Submodular welfare maximization

1 Introduction

We study different variants of the welfare maximization problem in combinatorial auctions. Assume there are n buyers interested in m items. Indeed they have different valuations on each item, i.e. buyer i has a valuation of $w_i(j)$ on item j. In the simplest model, a collection of items S worths $\sum_{j \in S} w_i(j)$ for buyer i; However, in practice, collection of items might have different values than the sum of values of individual items. Therefore, in a more accurate model, we should take this into account and define the valuation of each buyer as a real valued function on the family of subsets of items, i.e. $w_i : 2^{[m]} \to \mathbb{R}_+$. Our goal is to partition these items among buyers so as to maximize the total welfare. More precisely, we partition the set of items into subsets $S_1, \ldots, S_n$ and give bundle S_i to buyer i in order to maximize $\sum_{i=1}^n w_i(S_i)$.

Based on the above discussion, not only we need the valuation of each person on each item, but also we need to present the valuations on subsets of items which requires $O(n2^m)$ bits of data to be stored as the input of the algorithm. This huge amount of data is exponential in the number of items, which could make practical difficulties especially when the number of items, m, is very large. Hence, instead of presenting the whole data to the algorithm, we employ an oracle who answers queries when asked. Usually, two types of oracles have been considered:

Value oracle this type of oracle answers the questions of the type "What is the value of $w_i(S)$ for a person $1 \le i \le n$ and a collection of items S?" This is obviously the simplest type of oracle who just looks at the database of valuations with zero computational overload.

Demand oracle unlike value oracle which does not compute anything, in this case we have access to a more powerful oracle which has computational power to solve a maximization problem. In fact, we have access to a powerful black box which can solve the following problem for us in zero time: assume $p : [m] \to \mathbb{R}$ is a price function on items, find the set S which maximizes $w_i(S) - \sum_{j \in S} p_j$.

Even if we employ the powerful demand oracle, the problem could be very hard to compute in general. For an example, consider the scenario in which each buyer is only interested in an specific subset of items, denoted by T_i for buyer i. Therefore he is satisfied when he collects all the items in T_i, paying no attention to additional items, i.e. $w_i(S) = 1$ if $T_i \subseteq S$ and 0 otherwise. In this case, both value and demand queries are trivial, but the problem is equivalent to the set packing problem which has no $m^{-1/2+\epsilon}$-approximation unless $P = NP$ [Hås99, Zuc07]. Based on this discussion, we restrict the set of permissible utility functions to obtain non-trivial positive results. In particular, we add two assumptions which are reasonable in practice.

The first assumption is monotonicity. We expect a person to be more satisfied when he obtains a larger set of items, i.e. we do not have negative valuations or "disgusting" items. This translates to monotonicity of functions w_i:

Definition 1.1. A function $f : 2^X \to \mathbb{R}$ is monotone if $f(S) \leq f(T)$ whenever $S \subseteq T$.

The other assumption is related to the improvement in gains by giving extra items to players. In practice, a reach person is less excited by obtaining a Porsche than a relatively poor person. Therefore, it is reasonable to assume that adding an extra item to a large set is less attractive than adding it to a relatively smaller one. This translates to submodularity of functions. A function f defied on the family of subsets of a set is called submodular if

$$f(S \cup \{j\}) \geq f(T \cup \{j\}) \qquad S \subset T.$$

It could be shown that this definition of submodular functions is equivalent to the following:

Definition 1.2. A discrete function $f : 2^X \to \mathbb{R}$ is submodular if $f(S \cup T) + f(S \cap T) \leq f(S) + f(T)$. Usually we also assume that $f(\emptyset) = 0$.

Adding these two assumptions leads us to Submodular Welfare Problem. Conclusively,

Definition 1.3 (Submodular welfare maximization). m buyers and n items are given. Each buyer has a monotone submodular valuation $w_i : 2^{[m]} \to \mathbb{R}_+$ which is his interest in different subsets of items . The goal is to partition items into disjoint sets $S_1, S_2, \cdots, S_n$ in order to maximize $\sum_{i=1}^{n} w_i(S_i)$.

Submodular functions appear in other areas like rank functions of matroids, in covering problems, graph cut problems and facility location problems [Edm70, Lov, Sch03]. It could be shown that minimization of submodular functions could be done in polynomial time ([IFF01, Sch00]), however maximization of such functions is typically NP-hard. First studies on maximization of monotone submodular functions is due to Nemhauser, Wolsey and Fisher in the 1970s [NWF78, FNW78, NW78].

2 Algorithmic results

We can categorize welfare maximization problem based on two parameters: the first one is different classes of valuation functions, w_i. Usually assumptions on w_i are monotonicity and submodularity. The other parameter is the order in which items arrive. We may assume that the division process takes place when all the items and valuations are known (offline) items arrive one by one and valuations are known for so far received items and each item should be assigned upon arrival (online). Results on offline and online welfare maximization are given in the following. It should be noted that we do not seek the exact solution, instead an approximation of the solution suffices, i.e. a solution that is within a coefficient of the answer is acceptable.

2.1 Offline Model

In offline model, we should assign each item to a player after receiving the whole set of items. In [Von08], a randomized continuous greedy algorithm for the submodular welfare problem is derived which is a $(1-1/e)-$ approximation. Interestingly, in the special case when the valuations of players are identical, the optimal answer is obtained by uniform random solution. It is shown using information theoretic lower bounds that solving the problem more accurately (with better approximation factor), an exponential number of value queries is necessary. Furthermore, in this paper the problem is analyzed for the two classes *subadditive* and *superadditive* valuation functions. A set function f is said to be subadditive iff $f(S) + f(T) \geq f(S \cup T)$ and is said to be superadditive iff for disjoint sets S and T, $f(S) + f(T) \leq f(S \cup T)$. Note that subadditivity is similar to submodularity, however, submodularity is a stronger condition. It is shown that approximation factors $\frac{1}{\sqrt{m}}$ and $\frac{\sqrt{\log m}}{m}$ respectively for subadditive and superadditive valuations are the best approximation factors which can be obtained by asking a polynomial number of value queries, i.e. better approximation factors require asking super polynomial number of queries. This shows that the above approximation factors are the best possible ones.

2.2 Online Model

Definition 2.1. m items are arriving online, and each item should be allocated upon arrival to one of n agents whose interest in different subsets of items is expressed by valuation functions $w_i : 2^{[m]} \to \mathbb{R}_+$. Also it is assumed that we only know the agents' valuations on items arrived so far. The goal is to maximize $\sum_{i=1}^{n} w_i(Si)$ where S_i is the set of items allocated to agent i.

Since we should decide immediately upon arrival which person to give the current item, the simplest framework is that we give the current item to the person who gets excited the most, i.e. we assign the arrived item to the player

whose welfare increases the most. Fisher, Nemhauser and Wolsey who worked on problems involving maximization of submodular functions, introduced this greedy algorithm [NWF78, FNW78, NW78]. This greedy strategy is shown to be 1/2-competitive when valuation functions are monotone and submodular.

An special case of this problem is reduced to online bipartite matching analyzed by Karp, Vazirani and Vazirani [KVV90]. In the online bipartite matching problem, one vertex of the first part of the graph as well as its connections is given at a time, and upon receiving this information, we should decide which vertex of the second part to match to it so as to maximize the size of the matching at the end of the day. The following restriction on valuation functions reduces online welfare maximization to online bipartite matching; Assume each agent i is completely satisfied by only one item in an specific set $N(i)$, i.e. $w_i(S) = \min\{|S \cap N(i)|, 1\}$. Having this assumption, we form a bipartite graph in which the first part and second parts are representatives of the agents and the items respectively. Connect an agent i to his $N(i)$ interested items. Then online welfare maximization reduced to online bipartite matching, for which there exists an elegant $(1 - 1/e)$-competitive randomized algorithm in [KVV90]. Note that this is an improvement on the greedy 1/2-competitive solution.

3   Truthful Mechanism Design

So far we have assumed that the valuations of players are known to us when allocating the items. However, in practice, we might not have access to actual valuations. In this setting the aim of the mechanism designer is to design a computationally efficient mechanism in which he hopes that the agents are truthful and that achieves with an approximation factor the optimal solution found by the former version of the problem in which all the information is provided in advance. Now we give a formal definition of what it means for mechanism designer to hope that agents are incentive compatible ([DRY11]):

Definition 3.1. A mechanism with allocation and payment rules $\mathcal{A}$ and p is *truthfull-in-expectation* if every player always maximizes it's expected payoff by truthfully reporting it's valuation function meaning that

$$\mathbb{E}[v_i(\mathcal{A}(v)) - p_i(v)] \geq \mathbb{E}[v_i(\mathcal{A}(v'_i, v_{-i})) - p_i(v'_i, v_{-i})] \tag{3.1}$$

for every player i, (true) valuation function v_i, (reported) valuation function v'_i, and (reported) valuation functions v_{-i} of the other players. The expectation in (3.1) is over the coin flips of the mechanism.

In [DRY11], a $(1 - 1/e)$-approximation truthful in expectation mechanism for coverage valuations is derived. Also, it is shown, assuming $P! = NP$, even for known and explicitly given coverage valuations, the approximation factor could not be improved. However, for submodular valuations, no truthful-in-expectation mechanism exists [DV11].

It is worth noting that without incentive-compatibility constraints, the welfare maximization problem with submodular bidder valuations is completely solved. As was mentioned before, a $(1 - 1/e)$-approximation algorithm for the problem exists [Von08].

4 Applications of submodular optimization

As was stated before, submodular optimization has applications in other problems and areas. In this section we give some examples of such applications. In 4.1, we consider applications of submodular optimization in social network problems in which the most influential subgroup of a society is of our concern. In 4.2 we analyze the problem of finding correspondent words in translated sentences called *word alignment*. In 4.3, we want to summarize a number of related documents. In all of these problem, an optimization problem is introduced for which the correspondent objective function is submodular and efficient maximization methods for this class of functions are used to solve the problem.

4.1 Social Network

Assume you have a product and you want to advertise it in Facebook, but you have a limited budget and hence you can present your advertisement to a limited number of Facebook users. Indeed, some users are more social and effective in the society while some others are isolated. Social users can help distribute the information while the capability of isolated individuals for doing so is small. Therefore it is reasonable to present your product to the most influential nodes of the society. Domninigos and Richardson first posed this problem as a fundamental algorithmic problem. The optimization problem of finding such nodes is NP-hard in general, however acceptable approximation solutions exist for the problem. Using an analysis framework based on submodular functions, it is shown that a natural greedy algorithm, one can find a solution which is within 63% of the optimal solution of the problem [KKT03].

4.2 Word Alignment

Assume that we have a sentence in English as well as its French translation and we want to see the correspondence of the words in the two sentences so that we can know which French words correspond to each English word in the sentence. In general, we can model any correspondence between the two sentences by a bipartite graph. The nodes in the first part are the words in English while the nodes in the second part are the words in French. Each bipartite graph is uniquely determined by the set of its edges (say A) which is a subset of the collection of all the edges (say V). Assume that we have a function f which measures the quality of a correspondence $A \subset V$ as a real

nonnegative number. The words alignment problem is equivalent to maximizing $f : 2^V \to \mathbb{R}_+$ under certain constraints. When f is monotone and submodular, near-optimal solutions for this problem exit [LB11].

4.3   Document summarization

Assume we have a collection of related documents which we want to summarize. The way to approach this problem is to define appropriate objective functions and optimization problems. This problem is called *multi-document summarization*. A number of appropriate objective functions could be found in [CG98, FV0, TO09, RFHT10, SL10]. It is seen that these well-established summarization methods correspond to submodular function optimization [LB11]. Therefore, simple greedy algorithms for monotone submodular function maximization could be used for this problem to guarantee a summarization which is almost as good as the best possible solution obtained by explicitly solving the optimization problem.

References

[CG98]   Jaime G. Carbonell and Jade Goldstein. The Use of MMR, Diversity-Based Reranking for Reordering Documents and Producing Summaries. In *SIGIR*, pages 335–336. ACM, 1998.

[DRY11]   Shaddin Dughmi, Tim Roughgarden, and Qiqi Yan. From convex optimization to randomized mechanisms: toward optimal combinatorial auctions. In *Proceedings of the 43rd annual ACM symposium on Theory of computing*, STOC '11, pages 149–158, New York, NY, USA, 2011. ACM.

[DV11]   Shaddin Dughmi and Jan Vondrák. Limitations of Randomized Mechanisms for Combinatorial Auctions. In Rafail Ostrovsky, editor, *FOCS*, pages 502–511. IEEE, 2011.

[Edm70]   Jack Edmonds. Submodular Functions, Matroids, and Certain Polyhedra. *Combinatorial Structures and Their Applications*, pages 69–87, 1970.

[FNW78]   M. L. Fisher, G. L. Nemhauser, and L. A. Wolsey. An analysis of approximations for maximizing submodular set functions—II. *Mathematical Programming Studies*, 8:73–87, 1978.

[FV0]   Event-based extractive summarization. volume 111.

[Hås99]   Johan Håstad. Clique is hard to approximate within n1-epsilon. *Acta Mathematica*, 18:105–142, 1999.

[IFF01]   Satoru Iwata, Lisa Fleischer, and Satoru Fujishige. A combinatorial strongly polynomial algorithm for minimizing submodular functions. *J. ACM*, 48(4):761–777, July 2001.

[KKT03]   David Kempe, Jon Kleinberg, and Éva Tardos. Maximizing the spread of influence through a social network. In *Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining*, KDD '03, pages 137–146, New York, NY, USA, 2003. ACM.

[KVV90]   Richard M. Karp, Umesh V. Vazirani, and Vijay V. Vazirani. An Optimal Algorithm for On-line Bipartite Matching. In Harriet Ortiz, editor, *STOC*, pages 352–358. ACM, 1990.

[LB11]   Hui Lin and Jeff Bilmes. Word alignment via submodular maximization over matroids. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies: short papers - Volume 2*, HLT '11, pages 170–175, Stroudsburg, PA, USA, 2011. Association for Computational Linguistics.

[Lov]   L. Lovasz. Submodular functions and convexity. *Mathematical Programming: The state of the art*, pages 235–257.

[NW78]   G. L. Nemhauser and L. A. Wolsey. Best Algorithms for Approximating the Maximum of a Submodular Set Function. *Mathematics of Operations Research*, 3(3):177–188, August 1978.

[NWF78]   G. L. Nemhauser, L. A. Wolsey, and M. L. Fisher. An analysis of approximations for maximizing submodular set functions—I. *Mathematical Programming*, 14(1):265–294, 1978.

[RFHT10]   Korbinian Riedhammer, Benoît Favre, and Dilek Hakkani-Tür. Long story short - Global unsupervised models for keyphrase based meeting summarization. *Speech Communication*, 52(10):801–815, 2010.

[Sch00]   Alexander Schrijver. A Combinatorial Algorithm Minimizing Submodular Functions in Strongly Polynomial Time. *J. Comb. Theory, Ser. B*, 80(2):346–355, 2000.

[Sch03]   A. Schrijver. *Combinatorial Optimization - polyhedra and efficiency*. Springer-Verlag Berlin Heidelberg, 2003.

[SL10]   Chao Shen and Tao Li. Multi-Document Summarization via the Minimum Dominating Set. In Chu-Ren Huang and Dan Jurafsky, editors, *COLING*, pages 984–992. Tsinghua University Press, 2010.

[TO09]   Hiroya Takamura and Manabu Okumura. Text Summarization Model Based on Maximum Coverage Problem and its Variant. In Alex Lascarides, Claire Gardent, and Joakim Nivre, editors, *EACL*, pages 781–789. The Association for Computer Linguistics, 2009.

[Von08]   Jan Vondrák. Optimal approximation for the submodular welfare problem in the value oracle model. In Cynthia Dwork, editor, *STOC*, pages 67–74. ACM, 2008.

[Zuc07]   David Zuckerman. Linear Degree Extractors and the Inapproximability of Max Clique and Chromatic Number. *Theory of Computing*, 3(1):103–128, 2007.