left[\text{client takes item } c \text{ at price } p\right] \leq \left(\frac{1}{4} + \varepsilon\right) x_{c,p} \cdot \mathbb{P}\left[v_c \geq p\right]$$

The upperbound with $\left(\frac{1}{4} + \varepsilon\right)$ in the second inequality is crucial to obtain the approximation ratio $k + 4 + \varepsilon$ stated in Theorem 1.2. Without the subroutine we would only have an upperbound of $x_{c,p} \cdot \mathbb{P}\left[v_c \geq p\right]$, which would follow easily from the way the algorithm works, but it would be not enough for our needs.

Algorithm. With the subroutine to handle a single client, we are ready to state our algorithm for the BMUMD problem.

Algorithm 2 Auction mechanism

1: solve the BMUMD-LP
2: **for** each client $i \in I$ in random order **do**
3: construct a menu from the non-blocked items in J_i
4: **if** client i chooses c **then**
5: go over all sets of the decomposition and update them according to item c

4. OPEN PROBLEM

The results we obtained allowed for a Contention Resolution Scheme on k matroids such that $\mathbb{P}\left[e \text{ in the solution}\right] \geq \frac{1}{k+1} \cdot x_e$. What can be interesting is that it improved the results in an offline setting, even though it works in a random order. Given the potential of the approach, a natural question appears thus — can we apply the same combinatorial insight that uses the exchange mappings to construct Online Contention Resolution Schemes that were considered by [Feldman et al. 2016]. An interesting result would be to obtain the same guarantee of $\frac{1}{k+1}x_e$ in this more restricted setup. Such a result could be used to construct a Matroid Prophet Inequality as introduced by [Kleinberg and Weinberg 2012]. With a guarantee of $\frac{1}{k+1}x_e$ it would match their result for a single matroid, yielding a 2-approximation. However, for $k \geq 3$ it would already improve over their $4k - 2$ and $e \cdot k$ of [Feldman et al. 2016].

REFERENCES

Ausubel, L. M. and Milgrom, P. 2006. The lovely but lonely vickrey auction. In *Combinatorial Auctions, chapter 1.* MIT Press.

Chawla, S., Hartline, J. D., Malec, D. L., and Sivan, B. 2010. Multi-parameter mechanism design and sequential posted pricing. In *STOC.*

Chekuri, C., Vondrák, J., and Zenklusen, R. 2014. Submodular function maximization via the multilinear relaxation and contention resolution schemes. *SIAM J. Comput. 43,* 6, 1831–1879.

FELDMAN, M., SVENSSON, O., AND ZENKLUSEN, R. 2016. Online contention resolution schemes. In *Proceedings of the Twenty-Seventh Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2016, Arlington, VA, USA, January 10-12, 2016*. 1014–1033.

GUPTA, A. AND NAGARAJAN, V. 2013. A stochastic probing problem with applications. In *IPCO*.

KLEINBERG, R. AND WEINBERG, S. M. 2012. Matroid prophet inequalities. In *STOC*. 123–136.

MYERSON, R. B. 1981. Optimal auction design. *Math. Oper. Res. 6,* 1 (Feb.), 58–73.

SCHRIJVER, A. 2003. *Combinatorial Optimization - Polyhedra and Efficiency*.