Nanyang Technological University

INCENTIVE COMPATIBLE DESIGN OF REVERSE AUCTIONS

NIKOLAI GRAVIN

Division of Mathematical Sciences

School of Physical and Mathematical Sciences

2013

Acknowledgments

I would like to thank my adviser Dima Pasechnik for his guidance and support throughout my time as a graduate student. I will forever be indebted to him for fruitful discussions and his introduction and encouragement for me to perform research not alone but together with other people and on diverse topics. Dima not only equipped me with a broad and very useful perspective on many fields in mathematics and computer science but also turned a simple graduate program into an exciting journey in the world of research.

I am extremely grateful to my fellow graduate students, lab mates and collaborators for providing me with wonderful research and live experience. An especially large fraction of my gratitude goes to my coauthors Ning Chen, Pinayn Lu and Edith Elkind. I wish to thank Sinai Robins and my other Ph.D. adviser Dmitrii Karpov for their guidance throughout my time in NTU and PDMI and most importantly for the belief in me.

It is my pleasure to say thanks to Pinyan Lu for mentoring me at Microsoft Research Asia, to Peter Bro Miltersen for hosting me at Aarhus University and to Jennifer Chayes for mentoring me at Microsoft Research New England during my summer overseas voyages.

This work was made possible by the financial support of SINGA scholarship at the Nanyang Technological University; I am grateful to them for their generosity.

Finally, my warmest thanks go to my family and friends, without whose indispensable support and encouragement I could not possibly have finished this work. Thank you all.

Contents

Abstract

We consider two classes of optimization problems that emerge in the set up of the reverse auctions (a.k.a. procurement auctions). Unlike the standard optimization taking place for a commonly known input, we assume that every individual submits his piece of the input and may misreport his data or not follow the protocol, in order to gain a better outcome. The study of scenarios falling into this framework has been well motivated by the advent of the Internet and, in particular, by rapidly growing industries such as sponsored-search ad-auctions, on-line auction services for consumer-to-consumer sales, marketing in social networks, etc. Our work contributes to the field of algorithmic mechanism design, which seeks to obtain nearly optimal algorithms and protocols that are robust against strategic manipulations of selfish participants.

The first part of this thesis is devoted to the problem of payment minimization under feasibility constraints overlaid on top of an underlying combinatorial structure of the outcome. We analyze the performance of incentive compatible procedures against two standard benchmarks and introduce a general scheme that proved to be optimal on some subclasses of vertex cover and all k-paths set systems. Our results completely settled the design of optimal frugal mechanism in path auctions, a decade-long standing open problem in algorithmic mechanism design proposed by Archer and Tardos.

In the second part of this thesis we study procurement auctions in which sellers have private costs to supply their items and the auctioneer aims to maximize the value of a purchased item bundle, while keeping payments under a budget constraint. For a few important classes of auctioneer's value functions defined over all item sets we give budget feasible incentive compatible mechanisms with desirable approximation guarantees and answer the "fundamental question" posed by Dobzinski, Papadimitriou, Singer [30].

Chapter 1

Introduction

Auctions are probably one of the oldest and simplest examples of an algorithmic procedure, where self-interested parties report their private data to the algorithm that decides upon the public outcome. According to Herodotus, human beings were using auctions from as early as 500 b.c. Objects of art, variety of goods like tobacco, tulips, fresh fish, and metals were and are being sold through an auction format. Various kinds of bonds, like bonds on public utilities and long-term securities issued by the U.S. Treasury, are auctioned to big financial institutions. The rights to many public property resources ranging from natural resources such as timber and oil to the rights on broadcast spectrum are distributed by means of an auction. Finally, the expansion of the Internet and growth of areas such as Electronic commerce and social networks paved the way for many more applications of auctions, and speaking in a broader sense, for applications of mechanism design. There are websites that organize auctions with individuals selling items to other users; we are witnessing tremendous growth of on-line advertising industry that proceeds mostly in an auction format and which already has got a noticeable share in the whole advertising area.

Auctions were not studied in the computer science literature until the last fifteen years, despite the algorithmic nature of most of the auctions procedures. Auctions have become a major subject of interest in a computer science discipline called algorithmic mechanism design, which was initiated by Nisan and Ronen in their seminal paper [59]. In this discipline, speaking in a broader sense, the basic implicit assumption in combinatorial optimization and its computational counterparts is that a problem's parameters or input to an algorithm are explicitly given and represent exactly the problem we want to solve. It turns out that in many situations, especially those related to the world wide web, this assumption is violated. Many algorithmic questions such as routing a message in a computer network, scheduling of tasks, memory allocation must take into consideration that in an environment there are multiple owners of resources or requests. The algorithm must work well if participants behave in a selfish and strategic way.

As a concrete example consider the problem of opening a few facilities in a city to serve the population of its residents. The city may be thought of as a metric space with every individual member of the population residing in a point of this metric space. Given the set of opened facilities, each resident commutes to the closest one. The central authority would want to install facilities in the way that minimizes the total sum of commute distances taken over the population. This is a classic optimization problem of uncapacitated metric facility location with the best currently known polynomial-time 1.489- approximation algorithm. However, the problem changes dramatically if every individual could report his location differently from his real placement in the city [61]. To illustrate the difference let us consider a simple example of installing a single facility in the real line, with only two residents. One can also imagine a different setup for this instance, where two friends Xaver and Yu decide on a point in time when to go together for lunch. Imagine that they naturally agreed to use the midpoint rule. Now if Yu wants to go at 1 p.m. and Xaver wants to go later, at 2 p.m., then using the midpoint rule they will have lunch at 1:30 p.m.. Yu knows that Xaver likes to have lunch late, so she can claim 12 a.m. to be her most preferred time and the midpoint rule will result in a lunch at 1 p.m. (the most preferred time for Yu). Thinking similarly, Xaver will claim 3 p.m., but Yu may expect such a cheating move from Xaver and would claim 11 a.m., and so on and so forth. If Xaver and Yu would have agreed on another selection rule of a random dictator (i.e., the most preferred time either of Xaver or of Yu gets chosen by tossing a coin), then none of them would have an incentive to cheat and will truthfully declare his/her private data. This agreement between them constitutes an example of so called truthful mechanism.

In this thesis we consider two natural optimization problems with part of the parameters provided by the participants, each participant with his own incentive. As the main goal of our research, we seek protocols that have good performance and are compatible with incentives of every individual. As an additional goal, we ask for efficient implementation of these protocols. Both optimization problems of interest arise in the set up of reverse auctions (a.k.a. procurement auctions) with underlying combinatorial structure.

1.1 Auctions, Incentives and Mechanisms

Below we cite a few important examples of auctions and discuss the behavior of the participants in the game propagated by their different incentives.

First-price sealed-bid auction. Auctioneer sells a single item to a group of bidders. Every bidder sends the auctioneer his bid in a sealed envelop. The highest bid wins (gets the object) at the price that the winner puts in his envelop. Every bidder has no incentive to indicate his true value v_i for the object, as his gains $v_i - b_i$ could be only 0. Similarly, no one will ever bid above his true value. For every fixed bidding behavior of the other participants, a bidder has to decide on the optimal bid to make. A decrease of the bid will increase his possible gain from winning, meanwhile will decrease the probability of winning the competition. Thus there is a game played by all bidders, where each player will make the optimal bid for any fixed strategies of the others. Following the well-established line of research we use the game-theoretic solution concept of the Nash equilibrium to capture a stable outcome of the game, in which every player with the knowledge about strategies of the other players can gain nothing by changing only his own strategy unilaterally.

English (second-price) auction. This type of auction is another practical and common form of single item auction. The auctioneer starts with a zero price and proceeds by increasing the price in small increments as long as there is more than one bidder interested in the item. The auction stops when only one bidder remains interested. This bidder wins and pays the price at which the penultimate bidder dropped from the competition. The strategic consideration of a bidder in second-price auctions are much simpler than that in the first-price auction. In fact, English auctions are known to be strategically equivalent to second-price sealed-bid auctions, which are conducted in the same format as first-price sealed-bid auctions with only difference that the winner pays not his bid but the amount equal to the second highest bid. In the second-price sealed bid auction it is weakly dominant strategy (i.e., there is no better strategy) for every participant to submit his true value as the bid.

The above is quite a plausible property of the second-price format, as it eliminates all incentive considerations of the participants. In a broader context of mechanism design this property is called incentive compatibility (a.k.a. truthfulness) and in many settings is one of the necessary requirements. Generally speaking, the design of mechanisms or auction formats that are incentive compatible may be viewed as an inverse problem to the problem of computing the Nash equilibria in a specific game.

Indeed, instead of fixing the rules of a game and figuring out what an equilibrium solution is, we are looking for the rules of a game which resolves rationality problems of strategic agents towards the trivial behavior.

Mechanism Design. This is a more general framework that includes auction design. Formally, there is a set of social alternatives A from which a mechanism should choose one specific outcome based on the reported data from the set I of n individuals. Each individual i has a valuation $v_i : A \to \mathbb{R}$, where $v_i(a)$ is the "value" of i for the social alternative $a \in A$; each individual i is also charged or compensated an amount of money p_i and derives the utility $u_i(a) = v_i(a) - p_i$. Depending on a particular scenario, the space of possible valuations may vary, so the valuation function v_i of each individual i comes from some underlying space V_i. Utility is an abstract quantity, but a very important one for the player i, in that it's a quantity he would always want to maximize. Thereby, any mechanism $\mathcal{M}$ must decide upon two things based on the private data (bids) v_i provided by selfish participants. First, $\mathcal{M}$ should choose the alternative $a \in A$. Second, $\mathcal{M}$ must decide on the vector of payoffs $\mathbf{p} = (p_1, \ldots, p_n)$ to charge or compensate each participant with. As in the example discussed above, the mechanism does not *a priori* know values of any individual but rather would want to collect this information from the players in a way that cannot be strategically manipulated. The way to achieve this is to make sure that every player would maximize his/her utility by providing the true information $v_i(\cdot)$ to the mechanism. Mechanisms satisfying this property are called truthful or incentive compatible.

Vickrey-Clarke-Groves (VCG) Mechanism. This is the most celebrated incentive compatible mechanism that generalizes the previous example of second-price auction and is applied in a very wide spectrum of different settings. A VCG mechanism always selects an alternative $a \in A$ that maximizes the total social welfare $\mathsf{SW}(a) = \sum_{i \in I} v_i(a)$. Sometimes, the set of individuals I includes a dummy player with an *a priori* given and fixed valuation $v_0(\cdot)$ over the outcomes that represents the interest of the mechanism and is always reported truthfully ($v_0(\cdot) = 0$, if mechanism has no interest in the outcome). Thus, the allocation rule $\mathcal{A}$ is given by

$$\mathcal{A}(v_0, \ldots, v_n) \in \operatorname*{argmax}_{a \in A} \; \mathsf{SW}(a).$$

Let $\mathcal{O} = \mathcal{A}(v_0, \ldots, v_n)$ be the social alternative chosen by $\mathcal{M}$. The payoff to every player in VCG represents the externality that player i exerts on the other individuals by his presence in the society. There is more than one incentive compatible payment rules supporting the allocation rule $\mathcal{A}$. We denote by $\mathbf{v}_{-i}$ the vector of n functions $(v_0, \ldots, v_{i-1}, v_{i+1}, \ldots, v_n)$ and similarly by V_{-i} the space of all possible valuations excluding the i-th one. The complete list of payment rules is given by a set of functions $h_1, \ldots, h_n$, where $h_i : V_{-i} \to \mathbb{R}$ and the payment p_i to each agent $i \in I \setminus \{0\}$ is

$$p_i(v_0, \ldots, v_n) = h_i(\mathbf{v}_{-i}) - \sum_{k \in I \setminus \{i\}} v_k(\mathcal{O}).$$

Procurement Auctions. These are the "reverse" auctions, in which the auctioneer buys items from many suppliers who compete for the right to sell their goods, resources or services. Most real-life government purchases are being done in this way and the practice of procurement is quite common in business.

(Example 1.) One can also find a nice unexpected application of procurement auctions to influence maximization in Social Networks presented in [64]. In social network marketing the goal is to monetarily entice a small set of individuals in a social network to recommend a product, in a way that maximizes the word-of-mouth effect in the network. The market designer is naturally limited by the amount of rewards he can offer, and each individual has a different cost for making a recommendation to his/her friends in the network. Since individuals may lie about their costs, the market designer strives to design an incentive compatible mechanism that will maximize the word-of-mouth effect in the network.

(Example 2.) As another example (see p. 351 in [60]) let us consider a scenario where a company wants to purchase transportation services for a large number of "routes" from various providers (e.g., trucking or shipping companies). In the mechanism design formulation of this example (single-parameter version) each of n players may be represented by an edge in an underlying graph G.

The auctioneer purchases a service from a group of agents, so that the set of alternatives A consists of all subsets of edges (i.e., $A = 2^{E(G)}$). If a selected set S of edges contains a path from the source vertex s to the destination vertex t in G, then the auctioneer's valuation is defined to be $v_0(S) = 0$; otherwise we set $v_0(S) = -\infty$. Every agent i, if selected in set S, incurs losses c_i (i.e., $v_i(S) = -c_i$), and otherwise $v_i(S) = 0$. Thereby, every agent reports only one parameter c_i to the auctioneer, so that $V_i = \mathbb{R}$. The total social welfare in this scenario is $\mathsf{SW}(S) = v_0(S) - \sum_{i \in S} c_i$. In this setting the auctioneer pays agents, so $p_i \leq 0$. In order to avoid social welfare of $-\infty$, VCG must always select a set of edges that contains a path from s to t for any reported cost vector. In this setting there is the canonical VCG payment rule, which ensures individual rationality (i.e., $u_i \geq 0$) for every agent i. It this rule $h_i(\mathbf{v}_{-i})$ is equal to $\max\limits_{S \subset E(G)-i} \mathsf{SW}(S)$.

Now let us see how the VCG mechanism works on the graph G composed of a path P of length $n-1$ and a single edge e_{st} between s and t. Suppose that every edge on the path P reports 0 and edge e_{st} reports 1. The VCG must select the cheapest path P as the outcome $\mathcal{O}$. It turns out that $h_i(\mathbf{v}_{-i}) = -1$ and $p_i = -1$ for every $i \in P$; $h_i(\mathbf{v}_{-i}) = 0$ and $p_i = 0$ for $i = e_{st}$. Thereby, the auctioneer pays 1 to every edge in P with the total payment of $n-1$.

1.2 Models Considered

Frugality. In Example 2 of procurement auctions the natural optimization objective would be to minimize payments while purchasing a feasible path. Generally speaking, we study the design of truthful mechanisms for set systems, i.e., settings where a customer needs to hire a team of agents to perform a complex task. Given a set of agents $\mathcal{E}$, a subset $S \subseteq \mathcal{E}$ is said to be *feasible* if the agents in S can jointly perform the complex task. This setting can be described by a *set system* $(\mathcal{E}, \mathcal{F})$, where $\mathcal{E}$ is the set of agents and $\mathcal{F}$ is the collection of feasible sets. Each agent $e \in \mathcal{E}$ can perform a simple task at a privately known cost $c(e)$.

Each agent e submits a *bid* $b(e)$, the payment that he wants to receive. Based on these bids the customer selects a feasible set $S \in \mathcal{F}$ (the set of *winners*), and determines the payment to each winner. In this setting, frugality [3] provides a measure to evaluate the "cost of truthfulness", that is, the overpayment of a truthful mechanism relative to the "fair" payment, where the "fair" payment is defined as a Nash equilibrium in the first-price auction. Since Nash equilibrium might not be unique, two slightly different benchmarks depending upon the choice of the specific Nash equilibrium for all possible true costs were considered in the literature.

We study two specific set systems. In the first one, $\mathcal{E}$ is the set of edges in a given network and $\mathcal{F}$ consists of all k-edge disjoint paths from source s to destination t; in the second set system, $\mathcal{E}$ is the vertex set of a given graph G and $\mathcal{F}$ consists of all vertex covers[1] of G.

Budget Feasibility. The former Example 1 of procurement auction in the social network represents another type of optimization problems. Here, instead of payment minimization under feasibility constraint, the auctioneer would want to maximize the value with a constraint on the total payment. In more detail, in budget feasible mechanism design, one studies procurement combinatorial auctions in which the sellers have private costs to produce items, and the buyer (auctioneer) aims to maximize his valuation function defined over all possible bundles of items, under the budget constraint on the total payment. In this thesis we will be mostly dealing with the single parameter case, in which every agent provides only one item for sale. However, in Chapter 5 we will consider an extension to multi-parameter case in which a seller may provide many items.

Our main goal will be to find incentive compatible and budget feasible mechanisms that have good approximation compared to the optimal solution (in the full information case where all private costs are known). In Chapter 4 we will

[1] In graph theory a set of vertices is called a *vertex cover* if every edge of a graph contains (is covered by) at least one vertex from this set

solely take prior-free worst-case viewpoint, i.e., we require our mechanism to perform well w.r.t. the optimal solution for any possible vector of bids. In Chapter 5 we also will consider our problem in a softened Bayesian framework, which is the standard approach from economics and now is becoming popular in the algorithmic game theory community. In Bayesian framework the performance of a mechanism is measured in expectation for a given prior distribution over the profiles of costs.

1.3 Related Literature

The topics discussed in this thesis fall under the rubric of "algorithmic mechanism design", which is a fascinating area initiated by the seminal work of Nisan and Ronen [59]. For a survey on the area we recommend [60], where many mechanism design models are discussed.

1.3.1 Frugal Mechanism Design

There is a substantial literature on designing mechanisms with small payment for shortest path systems [3, 33, 36, 24, 31, 47, 69] as well as for other set systems [67, 14, 49, 32], starting with the seminal work of Nisan and Ronen [59]. Our work is most closely related to [49], [32] and [69]. we employ the frugality benchmark ν defined in [49] and analyze our mechanism w.r.t. one more frugality benchmark μ defined in [32]; we improve the bounds of [49] on the ν-frugality ratio for path auctions and show optimality of our mechanism w.r.t. the μ-benchmark; we refine and improve the bounds on frugality ratio of [32] for vertex cover set systems, namely, our bounds on the frugality ratio depend on a particular instance of a graph in the vertex cover problem rather than on a worst-case instance in the family of graphs with a fixed maximal degree [32]; we generalize the ν-frugality result of [69] for single path auctions by extending it to k-paths auctions.

Simultaneously and independently, the idea of bounding frugality ratios of set system auctions in terms of eigenvalues of certain matrices was considered by Kempe, Salek and Moore [50]. In contrast with our work, in [50] the authors only study the frugality ratio of their mechanisms with respect to the relaxed payment bound of [32]. They give a 2-competitive mechanism for vertex cover systems, $2(k+1)$-competitive mechanism for k-path systems, and a 4-competitive mechanism for a new class of cut set systems introduced therein.

1.3.2 Budget Feasible Mechanism Design

The study of approximate mechanism design with a budget constraint was originated by Singer [63], where he proposed constant approximation mechanisms for additive and submodular functions. Later we [21] constructed mechanisms with better approximation ratios for additive and submodular valuations. Dobzinski, Papadimitriou, and Singer [30] considered subadditive functions and presented $O(\log^2 n)$ approximation mechanism. Ghosh and Roth [37] use a budget feasible mechanism design model for selling privacy where there are externalities for each agent's cost. In [7] Badanidiyuru et al. consider a budget feasible model, in which agents arrive in on-line fashion, and study posted price mechanisms. All these models considered prior-free worst case analysis.

For Bayesian mechanism design, Hartline and Lucier [45] first proposed a Bayesian reduction in single-parameter settings that converts any approximation algorithm to a Bayesian truthful mechanism that approximately preserves social welfare. The black-box reduction results were later improved to multi-parameter settings in [10] and [44], independently. Chawla et al. [16] considered budget-constrained agents and gave Bayesian truthful mechanisms in various settings. A number of other Bayesian mechanism design works considered profit maximization,e.g., [46, 11, 17, 27, 15, 26]. In the current thesis we consider Bayesian analysis in budget feasible mechanisms with a focus on valuation (social welfare) maximization.

1.4 Contributions and Road Map

Part 1: Frugal Mechanism Design. In Chapter 2 we introduce the setup and provide necessary background. We further focus on possible benchmarks against which we measure frugality and in particular investigate questions in regard to the first-price Nash equilibrium in k-paths set systems. In particular we provide structural characterization of all first-price k-paths auctions in 2.3. This result published in [19] extends the previously known characterization of first-price single path auctions.

In Chapter 3 we propose a uniform scheme for designing frugal truthful mechanisms for general set systems. Our scheme is based on scaling the agents' bids using the eigenvector of a matrix that encodes the inter-dependencies between the agents. We demonstrate that the r-out-of-k-system mechanism and the $\sqrt{\ }$-mechanism [49] for buying a path in a graph can be viewed as instantiations of our scheme. We then apply our scheme to two other classes of set systems, namely, vertex cover systems and k-path systems, in which a customer needs to purchase k edge-disjoint source-sink paths. For both settings, we bound the frugality of our mechanism in terms of the largest eigenvalue of the respective interdependency matrix.

We show that our mechanism is optimal for a large subclass of vertex cover systems satisfying a simple local sparsity condition, which holds, e.g., for all triangle free graphs. For k-path systems, our mechanism is within a factor of $k+1$ from optimal if measured against ν-benchmark proposed in [49]; moreover, we show that this scheme is, in fact, *optimal* for k-paths, when one uses μ-benchmarks proposed in [32]. Our lower bound argument combines spectral techniques and Young's inequality, and is applicable to all set systems. As both r-out-of-k systems and single path systems can be viewed as special cases of k-path systems, our result improves the lower bounds of [49].

Our analysis employs tools from spectral graph theory which to the best of our knowledge have never been used in algorithmic game theory prior to our work [18].

Our main technical contribution in [18] consists of: a lower bound on any truthful mechanism's payment that combines Young's inequality and spectral techniques; upper bounds on our mechanism's payment and appropriate lower bounds on μ- and ν-benchmarks. The latter contribution heavily relies on the characterization of first-price equilibria from [19] and uses several subtle combinatorial lemmas about min-cost max-flow.

It should be noted that simultaneously and independently from our work [18] exactly the same eigenvector scaling scheme was proposed in [50]. The latter work is focused solely on the μ-benchmark analysis and for k-paths systems they showed that the eigenvalue scheme is within factor $2(k+1)$ from optimal truthful mechanism, whereas our work shows that the same scheme is in fact optimal. We note that [50] does not use Young's inequality (extra factor of 2 compared to [18]) and applies simpler lower bound on μ-benchmark (extra factor of $k+1$ compared to [18]). On the other hand, they obtain 2-competitive mechanism for all vertex cover instances and consider a new class of cut set systems for which they provide 4-competitive mechanism. In fact, one can show that their mechanism for vertex cover is optimal by applying our Young's inequality argument. Their mechanism for vertex cover set systems are not computationally efficient, while all mechanisms considered in our work admit efficient polynomial time implementations.

Part 2: Budget Feasible Mechanism Design. In Chapter 4 we provide an extended introduction and preliminaries for the original budget feasible model as it appeared in [63]. We then present few incentive compatible budget feasible schemes with a good approximation to the optimal solutions for various classes of auctioneer's valuations from the following classical hierarchy [54] of complement free functions:

$$\begin{aligned} \text{additive} \;\subset\; \text{gross substitutes} \;&\subset\; \text{submodular} \\ &\subset\; \text{XOS} \;\subset\; \text{subadditive.} \end{aligned}$$

We begin with the basic case of additive valuation and give a $(2+\sqrt{2})$-approximation deterministic mechanism (improving on the previous best-known result of 5), and a 3-approximation randomized mechanism. We complement the case of additive valuations with a lower bound of $1+\sqrt{2}$ on the approximation ratio of any deterministic and lower bound of 2 for any randomized truthful budget feasible mechanisms (improving on previous lower bound of 2 for deterministic mechanisms). These lower bounds are unconditional, and do not rely on any computational or complexity assumptions. Apart from new lower bound examples, our mechanisms for additive valuations are based on similar ideas as those considered in [63], however with significantly more accurate analysis.

We proceed then to monotone submodular valuations and present randomized mechanism with an approximation ratio of 7.91 (improving on the previous best-known result of 117.7), and a deterministic mechanism with an approximation ratio of 8.34. The natural greedy algorithm is a good candidate for designing budget feasible mechanisms due to its nice monotonicity property and small approximation ratio. Both ours and Singer's work are based on this greedy strategy, but have different ideas in the analysis. Computing the threshold payment to each winner might be a tricky task because each agent can manipulate her ranking position in the greedy algorithm, which results in different computations of the marginal contributions for the remaining agents, therefore, leading to unpredictable change in the set of winners. Singer's [63] approach is based on the complete payment's characterization of every winner, while our approach [21] uses simple upper bounds on the payments by exploiting combinatorial structure of submodular functions. In Chapter 4, we give a clean analysis for the upper bound on threshold payment by applying the combinatorial structure of submodular functions (Lemma 9). These upper bounds on payments suggest appropriate parameters in our randomized mechanism, which, roughly speaking, selects the greedy algorithm or the agent with the largest value at a certain probability.

Finally, we introduce a constant approximation randomized mechanism for

XOS (a.k.a. fractionally subadditive) valuations. No constant approximation mechanism for XOS valuations was known prior to our work [9]. In the last section we present two randomized mechanisms for arbitrary subadditive valuations. The first mechanism proceeds via a straightforward reduction to XOS valuation; its approximation ratio is $O(\mathcal{I})$, via the worst-case integrality gap $\mathcal{I}$ of the LP that describes the fractional cover of the valuation function. The second mechanism works in polynomial time and provides an $O(\frac{\log n}{\log \log n})$ sub-logarithmic approximation for arbitrary subadditive function. As for subadditive valuations $\mathcal{I} = O(\log n)$, both of our mechanisms improve upon the best previously known approximation ratio of $O(\log^2 n)$ known for subadditive functions. Our methods for XOS valuations relies on the idea of sampling randomly part of the agents in a group which we only use to learn a rough estimate on the optimum. The idea of random sampling is a classic tool in on-line algorithm literature and also has been used long before our work in algorithmic game theory community in other settings, e.g., in digital good auctions [39]. However, our work [9] is the first that uses random sampling approach in the context of budget feasible model. We use the properties of XOS valuations to find a "good" set on which we can approximate our valuation by an additive function without much loss in the value. This allows us to reduce the problem to the additive case. Overall, our design philosophy is different from [30], which is based on the search of a suitable vector of posted prices.

Throughout Chapter 4 we explore a question posed in [30] by Dobzinski, Papadimitriou, and Singer:

> *"A fundamental question is whether, regardless of computational constraints, a constant-factor budget feasible mechanism exists for subadditive functions."*

As we show in the last section of Chapter 5 this very question has a positive answer. Our argument is non-constructive, which is unusual for algorithmic mechanism design literature. Namely, in Chapter 5 we address the above question from a different viewpoint and analyze performance of an incentive compatible mechanism in the Bayesian framework, which is a standard approach from economics, and which is getting popular in the algorithmic game theory community. In the Bayesian framework, we provide a constant approximation mechanism for arbitrary subadditive valuations, using the $O(1)$-approximation prior-free mechanism for XOS valuations as a subroutine. Unlike most of the previous work done in the Bayesian framework, we allow for a non-trivial correlation in the distribution of the private costs. Then we show existence of a constant approximation mechanism in the worst-case prior-free framework by translating our results in the Bayesian framework with the usage of Yao's min-max principle.

In Chapter 5 we propose a multi-parameter extension of the budget feasible model, in which each seller may offer more than one item for sale. For this extension we give a constant approximation mechanism for the class of submodular valuations based on random sampling approach.

Part I

Hiring a Team of Agents

Chapter 2

Frugal Mechanism Design 1: Nash Equilibria

2.1 Introduction

Consider a scenario where a customer wishes to purchase the rights to have data routed on his behalf from a source s to a destination t in a network where each edge is owned by a selfishly motivated agent. Each agent incurs a privately known cost if the data is routed through his edge, and wants to be compensated for this cost, and, if possible, to make a profit. The customer needs to decide which edges to buy, and wants to minimize his total expense.

This problem is a special case of the *hiring-a-team* problem ([67, 49, 48, 22, 32]): Given a set of agents $\mathcal{E}$, a customer wishes to hire a team of agents capable of performing a certain complex task on his behalf. A subset $S \subseteq \mathcal{E}$ is said to be *feasible* if the agents in S can jointly perform the complex task. This scenario can be described by a *set system* $(\mathcal{E}, \mathcal{F})$, where $\mathcal{E}$ is the set of agents and $\mathcal{F}$ is the collection of feasible sets of agents. Each agent $e \in \mathcal{E}$ can perform a simple task at a privately known cost $c(e)$. In such environments, a natural way to make the hiring decisions is by means of *mechanisms* — each agent e submits a *bid* $b(e)$, i.e., the payment that he wants to receive, and based on these bids the customer selects a feasible set $S \in \mathcal{F}$ (the set of *winners*), and determines the payment to each winner.

A desirable property of mechanisms is that of *incentive compatibility* (a.k.a. truthfulness): it should be in the best interest of every agent e to bid his true cost, i.e., to set $b(e) = c(e)$, no matter what bids other agents submit; that is, truth-telling should be a dominant strategy for every agent. Truthfulness is a strong and very appealing concept: it obviates the need for agents to perform complex strategic computations, even if they do not know the costs and strategies of others.

One of the most celebrated truthful designs discussed in Section 1.1 is the VCG mechanism [68, 23, 42]. In VCG mechanism in the context of reverse auction the feasible set with the smallest total bid wins, and the payment to each agent e in the winning set is his threshold bid, i.e., the highest value that e could have bid to still be part of a winning set. The VCG mechanism is truthful. However, on the negative side, it can make the customer pay far more than the true cost of the winning set, or even the cheapest alternative, as illustrated by the following example: there are two parallel paths P_1 and P_2 from s to t, P_1 has one edge with cost 1 and P_2 has n edges with cost 0 each. VCG selects P_2 as the winning path and pays 1 to every edge in P_2. Hence, the total payment of VCG is n, the number of edges in P_2, which is far more than the total cost of both P_1 and P_2.

The VCG overpayment property illustrated above is clearly undesirable from the customer's perspective, and thus motivates the search for truthful mechanisms that are *frugal*, i.e., select a feasible set and induce truthful cost revelation without resulting in high overpayment. However, formalizing the notion of frugality is a challenging problem, as it is not immediately clear to what the payment of a mechanism should be compared.

The first candidate for a benchmark would be the actual cost of the cheapest feasible set. However, such a benchmark is not suitable for us as if the costs of other agents go arbitrarily high, then a mechanism's payment must be unbounded while benchmark's cost remains the same.

Another natural candidate for this benchmark is the total cost of the closest competitor, i.e., the cost of the cheapest feasible set among those that are disjoint from the winning set. This definition coincides with the second highest bid in single-item auctions and has been used in, e.g., [2, 3, 67, 33]. However, as observed by Karlin, Kempe and Tamir [49], such a feasible set may not exist at all, even in monopoly-free set systems (i.e., set systems where no agent appears in all feasible sets). To deal with this problem, [49] proposed an alternative benchmark, which is bounded for any monopoly-free set system and is closely related to Nash equilibria of the first-price auction.

The auctioneer in the first-price auction buys the cheapest feasible set at the prices that are equal to the bids. Although such an auction is simple and naturally happens in market environments, first-price auction is not incentive compatible. In fact, it defines a game among agents, where the strategy of each agent is her bid.

In this chapter we focus on studying the first-price benchmark and first address the questions related to Nash equilibria of the first-price auction for a specific case of hiring k-paths in a network. To illustrate an instance of k-paths set system, one may consider all vertices in a graph G given by their geographical locations and edges that correspond to the routes between them. A shipping company plans to carry k items from source s to destination t. Due to capacity constraint, every edge can carry at most one item. Further, for each edge, there is an associated cost $c(e)$ (e.g. maintenance) incurred to local carrier to provide his service. Therefore, the company has to make a payment to each edge it uses to recover those costs. By a standard game theoretical assumption, all edges are selfish and hope to receive as much payment as possible (given that their costs are recovered).

2.2 Preliminaries

A *set system* $(\mathcal{E}, \mathcal{F})$ is given by a set $\mathcal{E}$ of *agents* and a collection $\mathcal{F} \subseteq 2^{\mathcal{E}}$ of *feasible sets*. We restrict our attention to *monopoly-free* set systems, i.e., we require $\bigcap_{S \in \mathcal{F}} S = \emptyset$. Each agent $e \in \mathcal{E}$ has a privately known *cost* $c(e)$ that represents the expenses that agent e incurs if he is involved in performing the task. In particular, in a k-paths set system the agents are edges in a given network and feasible sets are all edge-disjoint k paths from a vertex s to a vertex t in this network.

A *mechanism* for a set system $(\mathcal{E}, \mathcal{F})$ takes a *bid vector* $\mathbf{b} = (b(e))_{e \in \mathcal{E}}$ as input and outputs a set of *winners* $S \in \mathcal{F}$ and a *payment* $p(e)$ for each $e \in \mathcal{E}$. We require mechanisms to satisfy *voluntary participation*, i.e., $p(e) \geq b(e)$ for each $e \in S$ and $p(e) = 0$ for each $e \notin S$. Given the output of a mechanism, the *utility* of an agent e is $p(e) - c(e)$ if e is a winner and 0 otherwise. We assume that agents are rational, i.e., aim to maximize their own utility. Thus, they may lie about their true costs, i.e., bid $b(e) \neq c(e)$ if they can profit by doing so. We say that a mechanism is *truthful* if every agent maximizes his utility by bidding his true cost, no matter what bids other agents submit. A weaker solution concept is that of *Nash equilibrium*: a bid vector constitutes a (pure) Nash equilibrium if no agent can increase his utility by unilaterally changing his bid.

There is a well-known characterization of winner selection rules that yield truthful mechanisms.

Theorem 1 ([52, 3]). *A mechanism is truthful if and only if its winner selection rule is* monotone*, i.e., no losing agent can become a winner by increasing his bid, given the fixed bids of all other agents. Further, for a given monotone selection rule, there is a unique truthful mechanism with this selection rule: the payment to each winner is his* threshold bid*, i.e., the supremum of the values he could bid and still win.*

In what follows we consider a specific k-paths set system, where auctioneer is given a directed graph $G = (V, E)$ with a bid $b(e)$ on each edge $e \in E$ and two specific vertices $s, t \in V$. In a market setting, each edge sets up a price $b(e) \geq c(e)$ asking for its service. Given all $(b(e))_{e \in E}$, the auctioneer applies the first-price auction and purchases k edge-disjoint paths of the smallest total cost, i.e. shortest k edge-disjoint paths between s and t with respect to $c(e)$.

The condition of a Nash equilibrium says that no agent e can change her bid $b(e)$ in order to increase her utility $u(e)$; where the utility of an agent is assumed to be quasi-linear, i.e., $u(e) = b(e) - c(e)$, if auctioneer purchases e, and 0 otherwise. In the case of k-paths set systems, the Nash equilibrium condition is simply equivalent to the following: the length of a shortest with respect to $b(\cdot)$ disjoint k-paths from s to t in G remains unchanged even after deletion of arbitrary single edge $e \in E$. In particular, for $k = 1$ after deleting any edge in a shortest path from s to t, there is still an s-t path of the same length. In this case it can be easily shown by Menger's theorem [55] applied to the subgraph consisting of all shortest s-t paths that then G must necessarily have two edge-disjoint shortest paths from s to t. In the following section we extend this result to shortest k edge-disjoint paths.

2.3 Equilibria of First-Price Auctions

The next theorem provides a simple characterization of all possible Nash equilibria of the first-price auction applied to k-paths set systems. Its proof involves a careful examination of a specific real-valued min-cost max-flow defined using the average of $|E|$ different integer-valued min-cost max-flows and showing that any s-t path with positive amount of flow on each edge forms a shortest path.

Theorem 2. *Let $G = (V, E)$ be a directed graph with a cost $b(e)$ on each edge $e \in E$. Given two specific vertices $s, t \in V$, assume that there are k edge-disjoint paths from s to t. Let $P_1, P_2, \cdots, P_k$ be k edge-disjoint s-t paths so that their length $L \triangleq \sum_{i=1}^{k} w(P_i)$ is minimized, where $w(P_i) = \sum_{e \in P_i} b(e)$. Further, suppose* *that for every edge $e \in E$, the graph $G - \{e\}$ has k edge-disjoint s-t paths with the same total length L. Then there exist $k+1$ edge-disjoint s-t paths in G such that each of them is a shortest path from s to t.*

Remark 1. *Note that the theorem implies, in particular, that the original k edge-disjoint s-t paths $P_1, P_2, \cdots, P_k$ are shortest paths.*

Proof. Given the graph G and integer k, we construct a flow network $\mathcal{N}_k(G)$ as follows: we introduce two extra nodes s_0 and t_0 and two extra edges $s_0 s$ and tt_0. The set of vertices of $\mathcal{N}_k(G)$ is $V \cup \{s_0, t_0\}$ and the set of edges is $E \cup \{s_0 s, tt_0\}$. The capacity $cap(\cdot)$ and cost per unit capacity $cost(\cdot)$ for each edge in $\mathcal{N}_k(G)$ are defined as follows:

- $cap(s_0 s) = cap(tt_0) = k$ and $cost(s_0 s) = cost(tt_0) = 0$.
- $cap(e) = 1$ and $cost(e) = b(e)$, for $e \in E$.

Given the above construction, every path from s to t in G naturally corresponds to a unit flow from s_0 to t_0 in $\mathcal{N}_k(G)$. Hence, the set of k edge-disjoint paths $P_1, P_2, \ldots, P_k$ in G corresponds to a flow F_G of size k in $\mathcal{N}_k(G)$. In addition, the minimality of $L = \sum_{i=1}^{k} w(P_i)$ implies that F_G achieves the minimum cost (which is L) for all *integer*-valued flows of size k, i.e., maximum flow in $\mathcal{N}_k(G)$. Since all capacities of $\mathcal{N}_k(G)$ are integers, we can conclude that F_G has the minimum cost among all *real* maximum flows in $\mathcal{N}_k(G)$, the details one can find in [41].

For simplicity, we denote the subgraph $G - \{e\}$ by $G - e$. By the fact that for any $e \in E$, the subgraph $G - e$ has k edge-disjoint s-t paths with the same total length L, we know that in the network $\mathcal{N}_k(G - e)$, there still is an integer-valued flow F_{G-e} of size k and cost L. So F_{G-e} is also an integer-valued flow of size k and cost L in $\mathcal{N}_k(G)$. Define a real-valued flow in $\mathcal{N}_k(G)$ by $F = \frac{1}{|E|} \sum_{e \in E} F_{G-e}$. We observe the following.

1. It is clear that $F(e) \leq cap(e)$ for every arc $e \in \mathcal{N}_k(G)$, where $F(e)$ is the amount of flow on edge e in F, as we have taken the average of the flows in the network.

2. F has cost $\frac{1}{|E|} \sum_{e \in E} cost(F_{G-e}) = \frac{1}{|E|} \cdot |E| \cdot L = L$.

3. Since $F_{G-e}(s_0 s) = k$ for any $e \in E$, we have $F(s_0 s) = k$. In addition, as each F_{G-e} is a feasible flow that satisfies all conservation conditions and F is defined by the average of all F_{G-e}'s, we know that F also satisfies all conservation conditions.

Therefore, F is a minimum cost maximum flow in $\mathcal{N}_k(G)$. In addition, F has the following nice property, which plays a fundamental role for the proof.

- For every edge $e \in \mathcal{N}_k(G)$ except $s_0 s$ and tt_0, we have $F(e) \leq cap(e) - \frac{1}{|E|}$, as F_{G-e} does not flow through e, i.e. $F_{G-e}(e) = 0$, and $F_{G-e'}(e)$ is either 0 or 1 for any $e' \in E$.

Let $E_+ = \{e \in \mathcal{N}_k(G) \mid F(e) > 0\}$. Suppose that there is a path $P' = (e_1, e_2, \ldots, e_r)$ from s_0 to t_0 which goes only along arcs in E_+ and is not a shortest path w.r.t. $cost(\cdot)$ from s_0 to t_0 in $\mathcal{N}_k(G)$. Let $\varepsilon = \min\left\{F(e_1), F(e_2), \ldots, F(e_r), \frac{1}{|E|}\right\}$. Since $P' \subseteq E_+$, we have $\varepsilon > 0$. Let P be a shortest path w.r.t. $cost(\cdot)$ from s_0 to t_0 in $\mathcal{N}_k(G)$. Define a new flow F' from F by adding ε amount of flow onto path P and removing ε amount of flow from path P'. We observe the following about F'.

1. The value of flow F' is k.

2. F' satisfies all conservation conditions as it is a linear combination of three flows from s_0 to t_0.

3. By the definition of ε, the amount of flow of each edge is non-negative in F'. Further, F' satisfies the capacity constraints. This follows from the facts that $\varepsilon \leq \frac{1}{|E|}$ and the above property established for F.

4. The cost of F' is smaller than L because $cost(F') = cost(F) - \varepsilon(cost(P') - cost(P))$, which is smaller than $L = cost(F)$ as $cost(P) < cost(P')$ by the assumption.

Hence, F' is a flow of size k in $\mathcal{N}_k(G)$ with cost smaller than F, a contradiction. Thus, every path from s_0 to t_0 in $\mathcal{N}_k(G)$ along the edges of E_+ is a shortest path w.r.t. $cost(\cdot)$.

We define a new network $\mathcal{N}'(G)$ obtained from $\mathcal{N}_{k+1}(G)$ by removing all other edges except for those in E_+. We claim that in $\mathcal{N}'(G)$ there is an integer-valued flow of size $k+1$. Indeed, otherwise, by max-flow min-cut theorem, there is a cut (S_{s_0}, T_{t_0}) in $\mathcal{N}'(G)$ with a size less than or equal to k. By definition, in $\mathcal{N}'(G)$ we have $cap(s_0s) = k+1$ and $cap(tt_0) = k+1$, which implies that $s_0, s \in S_{s_0}$ and $t_0, t \in T_{t_0}$. By the definition of F, we know that total amount of F passing through the edges of the cut (S_{s_0}, T_{t_0}) is k. Since $F(e) < 1$ for any edge e of G, we can conclude that there are at least $k+1$ edges from S_{s_0} to T_{t_0} in E_+. This leads to a contradiction, because we have shown that the size of the cut (S_{s_0}, T_{t_0}) is less than or equal to k.

Therefore, we can find an integer-valued flow of size $k+1$ on the edges of E_+ in the network $\mathcal{N}(G)$. Such a flow can be thought of as a union of $k+1$ edge-disjoint paths from s_0 to t_0. We know that every such path going along edges in E_+ is a shortest path from s_0 to t_0. This in turn concludes the proof, since we have found $k+1$ edge-disjoint shortest paths from s to t in G. $\square$

2.4 Possible Benchmarks to measure Frugality

A classic example of a truthful set system mechanism is given by the VCG mechanism [68, 23, 42]. However, as discussed in the Introduction of Chapter 2, VCG often results in a large overpayment to winners. Another natural mechanism for buying a set is the *first-price auction*: given the bid vector $\mathbf{b}$, pick a subset $S \in \mathcal{F}$ minimizing $b(S)$, and pay each winner $e \in S$ his bid $b(e)$. While the first-price auction is not truthful, and more generally, does not possess dominant strategies, it essentially admits a Nash equilibrium with a relatively small total payment. (More accurately, as observed by [47], a first-price auction may not have a pure strategy Nash equilibrium. However, this non-existence result can be circumvented in several ways, e.g., by considering instead an ε-Nash equilibrium for arbitrarily small $\varepsilon > 0$ or using oracle access to the true costs of agents to break ties.) The payment in a buyer-optimal Nash equilibrium would constitute a natural benchmark for truthful mechanisms. However, due to the difficulties described above, we use instead the following benchmark proposed by Karlin et al. [49], which captures the main properties of a Nash equilibrium.

2.4.1 ν-benchmark

Definition 1 (Benchmark $\nu(\mathbf{c})$ [49]). *Given a set system $(\mathcal{E}, \mathcal{F})$, and a feasible set $S \in \mathcal{F}$ of minimum total cost w.r.t. $\mathbf{c}$, let $\nu(\mathbf{c})$ be the value of an optimal solution to the following optimization problem:*

$$
\begin{aligned}
\min \quad & \sum_{e \in S} b(e) \\
\text{s.t.} \quad & (1)\ b(e) \geq c(e) \text{ for all } e \in \mathcal{E} \\
& (2)\ \sum_{e \in S \setminus T} b(e) \leq \sum_{e \in T \setminus S} c(e) \text{ for all } \ T \in \mathcal{F} \\
& (3)\ \text{For every } e \in S, \text{ there is a } T \in \mathcal{F} \text{ s.t. } e \notin T \\
& \qquad \text{and} \sum_{e' \in S \setminus T} b(e') = \sum_{e' \in T \setminus S} c(e')
\end{aligned}
$$

Intuitively, in the optimal solution of the above system, S is the set of winners in the first-price auction. By condition (3), no winner $e \in S$ can improve his utility by increasing his bid $b(e)$, as he would not be a winner anymore. In addition, by conditions (1) and (2), no agent $e \in \mathcal{E} \setminus S$ can obtain a positive utility by decreasing his bid. Hence, $\nu(\mathbf{c})$ gives the value of the cheapest Nash equilibrium of the first-price auction assuming that the most "efficient" feasible set S wins.

Definition 2 (Frugality Ratio). *Let $\mathcal{M}$ be a truthful mechanism for the set system $(\mathcal{E}, \mathcal{F})$ and let $p_{\mathcal{M}}(\mathbf{c})$ denote the total payment of $\mathcal{M}$ with the true costs given by a vector $\mathbf{c}$. Then the frugality ratio of $\mathcal{M}$ on $\mathbf{c}$ is defined as $\phi_{\mathcal{M}}(\mathbf{c}) = \frac{p_{\mathcal{M}}(\mathbf{c})}{\nu(\mathbf{c})}$. Further, the* frugality ratio *of $\mathcal{M}$ is defined as $\phi_{\mathcal{M}} = \sup_{\mathbf{c}} \phi_{\mathcal{M}}(\mathbf{c})$.*

2.4.2 μ-benchmark

It turns out that ν-benchmark has a few undesirable properties, as was mentioned in [32, 22, 50]. In particular, $\nu(\mathbf{c})$ may increase, if one introduces a few more feasible sets in our set system and, therefore, increases a competition between the agents. Moreover, $\nu(\mathbf{c})$ is NP-hard to find even in 1-path set system [22].

A weaker benchmark $\mu(\mathbf{c})$, namely, one that corresponds to a buyer-pessimal rather than buyer-optimal Nash equilibrium, was introduced in [32], and has been used by Kempe et al. [50]. As argued in [32] and [50], unlike ν, this benchmark enjoys natural monotonicity properties and is easier to work with.

Definition 3 (Benchmark $\mu(\mathbf{c})$ [32]). *Given a set system $(\mathcal{E}, \mathcal{F})$, and a feasible set $S \in \mathcal{F}$ of minimum total cost w.r.t. $\mathbf{c}$, let $\mu(\mathbf{c})$ be the value of an optimal solution to the following optimization problem:*

$$
\begin{array}{ll}
\max & \displaystyle\sum_{e \in S} b(e) \\
\text{s.t.} & (1)\ b(e) \geq c(e) \text{ for all } e \in \mathcal{E} \\
& (2)\ \displaystyle\sum_{e \in S \setminus T} b(e) \leq \sum_{e \in T \setminus S} c(e) \text{ for all } \ T \in \mathcal{F} \\
& (3) \text{ For every } e \in S \text{ there is a } T \in \mathcal{F} \text{ s.t. } e \notin T \\
& \qquad \text{and} \displaystyle\sum_{e' \in S \setminus T} b(e') = \sum_{e' \in T \setminus S} c(e')
\end{array}
$$

The programs for $\nu(\mathbf{c})$ and $\mu(\mathbf{c})$ differ in their objective function only: while $\nu(\mathbf{c})$ minimizes the total payment, $\mu(\mathbf{c})$ maximizes it. In particular, this means that in the program for $\mu(\mathbf{c})$ we can omit constraint (3), i.e., $\mu(\mathbf{c})$ can be obtained as a solution to a simpler linear program.

$$
\begin{aligned}
\max \quad & \sum_{e \in S} b(e) \\
s.t. \quad & (1)\ b(e) \geq c(e) \text{ for all } e \in \mathcal{E} \\
& (2) \sum_{e \in S \setminus T} b(e) \leq \sum_{e \in T \setminus S} c(e) \text{ for all } \ T \in \mathcal{F}
\end{aligned}
$$

Definition 4. *We will refer to the quantity* $\sup_{\mathbf{c}} \frac{p_{\mathcal{M}}(\mathbf{c})}{\mu(\mathbf{c})}$ *as the* μ-frugality ratio *of a truthful mechanism* $\mathcal{M}$*, where* $p_{\mathcal{M}}(\mathbf{c})$ *is the total payment of mechanism* $\mathcal{M}$ *on a bid vector* $\mathbf{c}$*.*

2.5 $\nu-$, $\mu-$Benchmarks for k-paths Set System

In this section we develop useful intuition about $\nu-$ and $\mu-$benchmark specifically for the k-paths set systems.

Proposition 1. *Each of* $\nu-$ *and* $\mu-$*benchmarks is a Nash equilibrium of the first-price auction.*

We first need the following definition.

Definition 5 (Minimum Longest Path $\delta_{k+1}(G, \mathbf{c})$)**.** *For any* $k+1$ *edge-disjoint s-t paths* $P_1, \ldots, P_{k+1}$ *in a directed graph* G*, let* $\delta_{k+1}(P_1, \ldots, P_{k+1}, \mathbf{c})$ *denote the length of the longest s-t path w.r.t. cost vector* $\mathbf{c}$ *in the subgraph* G' *composed of* $P_1, \ldots, P_{k+1}$ *(if* G' *contains a positive length cycle, set* $\delta_{k+1}(P_1, \ldots, P_{k+1}, \mathbf{c}) = +\infty$*). Define*

$$
\delta_{k+1}(G, \mathbf{c}) = \min \big\{ \delta_{k+1}(P_1, \ldots, P_{k+1}, \mathbf{c}) \mid P_1, \ldots, P_{k+1} \\ \text{are } k+1 \text{ edge-disjoint } s\text{-}t \text{ paths} \big\}.
$$

Our next lemma gives us a lower bound on $\nu(\mathbf{c})$ in terms of $\delta_{k+1}(G, \mathbf{c})$ and crucially relies on the characterization of Nash equilibria given by Theorem 2.

Lemma 1. *For any k-path system on a given graph G with costs $\mathbf{c}$, we have $\nu(\mathbf{c}) \geq k \cdot \delta_{k+1}(G, \mathbf{c})$.*

Proof. Fix a cost vector $\mathbf{c}$. Let E' be the winning set with respect to $\mathbf{c}$ in the first-price auction, and consider a bid vector $\mathbf{b}$ that satisfies conditions (1)–(3) in the definition of $\nu(\mathbf{c})$. Let $p(\mathbf{b})$ denote the total payment under $\mathbf{b}$. The set E' contains k edge-disjoint s-t paths. By condition (2), no agent in E' can obtain more revenue by increasing his bid. That is, for any $e \in E'$, there are k edge-disjoint s-t paths in $G \setminus \{e\}$ with the same total bid as E'. Applying Theorem 2 with $w(e) = b(e)$, we obtain that there are $k+1$ edge-disjoint shortest s-t paths with length $\frac{p(\mathbf{b})}{k}$ each w.r.t. $\mathbf{b}$. Consider the subgraph G' composed by these $k+1$ edge-disjoint paths. We know that $\delta_{k+1}(G', \mathbf{c}) \leq \frac{p(\mathbf{b})}{k}$ as $b(e) \geq c(e)$ for any edge e, i.e., the length of the longest s-t path in G' w.r.t. $\mathbf{c}$ is at most $\frac{p(\mathbf{b})}{k}$. Hence,

$$p(\mathbf{b}) \geq k \cdot \delta_{k+1}(G', \mathbf{c}) \geq k \cdot \delta_{k+1}(G, \mathbf{c}).$$

As this holds for any vector $\mathbf{b}$ that satisfies conditions (1)–(3), it follows that $\nu(\mathbf{c}) \geq k\delta_{k+1}(G, \mathbf{c})$. □

Consider an arbitrary network $\mathcal{F}$ with source s, sink t, integer edge capacities and costs per unit flow that are given by a vector $\mathbf{c}$. Let M be the value of the maximum flow in $\mathcal{F}$. For any (real) $x \in [0, M]$, let $C(x)$ be the cost of a cheapest flow of size x in $\mathcal{F}$ (i.e., the sum of costs on all edges, where the cost on an edge e is the amount of flow times $\mathbf{c}(e)$). The following lemma establishes several properties of the function $C(x)$ that will be used in the further bound on $\mu(\mathbf{c})$.

Lemma 2.

1. *$C(x)$ is a convex function on $[0, M]$.*

2. *For any integer $i \leq M-1$, $C(x)$ is a linear function on the interval $[i, i+1]$.*

Proof.

- **Convexity.** It suffices to show that for any $0 \leq \alpha \leq 1$ we have

  $$\alpha C(x) + (1-\alpha)C(y) \geq C(\alpha x + (1-\alpha)y).$$

  Let F_x and F_y be cheapest flows of size x and y, respectively. Their respective costs are $C(x)$ and $C(y)$. Then the flow $F = \alpha F_x + (1-\alpha)F_y$ is a flow of size $(\alpha x + (1-\alpha)y)$ and cost $\alpha C(x) + (1-\alpha)C(y)$ that satisfies all capacity constraints. Thus,

  $$\alpha C(x) + (1-\alpha)C(y) = \mathbf{c}(F) \geq C(\alpha x + (1-\alpha)y).$$

- **Linearity on intervals.** We first show that $C(x)$ is linear on the interval $[0,1]$. Let us fix $x_0 \in [0,1]$ and let F be a cheapest flow of size x_0. We can represent F as a finite sum of positive flows along s-t paths $p_1, \ldots, p_l$, i.e.,

  $$F = \sum_{i=1}^{l} \varepsilon_i p_i.$$

  We know that $C(1)$ is the cost of cheapest path p. Thus we have

  $$C(x_0) = \mathbf{c}(F) \geq \mathbf{c}(p) \sum_{i=1}^{l} \varepsilon_i = C(1)x_0.$$

  On the other hand, since $C(x)$ is convex, we have $x_0 C(1) + (1-x_0)C(0) \geq C(x_0)$. Hence $C(x_0) = x_0 C(1)$.

  In general, for the interval $[i, i+1]$ we first take a cheapest i-flow F_i (which we can choose to be integer) and consider the residual network $\mathcal{F}_i = \mathcal{F} - F_i$. We can then apply the argument for the $[0,1]$-case to $\mathcal{F}_i$.

$\square$

Our next lemma bounds $\mu(\mathbf{c})$ in terms of the difference between the cost of the cheapest flow of size k and that of the cheapest flow of size $k+1$.

Lemma 3. *Let $(\mathcal{E}, \mathcal{F})$ be a k-path system given by a directed graph $G = (V, E)$, source s and sink t, and let $\mathbf{c}$ be its cost vector. Then for the function $C(x)$ defined above we have $k \cdot (C(k+1) - C(k)) \leq \mu(\mathbf{c})$.*

Proof. For any cost vector $\mathbf{y} \in \mathbb{R}^{|E|}$, let $C_{\mathbf{y}}(x)$ denote the cost of the cheapest flow of size x in G with respect to the cost vector $\mathbf{y}$; we have $C_{\mathbf{c}}(x) = C(x)$.

Let F_k be a cheapest flow of size k, and let F_{k+1} be a cheapest flow of size $k+1$, both with respect to cost vector $\mathbf{c}$. Let n_k denote the number of edges in F_k. Assume without loss of generality that the edges in F_k are labeled as $e_1, \ldots, e_{n_k}$.

We will now gradually increase the costs of edges in F_k so that the resulting cost vector $\mathbf{y}$ satisfies certain conditions. Specifically, we start with $\mathbf{y} = \mathbf{c}$. Then, at i-th step, $i = 1, \ldots, n_k$, we increase $\mathbf{y}(e_i)$ as much as possible subject to the following constraints:

(a) $\mathbf{y}(F_k) = \sum_{e \in F_k} \mathbf{y}(e) = C_{\mathbf{y}}(k)$, i.e., F_k must remain the cheapest k-flow w.r.t. cost vector $\mathbf{y}$.

(b) $C_{\mathbf{y}}(k+1) - C_{\mathbf{y}}(k) = C(k+1) - C(k)$, i.e., $C_{\mathbf{y}}(k+1) - C_{\mathbf{y}}(k)$ does not change.

Since our k-path system is monopoly-free, in the end, all entries of $\mathbf{y}$ are finite. Further, when the process is over, we cannot increase the cost of any edge in F_k without violating (a) or (b).

Now, for each edge $e \in F_k$, we will define the *tight flow* $F(e)$ as below to be a flow that prevented us from raising $\mathbf{y}(e)$ beyond its current value. Specifically, consider each edge $e_i \in F_k$. Suppose first that when we were raising $\mathbf{y}(e_i)$, we had to stop because constraint (a) became tight. In this case, let $F(e_i)$ be some cheapest flow of size k in $G \setminus \{e_i\}$ with respect to the costs $\mathbf{y}$ at the end of stage i. Now, suppose that when we were raising $\mathbf{y}(e_i)$, constraint (b) became tight first. In this case, let $F(e_i)$ be some cheapest flow of size $k+1$ in $G \setminus \{e_i\}$ with respect to the costs $\mathbf{y}$ at the end of stage i. Observe that F_k remains a cheapest k-flow throughout the process; further, for all $e \in F_k$, the flow $F(e)$ is a cheapest flow of its size in G with respect to the final cost vector as well. In the following we consider the cost vector $\mathbf{y}$ at the end of the process.

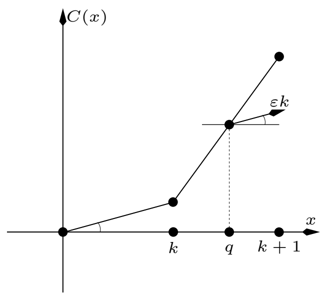

Figure 2.1: The graph of $C_{\mathbf{y}}(x)$

Let F^* be the average of all tight flows, i.e., set

$$F^* = \frac{1}{n_k} \sum_{e \in F_k} F(e).$$

Let q be the value of F^*; we have $k \le q \le k+1$. Note that F^* is a cheapest flow of size q by the second statement of lemma 2, as it is a convex combination of cheapest flows of size k and cheapest flows of size $k+1$. Further, since $e \notin F(e)$ for any $e \in F_k$, the amount of flow that passes through each edge e in F^* is strictly less than 1. Thus, for a sufficiently small $\varepsilon > 0$, flow $F^* + \varepsilon F_k$ is a valid flow of size $q + \varepsilon k$ in G. Moreover, we have

$$C_{\mathbf{y}}(q + \varepsilon k) \le \mathbf{y}(F^* + \varepsilon F_k) = C_{\mathbf{y}}(q) + \varepsilon C_{\mathbf{y}}(k).$$

This observation, together with the convexity of $C_{\mathbf{y}}(x)$, allows us to derive that $C_{\mathbf{y}}(x)$ is a linear function on the interval $[0, k+1]$. Indeed, by convexity of $C_{\mathbf{y}}(x)$ we have

$$\begin{aligned} C_{\mathbf{y}}(k) \le & \tfrac{q+(\varepsilon-1)k}{q+\varepsilon k} C_{\mathbf{y}}(0) + \tfrac{k}{q+\varepsilon k} C_{\mathbf{y}}(q+\varepsilon k) &= \frac{k}{q+\varepsilon k} C_{\mathbf{y}}(q+\varepsilon k) \\ C_{\mathbf{y}}(q) \le & \tfrac{\varepsilon k}{q+\varepsilon k} C_{\mathbf{y}}(0) + \tfrac{q}{q+\varepsilon k} C_{\mathbf{y}}(q+\varepsilon k) &= \frac{q}{q+\varepsilon k} C_{\mathbf{y}}(q+\varepsilon k) \end{aligned}$$

If any of the two inequalities above is an equality, then $C_{\mathbf{y}}(x)$ is linear on $[0, k+1]$ and we are done. Otherwise, both of these inequalities are strict, and using the above inequality $C_{\mathbf{y}}(q+\varepsilon k) \le C_{\mathbf{y}}(q) + \varepsilon C_{\mathbf{y}}(k)$ we can write

$$C_{\mathbf{y}}(q+\varepsilon k) \le C_{\mathbf{y}}(q)+\varepsilon C_{\mathbf{y}}(k) < \frac{q}{q+\varepsilon k}C_{\mathbf{y}}(q+\varepsilon k)+\varepsilon\frac{k}{q+\varepsilon k}C_{\mathbf{y}}(q+\varepsilon k) = C_{\mathbf{y}}(q+\varepsilon k).$$

The contradiction shows that $C_{\mathbf{y}}(x)$ is a linear function on $[0, k+1]$.

Since $\mathbf{y}$ satisfies conditions (1) and (2) in the definition of $\mu(\mathbf{c})$, we obtain

$$\mu(\mathbf{c}) \ge C_{\mathbf{y}}(k) = k(C_{\mathbf{y}}(k+1) - C_{\mathbf{y}}(k)) = k(C_{\mathbf{c}}(k+1) - C_{\mathbf{c}}(k)),$$

where the first equality follows from linearity of $C_{\mathbf{y}}(x)$ and last equality holds by construction of $\mathbf{y}$. Thus, the lemma is proven. □

Chapter 3

Frugal Mechanism Design 2: Incentive Compatible Mechanisms

3.1 Towards Incentive Compatible Mechanisms

We propose a general uniform scheme, which we call Pruning-Lifting Mechanism, for designing frugal truthful mechanisms for arbitrary set systems. At a high-level view, this mechanism consists of two key steps: pruning and lifting.

- Pruning. In a general set system, the relationships among the agents can be arbitrarily complicated. Thus, in the pruning step, we remove agents from the system so as to expose the structure of the competition. Intuitively, the goal is to keep only the agents who are going to play a role in determining the bids in Nash equilibrium; this enables us to compare the payoffs of our mechanism to the total equilibrium payment. Since we decide which agents to prune based on their bids, we have to make our choices carefully so as to preserve truthfulness.
- Lifting. The goal of the lifting process is to "lift" the bid of each remaining agent so as to take into account the size of each feasible set. For this purpose, we use a graph-theoretic approach inspired by the ideas in [49].

Namely, we construct a graph $\mathcal{H}$ whose vertices are agents, and there is an edge between two agents e and e' if removing both e and e' results in a system with no feasible solution. We call $\mathcal{H}$ the dependency graph of the pruned system. We then compute the largest eigenvalue of $\mathcal{H}$ (or, more precisely, the maximum of the largest eigenvalues of its connected components), which we denote by $\alpha_{\mathcal{H}}$, and scale the bid of each agent by the respective coordinate of the eigenvector that corresponds to $\alpha_{\mathcal{H}}$.

A given set system may be pruned in different ways, thus leading to different values of $\alpha_{\mathcal{H}}$. We will refer to the largest of them, i.e., $\alpha = \sup_{\mathcal{H}} \alpha_{\mathcal{H}}$, as the *eigenvalue* of our set system. It turns out that this quantity plays an important role in our analysis.

3.1.1 Frugality Performance

We apply our scheme to two classes of set systems: vertex cover systems, where the goal is to buy a vertex cover in a given graph, and k-path systems, where the goal is to buy k edge-disjoint paths between two specified vertices of a given graph. We note that that the r-out-of-k-system mechanism and the $\sqrt{\ }$-mechanism for the single path problem that were presented in [49] can be viewed as instantiations of our Pruning-Lifting Mechanism. In an r-out-of-k system, the set of agents $\mathcal{E}$ is a union of k disjoint subsets $S_1, \ldots, S_k$ and the feasible sets are unions of exactly r of those subsets.

Thus the k-path problem generalizes both the r-out-of-k problem and the single path problem, and captures many other natural scenarios. However, this problem received limited attention from the algorithmic mechanism design community so far (see, however, [47]), perhaps due to its inherent difficulty: the interactions among the agents can be quite complex, and, till recently, it was not known how to characterize Nash equilibria of the first-price auctions for this setting in terms of the network structure. In this chapter, we obtain a strong lower bound on the total payments in Nash equilibria. We then use this bound to show that a natural variant of the Pruning-Lifting Mechanism that prunes all edges except those in the cheapest flow of size $k+1$ has frugality ratio $\alpha\frac{k+1}{k}$. Moreover, we show that this bound can be improved by a factor of $k+1$ if we consider μ-benchmark, which in fact turns out to be the optimal μ-frugality value of any k-paths system.

For the vertex cover problem, an earlier work [32] described a mechanism with frugality ratio 2Δ, where Δ is the maximum degree of the input graph. Our approach results in a mechanism whose frugality ratio equals to the largest eigenvalue α of the adjacency matrix of the input graph. As $\alpha \leq \Delta$ for any graph G, this means that we improve the result of [32] by at least a factor of 2 for all graphs.

3.1.2 Lower Bounds

We complement the bounds on the frugality of the Pruning-Lifting Mechanism by proving lower bounds on the frugality of (almost) any truthful mechanism. In more detail, we exhibit a family of cost vectors on which the payment of any *measurable* truthful mechanism can be lower-bounded in terms of α, where we call a mechanism measurable if the payment to any agent—as a function of other agents' bids—is a Lebesgue measurable function. Lebesgue measurability is a much weaker condition than continuity or monotonicity; indeed, a mechanism that does not satisfy this condition is unlikely to be practically implementable! Our argument relies on Young's inequality and applies to any set system.

To turn this lower bound on payments into a lower bound on frugality, we need to understand the structure of Nash equilibria for the bid vectors employed in our proof. For k-path systems, we can achieve this by using the characterization of Nash equilibria in such systems given in the Chapter 2. As a result, we obtain a lower bound on frugality of any measurable truthful mechanism that shows that our mechanism is within a factor of $(k+1)$ from optimal. Moreover, it is, in fact, optimal, with respect to the μ-benchmark.

For the vertex cover problem, characterizing the Nash equilibria turns out to be a more difficult task: in this case, the graph $\mathcal{H}$ is equal to the input graph, and therefore is not guaranteed to have any regularity properties. However, we can still obtain non-trivial upper bounds on the payments in Nash equilibria. These bounds enable us to show that our mechanism for vertex cover is optimal for all triangle-free graphs, and, more generally, for all graphs that satisfy a simple local sparsity condition.

3.2 Pruning-Lifting Mechanism

In this section, we describe in detail a general scheme for designing truthful mechanisms for set systems, which we call Pruning-Lifting Mechanism. For a given set system $(\mathcal{E}, \mathcal{F})$, the mechanism is composed of the following steps:

- Pruning. The goal of the pruning process is to drop some elements of $\mathcal{E}$ to expose the structure of the competition between the agents; we denote the set of surviving agents by $\mathcal{E}^*$. We require the process to satisfy the following properties:

  Monotonicity: for any given vector of other agents' bids, if an agent e is dropped when he bids b, he is also dropped if he bids any $b' > b$. We set $t_1(e) = \inf\{b' \mid e \text{ is dropped when bidding } b'\}$.

  Bid-independence: for any given vector of other agents' bids $\mathbf{b}_{-e}$, let b_e and b'_e be two bids of agent e. If $e \in \mathcal{E}^*(b_e, \mathbf{b}_{-e})$ and $e \in \mathcal{E}^*(b'_e, \mathbf{b}_{-e})$, then $\mathcal{E}^*(b_e, \mathbf{b}_{-e}) = \mathcal{E}^*(b'_e, \mathbf{b}_{-e})$. That is, e cannot control the outcome of the pruning process as long as he survives. Monotonicity and bid-independence conditions are important to ensure the truthfulness of the mechanism.

  Monopoly-freeness: the remaining set system must remain monopoly-free, i.e., $\bigcap_{S \in \mathcal{F}^*} S = \emptyset$, where $\mathcal{F}^* = \{S' \in \mathcal{F} \mid S' \subseteq \mathcal{E}^*\}$. This condition is necessary because in the winner selection stage we will choose a winning feasible set from $\mathcal{F}^*$. Therefore, we have to make sure that no winning agent can charge an arbitrarily high price due to lack of competition.

- Lifting. The goal of the lifting process is to assign a weight to each agent in $\mathcal{E}^*$ so as to take into account the size of each feasible set. To this end, we construct an undirected graph $\mathcal{H}$ (see Fig. 3.1 for an example) by

  (a) introducing a node v_e for each $e \in \mathcal{E}^*$,

  (b) connecting v_e and $v_{e'}$ if and only if every feasible set in $\mathcal{F}^*$ contains e or e' (or both of them).

  We will refer to $\mathcal{H}$ as the *dependency graph* of $\mathcal{E}^*$. For each connected component $\mathcal{H}_j$ of $\mathcal{H}$, compute the largest eigenvalue α_j of its adjacency matrix A_j, and let $(w(v_e))_{v_e \in \mathcal{H}_j}$ be the eigenvector of A_j associated with α_j. That is, $A_j \mathbf{w}_j = \alpha_j \mathbf{w}_j$, where $\mathbf{w}_j = \left((w(v_e))_{v_e \in \mathcal{H}_j}\right)^T$. Set $\alpha = \max \alpha_j$.

- Winner selection. Define $b'(e) = \frac{b(e)}{w(v_e)}$ for each $e \in \mathcal{E}^*$, and select a feasible set $S \in \mathcal{F}^*$ with the smallest total bids w.r.t. $\mathbf{b}'$. We observe that every feasible set in $\mathcal{F}^*$ must be a vertex cover of $\mathcal{H}$. Although, in general not every vertex cover of $\mathcal{H}$ must be a feasible set. Let $t_2(e)$ be the threshold bid for $e \in \mathcal{E}^*$ to be selected at this stage.

- Payment. The payment to each winner $e \in S$ is $p(e) = \min\{t_1(e), t_2(e)\}$, where $t_1(e)$ and $t_2(e)$ are the two thresholds defined above.

Recall that the largest eigenvalue of the adjacency matrix of a connected graph is positive and its associated eigenvector has strictly positive coordinates [38]. Therefore, $w(v_e) > 0$ for all $e \in \mathcal{E}^*$.

We will now define a quantity $\alpha_{(\mathcal{E},\mathcal{F})}$ that will be instrumental in characterizing the frugality ratio of truthful mechanisms on $(\mathcal{E}, \mathcal{F})$. Let $\mathcal{S}(\mathcal{E}, \mathcal{F})$ be the collection

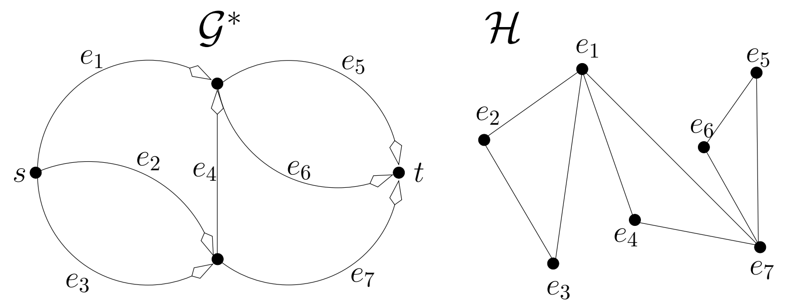

Figure 3.1: An example of the construction of $\mathcal{H}$ from G^* for 2-paths set system

of all monopoly-free subsets of $\mathcal{E}$, i.e., $\mathcal{S}(\mathcal{E},\mathcal{F}) = \{S \subseteq \mathcal{E} \mid \bigcap_{T\in\mathcal{F},T\subseteq S} T = \emptyset\}$. The elements of $\mathcal{S}(\mathcal{E},\mathcal{F})$ are the possible outcomes of the pruning stage. For any subset $S \in \mathcal{S}(\mathcal{E},\mathcal{F})$, let $\mathcal{H}_S$ be its dependency graph and let A_S be the adjacency matrix of $\mathcal{H}_S$. Let α_s be the largest eigenvalue of A_S (or the maximum of the largest eigenvalues of the adjacency matrices of the connected components of $\mathcal{H}_S$, if $\mathcal{H}_S$ is not connected). Set $\alpha_{(\mathcal{E},\mathcal{F})} = \max_{S\in\mathcal{S}(\mathcal{E},\mathcal{F})} \alpha_s$; we will refer to $\alpha_{(\mathcal{E},\mathcal{F})}$ as the *eigenvalue of the set system* $(\mathcal{E},\mathcal{F})$.

Note that once $\mathcal{E}^* \in \mathcal{S}(\mathcal{E},\mathcal{F})$ is selected in the pruning step, the computation of α and the weight vector $(w(v_e))_{e\in\mathcal{E}^*}$ does not depend on the bid vector. This property is crucial for showing that our mechanism is truthful.

Theorem 3. *The Pruning-Lifting Mechanism is truthful for any set system* $(\mathcal{E},\mathcal{F})$.

Proof. For any agent $e \in \mathcal{E}$ and given bids of other agents, we will analyze the utility of e in terms of his bid. We consider the following two cases.

Case 1 Agent e is not dropped during the pruning process when bidding $b(e) = c(e)$, i.e., $e \in \mathcal{E}^*$. By the definition of $t_1(e)$, we know that $t_1(e) \geq c(e)$. Consider the situation where e bids another value $b'(e) \neq b(e)$. If $b'(e) > t_1(e)$, then $e \notin \mathcal{E}^*$ and his utility is 0. If $b'(e) \leq t_1(e)$, by the bid-independence property, we know that the subset $\mathcal{E}^*$ remains the same. Given this fact, the structure of the graph $\mathcal{H}$ does not change, which implies that the eigenvectors and eigenvalues of its adjacency matrix do not change either. Hence, the threshold value $t_2(e)$ remains the same, which implies that the payment to agent e, $p(e) = \min\{t_1(e), t_2(e)\}$, will not change.

Case 2 Agent e is dropped during the pruning process when bidding $b(e) = c(e)$, i.e., $e \notin \mathcal{E}^*$. Consider the situation where e bids another value $b'(e) \neq b(e)$ and is not dropped out. By monotonicity and bid-independence, we know that $b'(e) \leq t_1(e) \leq b(e) = c(e)$. Hence, even though e could be a winner by bidding $b'(e)$, his payment is at most $t_1(e) \leq c(e)$, which implies that he cannot obtain a positive utility.

The case analysis above shows that the utility of each agent is maximized by bidding his true cost, and hence the mechanism is truthful. $\square$

In the rest of this section, we will show that the mechanisms for r-out-of-k systems and single path systems proposed in [49] can be viewed as instantiations of our Pruning-Lifting Mechanism. By Theorems 1 and 3, we can ignore the payment rule in the following discussions.

3.2.1 r-out-of-k Systems Revisited

In an r-out-of-k system, the set of agents $\mathcal{E}$ is a union of k disjoint subsets $S_1, \ldots, S_k$ and the feasible sets are unions of exactly r of those subsets. Given a bid vector $\mathbf{b}$, renumber the subsets $S_1, \ldots, S_k$ in order of non-decreasing bids, i.e., $b(S_1) \leq b(S_2) \leq \cdots \leq b(S_k)$.

The mechanism proposed in [49] deletes all but the first $r+1$ subsets, and then solves a system of equations given by

$$\beta = \frac{1}{rx_i} \cdot \sum_{j \neq i} x_j \cdot |S_j| \quad \text{for } i = 1, \ldots, r+1. \qquad (\Diamond)$$

It then scales the bid of each set S_i by setting $b'(S_i) = \frac{b(S_i)}{x_i}$, discards the set with the highest scaled bid w.r.t. $\mathbf{b}'$, and outputs the remaining sets.

Now, clearly, the first step of this mechanism can be interpreted as a pruning stage. Further, for r-out-of-k systems the graph $\mathcal{H}$ constructed in the lifting stage of our mechanism is a complete $(r+1)$-partite graph. It is not hard to verify that for any positive solution $(x_1, \dots, x_{r+1}, \beta)$ of the equation system $(\Diamond)$, $\beta \cdot r$ gives the largest eigenvalue of the adjacency matrix of $\mathcal{H}$ and $(x_1, \dots, x_1, \dots, x_{r+1}, \dots, x_{r+1})$ is the corresponding eigenvector. Thus, the mechanism of [49] implements Pruning-Lifting Mechanism for r-out-of-k systems.

In [49] it is shown that the frugality ratio of this mechanism is β, and the frugality ratio of any truthful mechanism for r-out-of-k systems is at least $\frac{\beta}{2}$. As r-out-of-k systems can be viewed as a special case of r-path systems, Theorem 9 allows us to improve this lower bound to $\frac{\beta r}{r} = \beta$.

3.2.2 Single Path Mechanisms Revisited

In a single path system, agents are edges of a given directed graph $G = (V, E)$ with two specified vertices s and t, i.e., $\mathcal{E} = E$ and $\mathcal{F}$ consists of all sets of edges that contain a path from s to t.

Given a bid vector $\mathbf{b}$, the $\sqrt{\ }$-mechanism [49] first selects two edge-disjoint s-t paths P and P' that minimize $b(P) + b(P')$. Assume that P and P' intersect at $s = v_1, v_2, \dots, v_{\ell+1} = t$, where the vertices are listed in the order in which they appear in P and P'. Let P_i and P'_i be the subpaths of P and P' from v_i to v_{i+1}, respectively. The $\sqrt{\ }$-mechanism sets $b'(e) = b(e)\sqrt{|P_i|}$ for $e \in P_i$, $b'(e) = b(e)\sqrt{|P'_i|}$ for $e \in P'_i$, and chooses a cheapest path in $P \cup P'$ w.r.t. $\mathbf{b}'$.

As in the previous case, the selection of P and P' can be viewed as the pruning process. The corresponding graph $\mathcal{H}$ consists of ℓ connected components, where the i-th component $\mathcal{H}_i$ is a complete bipartite graph with parts of size $|P_i|$ and $|P'_i|$. Its largest eigenvalue is given by $\alpha_i = \sqrt{|P_i||P'_i|}$, and the coordinates of the corresponding eigenvector are given by $w(v_e) = 1/\sqrt{|P_i|}$ for $e \in P_i$ and

$w(v_e) = 1/\sqrt{|P_i'|}$ for $e \in P_i'$. Thus, the $\sqrt{\ }$-mechanism can be viewed as a special case of the Pruning-Lifting Mechanism. It is shown that the frugality ratio of the $\sqrt{\ }$-mechanism is within a factor of $2\sqrt{2}$ from optimal; Using Pruning-Lifting Mechanism this bound can be improved by a factor of $\sqrt{2}$ (this has also been shown by Yan [69] via a proof that is considerably more complicated than ours).

3.3 Vertex Cover Systems

In the vertex cover problem, we are given a connected graph $G = (V, E)$ whose vertices are owned by selfish agents. Our goal is to purchase a vertex cover of G. That is, we have $\mathcal{E} = V$, and $\mathcal{F}$ is the collection of all vertex covers of G. Let A denote the adjacency matrix of G, and let Δ, $\alpha = \alpha_{(\mathcal{E},\mathcal{F})}$ and $\mathbf{w} = (w(v))_{v\in V}$ denote, respectively, the maximum degree of G, the largest eigenvalue of A and the corresponding eigenvector.

We will use the pruning-lifting scheme to construct a mechanism whose frugality ratio is α; this improves the bound of 2Δ given in [32] by at least a factor of 2 for all graphs, and by as much as a factor of $\Theta(\sqrt{n})$ for some graphs (e.g., the star).

Observe first that the vertex cover system plays a special role in the analysis of the performance of the pruning-lifting scheme. Indeed, on the one hand, it is straightforward to apply the Pruning-Lifting Mechanism to this system: since removing any agent will make each of its neighbors a monopolist, the pruning stage of our scheme is redundant, i.e., $\mathcal{H} = G$. That is, there is a unique implementation of Pruning-Lifting Mechanism for vertex cover systems: we set $b'(v) = \frac{b(v)}{w(v)}$ for all $v \in V$, pick any $S \in \operatorname{argmin}\{b'(T) \mid T \text{ is a vertex cover for } G\}$ to be the winning set, and pay each agent $v \in S$ his threshold bid. On the other hand, for general set systems, any feasible set in the pruned system corresponds to a vertex cover of $\mathcal{H}$: indeed, by construction of the graph $\mathcal{H}$, any feasible set must contain at least one endpoint of any edge of $\mathcal{H}$. (In general, the converse is not true: a vertex cover of $\mathcal{H}$ is not necessarily a feasible set. However, for k-path systems it can be shown that any cover of $\mathcal{H}$ corresponds to a k-flow.)

We will now bound the frugality of Pruning-Lifting Mechanism for vertex cover systems.

Theorem 4. *The frugality ratio of Pruning-Lifting Mechanism for vertex cover systems on a graph G is at most $\alpha = \alpha_{(\mathcal{E},\mathcal{F})}$.*

Proof. By Theorem 3 our mechanism is truthful, i.e., we have $b(v) = c(v)$ for all $v \in V$. By optimality of S we have $b'(v) \leq \sum_{uv \in E, u \notin S} b'(u)$, and therefore v's payment satisfies $p(v) \leq w(v) \sum_{uv \in E, u \notin S} \frac{c(u)}{w(u)}$. Thus, we can bound the total payment of our mechanism as

$$\begin{aligned}
\sum_{v \in S} p(v) &\leq \sum_{v \in S} w(v) \sum_{uv \in E, u \notin S} \frac{c(u)}{w(u)} \\
&= \sum_{u \notin S} \frac{c(u)}{w(u)} \sum_{uv \in E} w(v) \\
&= \sum_{u \notin S} \frac{c(u)}{w(u)} \alpha w(u) \\
&= \alpha \sum_{u \notin S} c(u).
\end{aligned}$$

Lemma 8 in [32] shows that for any cost vector $\mathbf{c}$, we have $\nu(\mathbf{c}) \geq \sum_{u \notin S} c(u)$.

Lemma (Elkind et al. [32])**.** *For a vertex cover instance $G = (V, E)$ in which S is a minimum vertex cover, $\nu(\mathbf{c}) \geq c(V \setminus S)$*

Therefore, the frugality ratio of Pruning-Lifting Mechanism for vertex cover on G is at most α. $\square$

In Section 3.5.1 we show that our mechanism is optimal for a large class of graphs.

3.4 Multiple Paths Systems

In this section, we study in detail k-path systems for a given integer $k \geq 1$. In these systems, the set of agents $\mathcal{E}$ is the set of edges of a directed graph $G = (V, E)$ with two specified vertices $s, t \in V$. The feasible sets are sets of edges that contain k edge-disjoint s-t paths. Clearly, these set systems generalize both r-out-of-k systems and single path systems.

Our mechanism for k-path systems for a given directed graph G, which we call Pruning-Lifting k-Paths Mechanism, is a natural generalization of the $\sqrt{}$-mechanism [49]: In the pruning stage of our mechanism, given a bid vector $\mathbf{b}$, we pick $k+1$ edge-disjoint s-t paths $P_1, \ldots, P_{k+1}$ so as to minimize their total bid w.r.t. the bid vector $\mathbf{b}$. Clearly, this procedure is monotone and bid-independent. Let $G^*(\mathbf{b})$ denote the subgraph composed of these $k+1$ paths. The remaining steps of the mechanism (lifting, winner selection, payment determination) are the same as in the general case (Section 3.2). Since the Pruning-Lifting k-Paths Mechanism is an implementation of the Pruning-Lifting Mechanism, Theorem 3 implies that it is truthful.

Let $\mathcal{G}^{k+1}$ denote the set of all subgraphs of G that can be represented as a union of $k+1$ edge-disjoint s-t paths in G. For any $G' \in \mathcal{G}^{k+1}$, let $\mathcal{H}(G')$ denote the dependency graph of G', and let $\alpha(G')$ denote the maximum of the largest eigenvalues of the connected components of $\mathcal{H}(G')$. Set $\alpha_{k+1} = \max\{\alpha(G') \mid G' \in \mathcal{G}^{k+1}\}$. Let $L(G, \mathbf{c})$ be the length of the longest path in $G^*(\mathbf{c})$, where $G^*(\mathbf{c})$ is the output of our pruning process on the bid vector $\mathbf{c}$. Our next lemma gives an upper bound on the payment of our mechanism in terms of $L(G, \mathbf{c})$

Lemma 4. *For any k-path system on a given graph G with costs $\mathbf{c}$, the total payment of Pruning-Lifting k-Paths Mechanism is at most $\alpha(G^*(\mathbf{c}))L(G, \mathbf{c})$.*

Proof. Fix a cost vector $\mathbf{c}$ and set $G^* = G^*(\mathbf{c})$, $\mathcal{H} = \mathcal{H}(G^*(\mathbf{c}))$, $\alpha = \alpha(G^*(\mathbf{c}))$. Observe that since G^* is the cheapest collection of $k+1$ edge-disjoint paths in G, it is necessarily cycle-free. For each vertex $v \in V(\mathcal{H})$, let e_v be the corresponding edge in G^*.

We claim that there is a natural one-to-one correspondence between minimal vertex covers in $\mathcal{H}$ (i.e., vertex cover such that removing any node results in an uncovered edge) and k-flows in G^*. To show this, we need the following claim.

Claim 1. *Let u and v be two vertices of $\mathcal{H}$. Then $uv \notin E(\mathcal{H})$ if and only if there is an s–t path in G^* going through both e_u and e_v.*

Proof. If there is a path $P \subseteq G^*$ such that $e_u, e_v \in P$, then in $G^* \setminus \{e_u, e_v\}$ there are k edge-disjoint s–t paths. Hence there is no edge between u and v. Conversely, if $uv \notin E(\mathcal{H})$, then $G^* \setminus \{e_u, e_v\}$ has k edge-disjoint s–t paths. Removing these k paths from G^* leads to an s–t path going through e_u and e_v. □

For any k-flow in G^*, the remaining agents form an s–t path and hence (by Claim 1) an independent set in $\mathcal{H}$. On the other hand, since in G^* there are no cycles, for any independent set in $\mathcal{H}$ one can find a complete order $\prec$ of agents in $\mathcal{H}$ such that $u \prec v$ whenever e_u precedes e_v in an s–t path in G^*. Moreover, for any independent set in $\mathcal{H}$ we can find a single s–t path of G^* that contains all the edges that correspond to the vertices of this independent set. These two observations conclude the proof of one-to-one correspondence.

Recall that in the proof of Theorem 4, we upper-bounded the total payment to the winning set S in a vertex cover set system as $\sum_{v \in S} p(v) \leq \alpha \sum_{u \notin S} c(u)$. The same upper bound applies here as well, since we have a bijection between vertex covers and k-flows. Recall that $\bar{S} \subset V(\mathcal{H})$ corresponds to a path in G^*. Therefore, we have $\alpha \sum_{u \notin S} c(u) \leq \alpha(G^*(\mathbf{c}))L(G, \mathbf{c})$, since $L(G, \mathbf{c})$ is the length of the longest path in G^*. This concludes the proof of the lemma. □

We can now bound the frugality ratio of our mechanism as follows.

Theorem 5. *The frugality ratio of* Pruning-Lifting k-Paths Mechanism *is at most $\alpha_{k+1} \frac{k+1}{k}$.*

Proof. We will now show how Lemmas 1 and 4 imply Theorem 5. We fix an arbitrary cost vector $\mathbf{c}$. Suppose that in the pruning stage we pick a graph G^*. By Lemma 4, the total payment of our mechanism is at most $\alpha(G^*)L(G,\mathbf{c})$. Since $G^* \in \mathcal{G}^{k+1}$, we have $\alpha(G^*) \leq \alpha_{k+1}$. Consider a collection $P_1, \ldots, P_{k+1}$ of $k+1$ edge-disjoint paths in G such that $\delta_{k+1}(P_1, \ldots, P_{k+1}, \mathbf{c}) = \delta_{k+1}(G, \mathbf{c})$. Since G^* is the cheapest collection of $k+1$ edge-disjoint paths in G, we have $\sum_{e \in G^*} c(e) \leq \sum_{i=1}^{k+1} \sum_{e \in P_i} c(e)$. We obtain

$$\begin{aligned} L(G,\mathbf{c}) &\leq \sum_{e \in G^*} c(e) \\ &\leq \sum_{i=1}^{k+1} \sum_{e \in P_i} c(e) \\ &\leq (k+1)\delta_{k+1}(P_1, \ldots, P_{k+1}) \\ &= (k+1)\delta_{k+1}(G,\mathbf{c}). \end{aligned}$$

Thus, the frugality ratio of Pruning-Lifting k-Paths Mechanism on $\mathbf{c}$ is at most

$$\begin{aligned} \frac{\alpha(G^*)L(G,\mathbf{c})}{k\delta_{k+1}(G,\mathbf{c})} &\leq \frac{\alpha_{k+1}(k+1)\delta_{k+1}(G,\mathbf{c})}{k\delta_{k+1}(G,\mathbf{c})} \\ &= \frac{\alpha_{k+1}(k+1)}{k}, \end{aligned}$$

which completes the proof of Theorem 5. □

Theorem 6. *The μ-frugality ratio of Pruning-Lifting k-Paths Mechanism is at most $\frac{\alpha_{k+1}}{k}$.*

Proof. We recall that $L(G,c)$ by definition is the length of the longest path in $G^*(\mathbf{c})$, where $G^* = G^*(\mathbf{c})$ is the cheapest $k+1$-flow w.r.t. $\mathbf{c}$. Hence, $C(k+1) = \mathbf{c}(G^*)$. Further we can decompose G^* into the sum of a path with the cost $L(G,c)$ and some k-flow, which implies that $C(k+1) \geq L(G,c) + C(k)$. Rewriting the last inequality we get $C(k+1) - C(k) \geq L(G,c)$. Combining this observation with Lemma 4 we easily derive Theorem 6. □

3.5 Lower Bounds

We say that a mechanism $\mathcal{M}$ for a set system $(\mathcal{E}, \mathcal{F})$ is *measurable* if the payment $p(e)$ of any agent $e \in \mathcal{E}$ is a Lebesgue measurable function of all agents' bids. We will now use Young's inequality [43] to give a lower bound on total payments of any measurable truthful mechanism with bounded frugality ratio.

Theorem 7 (Young's inequality)**.** *Let $f_1 : [0, a] \to \mathbb{R}^+ \cup \{0\}$ and $f_2 : [0, b] \to \mathbb{R}^+ \cup \{0\}$ be two Lebesgue measurable functions that are bounded on their domain. Assume that whenever $y > f_1(x)$ for some $0 < x \le a$, $0 < y \le b$, we have $x \le f_2(y)$. Then*

$$\int_0^a f_1(x)dx + \int_0^b f_2(y)dy \ge ab.$$

This inequality follows from the observation that $\int_0^a f_1(x)dx$ equals to the measure of points $\{(x, y) \mid 0 < x \le a, 0 < y \le f_1(x)\}$, whereas $\int_0^b f_2(y)dy$ equals to the measure of points $\{(x, y) \mid 0 < y \le b, 0 < x \le f_2(y)\}$. These two sets cover $\{(x, y) \mid 0 < x \le a, 0 < y \le b\}$, so the sum of their measures is at least ab.

Fix a set system $(\mathcal{E}, \mathcal{F})$ with $|\mathcal{E}| = n$ and let $S_{(\mathcal{E},\mathcal{F})} \in \mathcal{S}(\mathcal{E}, \mathcal{F})$ be a subset with $\alpha_S = \alpha_{(\mathcal{E},\mathcal{F})}$ (recall that $\mathcal{S}(\mathcal{E}, \mathcal{F})$ is the collection of all monopoly-free subsets and $\alpha_{(\mathcal{E},\mathcal{F})}$ is the eigenvalue of the system). For any $e \in S_{(\mathcal{E},\mathcal{F})}$, let $\mathbf{c}_{e,x}$ denote a bid vector where e bids x, all agents in $S_{(\mathcal{E},\mathcal{F})} \setminus \{e\}$ bid 0, and all agents in $\mathcal{E} \setminus S_{(\mathcal{E},\mathcal{F})}$ bid $n + 1$.

Lemma 5. *For any set system $(\mathcal{E}, \mathcal{F})$ and any measurable truthful mechanism $\mathcal{M}$ with bounded frugality ratio, there exists an agent $e \in S_{(\mathcal{E},\mathcal{F})}$ and a real value $0 < x \le 1$ such that the total payment of $\mathcal{M}$ on the bid vector $\mathbf{c}_{e,x}$ is at least $\alpha_{(\mathcal{E},\mathcal{F})}x$.*

Proof. Set $S = S_{(\mathcal{E},\mathcal{F})}$, $\mathcal{H} = \mathcal{H}_S$, $A = A_S$, $\alpha = \alpha_S = \alpha_{(\mathcal{E},\mathcal{F})}$. We will assume from now on that $\mathcal{H} = (S, E(\mathcal{H}))$ is connected; if this is not the case, our argument can be applied without change to the connected component of $\mathcal{H}$ that corresponds to α. Let $\mathbf{w} = (w_v)_{v \in S}$ be the eigenvector of A that is associated with α. By normalization, we can assume that $\max_{v \in S} w_v = 1$.

The proof is by contradiction: assume that there is a truthful mechanism $\mathcal{M}$ that pays less than αx on any bid vector of the form $\mathbf{c}_{e,x}$ for all $e \in S$ and all $0 < x \leq 1$. Recall that for any such bid vector, the cost of each agent in $\mathcal{E} \setminus S$ is $n + 1$. Since $\alpha \leq n$ and $x \leq 1$, this implies that $\mathcal{M}$ never picks any agents from $\mathcal{E} \setminus S$ on any $\mathbf{c}_{e,x}$, i.e., effectively $\mathcal{M}$ operates on S. For any edge vu of $\mathcal{H}$ and any $x > 0$, let $p_{uv}(x)$ denote the payment to v on the bid vector $\mathbf{c}_{u,x}$. Observe that measurability of $\mathcal{M}$ implies that $p_{uv}(x)$ is measurable (since it is a restriction of a measurable function). In this notation, our assumption can be restated as

$$\sum_{uv \in E(\mathcal{H})} p_{uv}(x) < \alpha x \tag{*}$$

for all $u \in S$ and any $0 < x \leq 1$.

It is easy to see that given a bid vector $\mathbf{c}_{u,z}$ with $0 < z \leq 1$, $\mathcal{M}$ never selects u as a winner. Indeed, suppose that u wins given $\mathbf{c}_{u,z}$. Then by the truthfulness of $\mathcal{M}$, if we reduce u's true cost from z to 0, u still wins and receives a payment of at least z. Since the set system restricted to S is monopoly-free, the resulting cost vector $\mathbf{c}'$ satisfies conditions (1)–(3) in the definition of ν, and hence $\nu(\mathbf{c}') = 0$. Thus the frugality ratio of $\mathcal{M}$ is $+\infty$, a contradiction. By the construction of $\mathcal{H}$, this means that any $v \in S$ with $uv \in E(\mathcal{H})$ wins given $\mathbf{c}_{u,z}$.

Now, fix some x, y such that $0 < x, y \leq 1$ and $y > p_{vu}(x)$, and consider a situation where v bids x, u bids y, all agents in $S \setminus \{u, v\}$ bid 0, and all agents in $\mathcal{E} \setminus S$ bid $n+1$. Clearly, in this situation agent u loses and thus v wins with a payment of at least x. By the truthfulness of $\mathcal{M}$, the same holds if v lowers his bid to 0. Thus, for any $0 < x, y \leq 1$, $y > p_{vu}(x)$ implies $p_{uv}(y) \geq x$.

By our assumption, we have $p_{uv}(x) \leq \alpha x$, $p_{vu}(x) \leq \alpha x$ for $x \in [0, 1]$. Hence, for any $uv \in E(\mathcal{H})$ the functions $p_{uv}(x)$ and $p_{vu}(x)$ satisfy all conditions of Young's inequality on $[0, 1]$.

Let $A = (a_{uv})_{u,v \in S}$, and consider the scalar product $\langle \mathbf{w}, A\mathbf{w} \rangle = \langle \mathbf{w}, \alpha\mathbf{w} \rangle = \alpha\langle \mathbf{w}, \mathbf{w} \rangle$. We have $\langle \mathbf{w}, A\mathbf{w} \rangle = \sum_{uv \in E(\mathcal{H})} w_u w_v$. As we normalized $\mathbf{w}$ so that

$w_u, w_v \leq 1$, by Young's inequality, we can bound $w_u w_v$ by

$$\int_0^{w_u} p_{uv}(x)dx + \int_0^{w_v} p_{vu}(x)dx.$$

Therefore,

$$\begin{aligned}
\alpha\langle \mathbf{w}, \mathbf{w}\rangle &= \sum_{u,v\in S} a_{uv} w_u w_v \\
&\leq \sum_{u,v\in S} \left(\int_0^{w_u} a_{uv} p_{uv}(x)dx + \int_0^{w_v} a_{uv} p_{vu}(y)dy \right) \\
&= 2\sum_{u\in S} \int_0^{w_u} \sum_{v\in S} a_{uv} p_{uv}(x)dx \\
&< 2\alpha \sum_{u\in S} \int_0^{w_u} x dx \\
&= \alpha \sum_{u\in S} w_u^2 = \alpha\langle \mathbf{w}, \mathbf{w}\rangle,
\end{aligned}$$

where the last inequality follows from (*). This is a contradiction, so the proof is complete. □

3.5.1 Vertex Cover Systems

For vertex cover systems, deleting any of the agents would result in a monopoly. Therefore, Lemma 5 simply says that for any measurable truthful mechanism $\mathcal{M}$ on a graph $G = (V, E)$, there exists a $v \in V$ such that the total payment on bid vector $x \cdot \mathbf{c}_v$ is at least αx, where α is the largest eigenvalue of the adjacency matrix of G and $\mathbf{c}_v$ is the cost vector given by $c_v(u) = 1$ if $u = v$, and $c_v(u) = 0$ if $u \in V \setminus \{v\}$.

Given a graph $G = (V, E)$ and a vertex $v \in V$, let $\mathcal{CL}_v$ denote the set of all maximal cliques in G that contain v. Let ρ_v denote the size of the smallest clique in $\mathcal{CL}_v$.

Lemma 6. *We have $\nu(x \cdot \mathbf{c}_v) \leq x(\rho_v - 1)$ for any $x > 0$.*

Proof. Let C_v be some clique of size ρ_v in $\mathcal{CL}_v$, and consider the bid vector $\mathbf{b}$ given by $b(u) = x$ if $u \in C_v$ and $b(u) = 0$ if $u \in V \setminus C_v$. Since C_v is a clique, any vertex cover for G must contain at least $\rho_v - 1$ vertices of C_v. Thus, any cheapest feasible set with respect to the true costs contains all vertices in $C_v \setminus \{v\}$; let S denote some such set. Moreover, for any $u \in C_v \setminus \{v\}$, any vertex cover that does not contain u must contain v, so $\mathbf{b}$ satisfies condition (2) with respect to the set S in the definition of the benchmark ν. To see that is also satisfies condition (3), note that if any vertex in $C_v \setminus \{v\}$ decides to raise its bid, it can be replaced by its neighbors at cost x. Now, consider any $w \in (V \setminus C_v) \cap S$. The vertex w cannot be adjacent to all vertices in C_v, since otherwise C_v would not be a maximal clique. Thus, if $w \in S$, we can obtain a vertex cover of cost $x(\rho_v - 1)$ that does not include w by taking all vertices of cost 0 as well as all vertices in C_v that are adjacent to w. $\square$

Combining Lemma 6 with Lemma 5 yields the following result.

Theorem 8. *For any graph G, the frugality ratio of any measurable truthful vertex cover auction on G is at least $\frac{\alpha}{\rho-1}$, where α is the largest eigenvalue of the adjacency matrix of G, and $\rho = \max_{v \in V} \rho_v$.*

The bound given in Lemma 6 is not necessarily optimal; we can construct a family of graphs where for some vertex v the quantity ρ_v is linear in the size of the graph, while $\nu(\mathbf{c}_v) = O(1)$. Nevertheless, Theorem 8 shows that the mechanism described in Section 3.3 has optimal frugality ratio for, e.g., all triangle-free graphs and, more generally, all graphs G such that the for each vertex $v \in G$, the induced subgraph on the neighbors of v contains an isolated vertex.

3.5.2 Multiple Path Systems

Let $(\mathcal{E}, \mathcal{F})$ be a k-path system on a graph $G = (V, E)$. Consider a set $S \in \mathcal{S}(\mathcal{E}, \mathcal{F})$ with $\alpha_S = \alpha_{(\mathcal{E},\mathcal{F})}$. It is not hard to see that S is a union of $k+1$ edge-disjoint paths; this follows, e.g., from the proof of Theorem 2. Hence, we have $\alpha_{(\mathcal{E},\mathcal{F})} = \alpha_{k+1}$.

As before, for any $e \in S$, let $\mathbf{c}_{e,x}$ denote the cost vector with $\mathbf{c}_{e,x}(e) = x$, $\mathbf{c}_{e,x}(u) = 0$ for all $u \in S \setminus \{e\}$, $\mathbf{c}_{e,x}(w) = n + 1$ for all $w \in E \setminus S$. It is easy to see that we have $\mu(\mathbf{c}_{e,x}) = \nu(\mathbf{c}_{e,x}) = kx$ for any $e \in S$, $x > 0$. Combining this observation with Lemma 5, we obtain the following result.

Theorem 9. *For any graph $G = (V, E)$, both the frugality ratio and the μ-frugality ratio of any measurable truthful k-path mechanism on G are at least $\frac{\alpha_{k+1}}{k}$.*

In Section 3.4, we showed that the frugality ratio and the μ-frugality ratio of Pruning-Lifting k-Paths Mechanism are bounded by, respectively, $\alpha_{k+1}\frac{k+1}{k}$ and $\frac{\alpha_{k+1}}{k}$. Together with Theorem 9, this implies that Pruning-Lifting k-Paths Mechanism has optimal μ-frugality ratio; this gives further evidence that Pruning-Lifting k-Paths Mechanism is indeed the optimal mechanism for k-path systems.

3.6 Implementation & Concluding Remarks

We conclude this chapter with a discussion on the computationally efficient implementation of our two mechanisms for k-paths and vertex cover set systems.

k-paths set systems. The Pruning-Lifting k-Paths Mechanism as it is described can be implemented efficiently in polynomial in n time. Indeed, the pruning step requires nothing but a search for the cheapest $k+1$-flow from s to t, which can be done efficiently via standard min-cost max-flow algorithm. The construction of the undirected dependency graph $\mathcal{H}$ and computation of the eigenvector of its adjacency matrix corresponding to the maximal eigenvalue can be also performed in polynomial in n time. The final winners' selection is equivalent to the task of choosing the cheapest k-flow w.r.t. scaled costs and, therefore, admits an efficient implementation. Finally, computation of the payments would not be a problem, e.g., due to the fact that we can do a binary search for each of the threshold values. Summarizing all the results from previous sections we have

Theorem 10. *For an arbitrary k-paths set system Pruning-Lifting k-Paths Mechanism is an incentive compatible mechanism working in polynomial time. It has*

ν-frugality ratio within factor of $k+1$ from any other truthful mechanism and has the optimal μ-frugality ratio.

One might hope that a different pruning approach leads to a smaller ν-frugality ratio. In particular, the proof of Theorem 5 suggests that we could obtain a stronger result by pruning the graph so as to minimize the length of the longest path $\delta_{k+1}(G^*, \mathbf{c})$ in the surviving graph G^*. While —under truthful bidding—such mechanism would have an optimal frugality ratio, unfortunately, this pruning process would not be monotone [25].

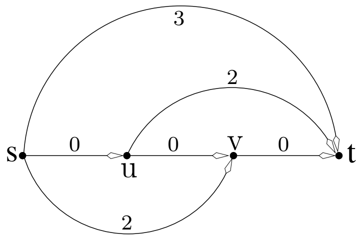

Figure 3.2: An instance of 1-path problem on which Pruning-Lifting k-Paths Mechanism does not choose the path that minimizes $\delta_2(G, \mathbf{c})$

Vertex Cover set systems. To implement Pruning-Lifting Mechanism for vertex covers, we need to select a vertex cover that minimizes the scaled costs given by $(b'(v))_{v \in V}$, i.e., to solve an NP-hard problem. Moreover, for a simple vector of costs $\mathbf{c}_{v,1}$, such that $c_v = 1$ and all the rest of the vertices have cost 0, it is hard to approximate $\mu(\mathbf{c}_{v,1})$. As was pointed out in [50] $\mu(\mathbf{c}_{v,1})$ is the fractional clique number of the graph induced by the neighbors of v, without v itself. This implies that unless ZPP=NP, $\mu(\mathbf{c}_{v,1})$ cannot be computed in polynomial time even approximately within a factor of $O(n^{1-\epsilon})$ for any $\epsilon > 0$. Such a strong inapproximability result for computing efficiently the value of the benchmark casts doubt on the existence of *efficient* truthful mechanism that performs almost as well as the optimal mechanism and also casts doubt on the importance of an efficient implementation in the frugality framework for vertex cover set systems.

On the positive side, the argument in Theorem 4 applies to any truthful mechanism that selects a *locally optimal* solution, i.e., a vertex cover S that satisfies $b'(v) \leq \sum_{uv \in E, u \notin S} b'(u)$ for all $v \in S$. Paper [32] argues that any monotone winner selection algorithm for vertex cover can be transformed into a locally optimal one, and shows that a variant of the classic 2-approximation algorithm for this problem [8] is monotone. This leads to a truthful polynomial time mechanism, although with a hard-to-estimate ratio between its payment and each of μ- or ν- benchmark.

Part II

Budget Feasible Mechanism Design

Chapter 4

Budget Feasible Mechanism Design 1: Basic Model

4.1 The model

It is well-known that a mechanism may have to pay a large amount to enforce incentive compatibility (i.e., truthfulness). For example, as was discussed in Chapter 2, the seminal VCG mechanism may have unbounded payment (compared to the shortest path) in path auctions. The negative effect of truthfulness on payments leads to a study of frugal mechanism design, i.e., how should one minimize his payment to get a feasible output with selfish agents? While we discussed a few interesting instances of frugal mechanism design in the previous chapters, in practice, one cannot expect to have big over payments for a few perspectives, e.g., budget or resource limit.

Recently, Singer [63] initiated a study of mechanism design of reverse auctions from a different prospective. Let us consider a procurement combinatorial auction problem, where a buyer wishes to purchase some resources from a set of agents A. Each agent $i \in A$ is able to supply a resource at an incurred cost c_i. The buyer has a budget B that gives an upper bound on the compensation which is distributed among agents, and a function $v(\cdot)$ describing the valuation that the buyer obtains for every subset of A. This defines a natural optimization problem: find a subset $S \subseteq A$ that maximizes $v(S)$ subject to $\sum_{i \in S} c_i \leq B$. Such combinatorial optimization problems have been considered in a variety of domains. In particular, one may recall the following familiar optimization problems:

Knapsack: Given a budget B and a set of items $N = \{1, \ldots, n\}$ where each item has a cost c_i and a value v_i, find a $S \subset N$, which maximizes $\sum_{i \in S} v_i$ and satisfies budget constraint $\sum_{i \in S} c_i \leq B$.

Coverage: Given a budget B and subsets $N = \{T_1, \ldots, T_n\}$ of a ground set, each T_i with a cost c_i, find a subset $S \subset N$ that maximizes $\bigcup_{i \in S} T_i$ under the budget constraint $\sum_{i \in S} c_i \leq B$.

Matching: Given a budget B and graph G, where $E(G) = \{e_1, \ldots, e_n\}$ and each edge e has a cost c_e and a value v_e, find a matching $S \subset E$ that maximizes $\sum_{e \in S} v_e$ under the budget constraint.

Knapsack and coverage problems fall in a more general framework of submodular maximization, which received significant attention in the past few decades (e.g. [58, 66, 51, 35, 53] consider submodular optimization with various conditions).

Singer proposed studying these problems in a game theoretic scenario, where one assumes agents to be self-interested entities so that each agent would want to get as much subsidy as possible. In particular, an agent can conceal his true incurred cost c_i (which is known only to himself) and claim 'any' amount b_i instead. Thus, given submitted bids b_i from all agents, a *mechanism* decides on a winning set $S \subseteq A$ and a payment p_i to each winner $i \in S$. A mechanism is called *truthful* (a.k.a. incentive compatible) if for every agent it is a dominant strategy to bid his true cost, i.e., $b_i = c_i$. In this model we assume that agents may behave strategically, whereas the buyer may not. We thus assume that the information related to the buyer, i.e., budget B and valuation function $v(\cdot)$, is a public knowledge to everyone.

Our mechanism design problem has an important and practical ingredient: the budget, i.e., the total payment of a mechanism should be upper bounded by B. Although budget is a realistic condition that appears almost everywhere in daily life, it has not received much attention until recently [29, 11, 15, 63, 21, 9, 7]. In the framework of worst case analysis, most results are negative [29]. A possible explanation for this could be that budget constraint introduces a new dimension to mechanism design and restricts the space of truthful mechanisms. For example, in single-parameter domains where the private information of every individual is a single value (which is the case in our model), a monotone allocation rule with associated threshold payments provides a sufficient and necessary condition for truthfulness [57]. However, it may not necessarily generate a budget feasible solution. Thus, a number of classic truthful designs as the seminal VCG mechanism [68, 23, 42] do not apply, and new ideas have to be developed.

Another significant challenge due to the budget constraint is that, unlike the VCG mechanism which always generates a socially optimal solution, we cannot hope to have a solution that is both socially optimal and budget feasible even if we are given unlimited computational power. Indeed, in a simple setting like path procurement with 0 or 1 valuations [63], any budget feasible mechanism may have an arbitrarily bad solution. Therefore, the question that one may ask is "under which valuation domains do there exist truthful budget feasible mechanisms that admit 'small' approximations (compared to the socially optimal solution)?"

The answer to this question depends on the properties of the valuation function under consideration. As the example of path procurement shows, a valuation with complements does not work well in our framework. Therefore, it seems natural to focus on complement-free valuations. There is a broad classification of various definitions of complement-free valuations briefly summarized in following hierarchy [54]:

$$\begin{aligned} \text{additive} \ \subset \ \text{gross substitutes} \ &\subset \ \text{submodular} \\ \subset \ \text{XOS} \ &\subset \ \text{subadditive}, \end{aligned}$$

In this chapter we provide several constant approximations for the Additive(knapsack), Submodular (includes coverage problem), XOS (includes matching) classes of valuations and give $O(\log n)$-approximation for subbaditive $v(\cdot)$. The case of general subadditive valuations seems to be the most challenging and interesting, as it was remarked in [30]:

> *"A fundamental question is whether, regardless of computational constraints, a constant-factor budget feasible mechanism exists for subadditive functions."*
>
> — Dobzinski, Papadimitriou, Singer

For the reader's convenience we summarize the currently best known results in the Table 4.1 below. We note that all randomized mechanism below are universally truthful mechanisms, i.e., randomization is taken over deterministic truthful mechanisms.

4.2 Preliminaries

In a marketplace there are n agents denoted by $A = \{1, \ldots, n\}$, each offering a single item for sale. Each agent i has a privately known incurred *cost* c_i (or denoted by $c(i)$). We denote by $\mathbf{c} = (c(i))_{i \in A}$ the cost vector of the agents. For any given subset $S \subseteq A$ of agents, there is a publicly known valuation $v(S)$, meaning the buyer's welfare derived from S. We assume $v(\emptyset) = 0$ and $v(S) \leq v(T)$ for any $S \subset T \subseteq A$.

We list below the definitions of various subclasses of complement-free valuations moving from more basic classes to the most general one.

	Additive valuation				Submodular valuation			
	deterministic		randomized		deterministic		randomized	
	upper	lower	upper	lower	upper	lower	upper	lower
Singer [63]	5	2	–	–	–	2	117.7	–
Chapter 4	$2+\sqrt{2}$	$1+\sqrt{2}$	3	2	8.34*	$1+\sqrt{2}$	7.91	2

*It may require exponential running time for general monotone submodular functions.

	XOS valuation		Subadditive valuation	
	deterministic	randomized	deterministic	randomized
Dobzinski et al. [30]	–	–	$O(\log^3 n)$	$O(\log^2 n)$
Chapter 4	–	$O(1)$	–	$O(\frac{\log n}{\log\log n})$

Table 4.1: Columns "Randomized" and "Deterministic" indicate whether a truthful mechanism is allowed to use random coin flips. The columns "lower" and "upper" denote the upper and lower bounds on approximation ratio obtained for corresponding randomized or deterministic truthful mechanisms.

Additive: $v(S)=\sum_{i\in S} v(\{i\}) = \sum_{i\in S} v_i$ for any $S\subseteq A$.

Submodular: $v(S)+v(T)\ge v(S\cap T)+v(S\cup T)$ for any $S,T\subseteq A$.

XOS: there is a set of additive functions $f_1,\dots,f_m$ such that for any $S\subseteq A$

$$v(S)=\max\big\{f_1(S), f_2(S),\dots,f_m(S)\big\}.$$

Note that the number of functions m can be exponential in $n=|A|$.

Subadditive : $v(S)+v(T)\ge v(S\cup T)$ for any $S,T\subseteq A$.

All three examples of the problems mentioned in Section 4.1 nicely fit into these hierarchy of classes. In particular, the knapsack problem corresponds exactly to the case of additive valuations, coverage is a special class of submodular functions, and matching is an example of XOS valuation (for every legal matching M_i of G let $f_i(S) = \sum_{e \in S \cap M_i} v_i$).

An equivalent definition of submodular valuation can be formulated in terms of the property of decreasing marginal returns, i.e., $v(S \cup \{i\}) - v(S) \geq v(T \cup \{i\}) - v(T)$ for any $S \subseteq T \subseteq A$ and $i \in A$.

For XOS (a.k.a. fractionally subadditive) valuation there is also an equivalent definition [34] $\forall S \subset A$:

$$\begin{aligned}
& v(S) \leq \sum_{T \subseteq A} x(T) \cdot v(T) & \\
s.t. \quad & \sum_{T \subseteq A:\ i \in T} x(T) \geq 1 & \forall i \in S \\
& 0 \leq x(T) \leq 1 & \forall T \subset A
\end{aligned}$$

That is, if every element in S is fractionally covered, then the sum of the values of all subsets weighted by the corresponding fractions is at least as large as $v(S)$.

We note that the representation of a submodular/XOS/subadditive valuation function usually requires exponential size in n. Thus, we assume that we are given access to a *demand oracle*, which, for any given price vector $p = (p_1, \ldots, p_n)$, returns us a subset

$$T \in \underset{S \subseteq A}{\text{argmax}} \left(v(S) - \sum_{i \in S} p(i) \right).$$

Every such a query is assumed to take unit time. The demand oracle is used in [30] as well, and it was shown that a weaker value query oracle is not sufficient [63].

Upon receiving a *bid* cost b_i from each agent, a mechanism decides upon an *allocation* $S \subseteq A$ as winners and a *payment* p_i to each $i \in A$. We assume that a mechanism may not make positive transfers (i.e., $p_i = 0$ if $i \notin S$) and is individually rational (i.e., $p_i \geq b_i$ if $i \in S$). Agents bid strategically on their costs and would like to maximize their quasi-linear utilities, which are given for every agent i as $p_i - c_i$, if i is a winner, and 0 otherwise. We say a mechanism is *incentive compatible* or *truthful* if it is in the best interest for each agent to report his true cost. For randomized mechanisms, we consider universal truthfulness in this paper (i.e., a randomized mechanism takes a distribution over deterministic truthful mechanisms).

Our setting is in single parameter domain, as each agent has only one cost to report on. For single parameter domains it is well-known [57] that a mechanism is truthful if and only if its allocation rule is monotone (i.e., a winner keeps winning if he unilaterally decreases his bid) and the payment to each winner is his threshold bid (i.e., the maximal bid for which the agent still wins). Therefore, we will only focus on designing monotone allocations and do not specify the payment to each winner explicitly. Although we will always implicitly verify that the total payment is below the budget B.

A mechanism is said to be *budget feasible* if $\sum_i p_i \leq B$ on every instance of bids and randomness in the mechanism, where B is a given budget constraint. Assume without loss of generality that $c_i \leq B$ for any agent $i \in A$, since otherwise he would never win in any (randomized) budget feasible truthful mechanism. Our objective is to design truthful budget feasible mechanisms with every output being approximately close to the social optimum. That is, we want to minimize the *approximation ratio* of a mechanism, which is defined as $\max_I \frac{opt(I)}{\mathcal{M}(I)}$, where $\mathcal{M}(I)$ is the (expected) value of mechanism $\mathcal{M}$ on instance I and $opt(I)$ is the optimal value of the integer program: $\max_{S \subseteq A} v(S)$ subject to $c(S) \leq B$, where $c(S) = \sum_{i \in S} c_i$.

4.3 Additive Valuation.

We consider here the most basic case where the valuation of the buyer is additive, i.e., $v(S) = \sum_{i \in S} v_i$ for any $S \subseteq A$. This leads to an instance of Knapsack problem, where items correspond to agents and the size of the knapsack corresponds to budget B.

We present below two truthful mechanism, one with randomized and another one with deterministic implementations. Our randomized mechanism is universally truthful, meaning that for any fixed realization of the random coin flips the mechanism is truthful as a deterministic mechanism.

4.3.1 Deterministic Mechanism.

We consider the following greedy strategy studied by Singer [63].

Greedy-KS(A)

1. `Order all items in` A `s.t.` $\frac{v_1}{c_1} \geq \frac{v_2}{c_2} \geq \cdots \geq \frac{v_{|A|}}{c_{|A|}}$
2. `Let` $k = 1$ `and` $S = \emptyset$
3. `While` $k \leq |A|$ `and` $c_k \leq B \cdot \frac{v_k}{\sum_{i \in S \cup \{k\}} v_i}$
   - $S \leftarrow S \cup \{k\}$
   - $k \leftarrow k + 1$
4. `Return winning set` S

It is shown that the above greedy strategy is monotone (and therefore truthful). Actually, it has the following remarkable property: any $i \in S$ cannot control the output set, given that i is guaranteed to be a winner. That is, if the winning sets are S and S', provided that i bids c_i and c_i', respectively, where $i \in S \cap S'$, then $S = S'$.

Otherwise, let i_0 be the item with the smallest index in $(S \setminus S') \cup (S' \setminus S)$. Without loss of generality let us assume $i_0 \in S \setminus S'$. That only could happen, if i_0 appeared earlier in S than i and i_0 was rejected from S' after i has been taken into S'. Let $T = \{j \in S \cap S' \mid j < i_0, j \neq i\}$ be the winning items in $S \cap S' \setminus \{i\}$ before i_0. Since i got accepted in S claiming the cost c_i after i_0, we have

$$\frac{v_{i_0}}{c_{i_0}} \geq \frac{v_i}{c_i} \geq \frac{\sum\limits_{j \in S, j \leq i)} v_j}{B} \geq \frac{\sum_{j \in T} v_j + v_i + v_{i_0}}{B}.$$

Rewriting the above inequality, we get

$$c_{i_0} \le B \cdot \frac{v_{i_0}}{\sum_{j\in T} v_j + v_i + v_{i_0}},$$

which implies that i_0 should be a winner in S' as well, a contradiction.

Given the greedy strategy described above, our mechanism for knapsack is as follows (where $f_{\text{opt}}(A)$ denotes the value of the optimal fractional solution to knapsack, which can be computed in polynomial time).

> Deterministic-KS
>
> 1. Let $A = \{i \mid c_i \le B\}$ and $i^* \in \text{argmax}_{i\in A}\, v_i$
> 2. If $(1+\sqrt{2}) \cdot v_{i^*} \ge f_{\texttt{opt}}(A \setminus \{i^*\})$, return i^*
> 3. Otherwise, return $S = \text{Greedy-KS}(A)$

Theorem 11. Deterministic-KS *is a deterministic budget feasible truthful mechanism that is a* $2+\sqrt{2}$ *approximation for any additive valuations.*

Proof. The proof proceeds by verifying each property stated in the claim.

- *Truthfulness.* We analyze monotonicity of the mechanism according to the condition of Steps 2 and 3, respectively. If i^* wins in Step 2 (note that the fractional optimal value computed in Step 2 is independent of the bid of i^*), then i^* still wins if he decreases his bid.

  If the condition in Step 2 fails and the mechanism continues to Step 3, for any $i \in S$, the subset S remains the same if i decreases his bid. Note that if $i \ne i^*$, when i decreases his bid, the value of the fractional optimal solution computed in Step 2 will not decrease. Hence i is still a winner, which implies that the mechanism is monotone and, therefore, combined with the threshold payments is incentive compatible.

- *Individual rationality and budget feasibility.* If i^* wins in Step 2, his payment is the threshold bid B. Otherwise, assume that all sellers in A are ordered by $1, 2, \ldots, n$; let $S = \{1, \ldots, k\}$. Note that it is possible that $i^* \in S$. For any $i \in S$, let q_i be the maximum cost that i can bid such that the fractional optimal value on instance $A \setminus \{i^*\}$ is still larger than $(1+\sqrt{2})v_{i^*}$. Note that $c_i \leq q_i$.

  Thus, the payment to any winner is

  $$p_i = \min\left\{v_i \cdot \frac{c_{k+1}}{v_{k+1}}, B \cdot \frac{v_i}{\sum_{j \in S} v_j}, q_i\right\}, \quad i \in S \setminus \{i^*\}$$

  and

  $$p_{i^*} = \min\left\{v_{i^*} \cdot \frac{c_{k+1}}{v_{k+1}}, B \cdot \frac{v_{i^*}}{\sum_{j \in S} v_j}\right\}, \quad \text{if } i^* \in S.$$

  Further, $\sum_{i \in S} p_i \leq \sum_{i \in S} B \cdot \frac{v_i}{\sum_{j \in S} v_j} = B$, which implies that the mechanism is budget feasible. The mechanism is individually rational, since we use a threshold payment rule.

- *Approximation.* Assume that all sellers in A are ordered by $1, 2, \ldots, n$, and $T = \{1, \ldots, k\}$ is the subset returned by Greedy-KS(A). Let ℓ be the maximal item for which $\sum_{i=1,\ldots,\ell} c_i \leq B$. Let $c'_{\ell+1} = B - \sum_{i=1,\ldots,\ell} c_i$ and $v'_{\ell+1} = v_{\ell+1} \cdot \frac{c'_{\ell+1}}{c_{\ell+1}}$. Hence, the optimal fractional solution is

  $$f_{\text{opt}}(A) = \sum_{i=1}^{\ell} v_i + v'_{\ell+1}.$$

  For any $j = k+1, \ldots, \ell$, we have

  $$\frac{c_j}{v_j} \geq \frac{c_{k+1}}{v_{k+1}} > \frac{1}{v_{k+1}} \cdot B \cdot \frac{v_{k+1}}{\sum_{i=1}^{k+1} v_i},$$

  where the last inequality follows from the fact that the greedy strategy stops at item $k+1$. Hence, $c_j > B \cdot \frac{v_j}{\sum_{i=1}^{k+1} v_i}$. As $\frac{c'_{\ell+1}}{v'_{\ell+1}} = \frac{c_{\ell+1}}{v_{\ell+1}}$, we have $c'_{\ell+1} > B \cdot \frac{v'_{\ell+1}}{\sum_{i=1}^{k+1} v_i}$. Therefore,

$$B \cdot \frac{\sum_{j=k+1}^{\ell} v_j + v'_{\ell+1}}{\sum_{i=1}^{k+1} v_i} < \sum_{j=k+1}^{\ell} c_j + c'_{\ell+1} < B,$$

which implies that $\sum_{i=1}^{k} v_i > \sum_{j=k+2}^{\ell} v_j + v'_{\ell+1}$. Hence

$$f_{\text{opt}}(A) = \sum_{i=1}^{\ell} v_i + v'_{\ell+1} < 2 \sum_{i \in S} v_i + v_{i^*}.$$

The following are some basic properties of the optimal fractional solution

$$f_{\text{opt}}(A) - v_{i^*} \leq f_{\text{opt}}(A \setminus \{i^*\}) \leq f_{\text{opt}}(A)$$

Hence, if the condition in Step 2 holds and the mechanism outputs i^*, we have

$$f_{\text{opt}}(A) \leq f_{\text{opt}}(A \setminus \{i^*\}) + v_{i^*} \leq (2 + \sqrt{2}) \cdot v_{i^*}$$

On the other hand, if the mechanism outputs S in Step 3, we have

$$\begin{aligned} (1 + \sqrt{2}) \cdot v_{i^*} &< f_{\text{opt}}(A \setminus \{i^*\}) \\ &\leq f_{\text{opt}}(A) \\ &< 2 \sum_{i \in S} v_i + v_{i^*} \end{aligned}$$

which implies that $v_{i^*} < \sqrt{2} \cdot \sum_{i \in S} v_i$. Hence,

$$\begin{aligned} opt(A) \leq f_{\text{opt}}(A) &= \sum_{i=1,\ldots,\ell} v_i + v'_{\ell+1} \\ &< 2 \sum_{i \in S} v_i + v_{i^*} \\ &\leq (2 + \sqrt{2}) \cdot \sum_{i \in S} v_i. \end{aligned}$$

Therefore, Deterministic-KS has approximation ratio of at most $(2 + \sqrt{2})$.

$\square$

4.3.2 Randomized Mechanism.

Our randomized mechanism for knapsack is as follows.

Random-KS

1. `Let` $A = \{i \mid c_i \leq B\}$ `and` $i^* \in \text{argmax}_{i \in A} v_i$
2. `With probability` $\frac{1}{3}$, `return` i^*
3. `With probability` $\frac{2}{3}$, `return` Greedy-KS(A)

Theorem 12. Random-KS *is a universal truthful budget feasible mechanism that is a* 3 *approximation for any additive valuation.*

Proof. Since both mechanisms in Steps 2 and 3 are budget feasible and truthful, it remains to establish the approximation ratio.

Using the same notation and argument as in the proof of Theorem 11, assume that all sellers in A are ordered from 1 to n, and $T = \{1, \ldots, k\}$ is the subset returned by Greedy-KS(A). Let ℓ be the maximal-value item for which $\sum_{i=1,\ldots,\ell} c_i \leq B$. Let $c'_{\ell+1} = B - \sum_{i=1,\ldots,\ell} c_i$ and $v'_{\ell+1} = c'_{\ell+1} \cdot \frac{v_{\ell+1}}{c_{\ell+1}}$. Hence, the optimal fractional solution is

$$f_{\text{opt}}(A) = \sum_{i=1}^{\ell} v_i + v'_{\ell+1}$$

and

$$f_{\text{opt}}(A) = \sum_{i=1}^{\ell} v_i + v'_{\ell+1} < v_{i^*} + 2\sum_{i \in S} v_i.$$

The expected value of Random-KS is therefore

$$\frac{1}{3}v_{i^*} + \frac{2}{3}\sum_{i \in S} v_i = \frac{1}{3}\Big(v_{i^*} + 2\sum_{i \in S} v_i\Big) > \frac{1}{3}opt.$$

□

4.4 Lower Bounds

In this section we discuss the lower bounds on the approximation ratio of truthful budget feasible mechanisms for additive valuations. In [63], a lower bound of 2 is obtained by the following argument. Consider the case with two items, each of value 1. If their costs are $(B - \epsilon, B - \epsilon)$, at least one item should win, otherwise the approximation ratio is infinite. Without loss of generality, one can assume that the first item wins, and as a result its payment is at least $B-\epsilon$. Now consider another profile $(\epsilon, B - \epsilon)$, the first item must win (due to monotonicity) and get a payment of at least $B - \epsilon$ due to truthfulness. The second item then does not win because of the budget constraint and individual rationality. Therefore, the mechanism may only achieve a value of 1 for such instance while the optimal solution is 2. This gives us the lower bound of 2.

We improve the deterministic lower bound to $1 + \sqrt{2}$ by a more involved argument. We also provide a lower bound of 2 for any randomized universally truthful mechanism. All these lower bounds are unconditional, i.e., one does not impose any complexity assumptions and constraints on the running time of the mechanism.

4.4.1 Lower Bound for Deterministic Mechanisms

Theorem 13. *No deterministic truthful budget feasible mechanism can achieve an approximation ratio better than $1 + \sqrt{2}$, even if there are only three items.*

Assume otherwise that there is a budget feasible truthful mechanism that can achieve a ratio better than $1 + \sqrt{2}$. We consider the following scenario: budget $B = 1$, and values $v_1 = \sqrt{2}$, $v_2 = v_3 = 1$. Then the mechanism for any bidding scenario satisfies the following two properties:

- if all items are winners in the optimal solution, the mechanism must output at least two items;

- if $\{1,2\}$ or $\{1,3\}$ is the optimal solution, the mechanism cannot output either $\{2\}$ or $\{3\}$ (i.e., a single item with unit value).

For any item i, let function $p_i(c_j, c_k)$ be the payment offered to item i given that the bids of the other two items are c_j and c_k. That is, $p_i(c_j, c_k)$ is the threshold bid of i to be a winner.

Lemma 7. *For any $c_3 > 0.5$ and any domain $(a,b) \subset (0, 1-c_3)$, there is $c_2 \in (a,b)$ such that $p_1(c_2, c_3) < 1 - c_2$.*

Proof. Assume otherwise that there are $c_3 > 0.5$ and $(a,b) \subset (0, 1-c_3)$ such that for any $c_2 \in (a,b)$, $p_1(c_2, c_3) \geq 1 - c_2$. Let $c_1 = 1 - c_3 - b$. Then $c_1 + c_2 + c_3 < 1 = B$, which implies that the mechanism has to output at least two items. Since $0 < c_1 = 1 - c_3 - b < 1 - c_2 \leq p_1(c_2, c_3)$, item 1 is a winner. Further, $p_1(c_2, c_3) \geq 1 - c_2 > 0.5$, which together with budget feasibility implies that item 3 cannot be a winner. Therefore, item 2 must be a winner with payment $p_2(c_1, c_3) = c_2$ due to individual rationality and budget feasibility. The same analysis still holds if the true cost of item 2 becomes $c_2' = \frac{c_2+b}{2}$, i.e., item 2 is still a winner with payment c_2'. Thus for the sample (c_1, c_2, c_3) the payment satisfies $p_2(c_1, c_3) \geq c_2' > c_2$, a contradiction. □

Since items 2 and 3 are identical, the above lemma still holds if we switch items 2 and 3 in the claim. We are now ready to prove Theorem 13.

Proof of Theorem 13. Define $c_3 = 0.7$ and $(a,b) = (0.2, 0.3)$. Note that c_3 and (a,b) satisfy the condition of Lemma 7. Hence, there is $c \in (0.2, 0.3)$ such that $p_1(c, 0.7) < 1 - c$. Define $p_1(c, 0.7) = 1 - c - x$, where $x > 0$. Symmetrically, define $c_2 = 0.7$ and $(a', b') = (c, \min\{0.3, c + x\})$. Again by Lemma 7, there is $d \in (a', b')$ such that $p_1(0.7, d) < 1 - d$. Define $p_1(0.7, d) = 1 - d - y$, where $y > 0$. Pick $c_1 = 1 - d - \epsilon$, where $\epsilon > 0$ is sufficiently small so that $c_1 \in (1 - c - x, 1 - c) \cap (1 - d - y, 1 - d)$. Note that since $d \in (c, c + x)$, c_1 is well-defined.

Consider a true cost vector $(c_1, c, 0.7)$. Since $p_1(c, 0.7) = 1 - c - x < c_1$, item 1 cannot be a winner. Since $c_1 + c = 1 - d - \epsilon + c < 1$, the optimal solution has a value of at least $v_1 + v_2 = 1 + \sqrt{2}$; therefore the mechanism has to output both items 2 and 3. Hence, $p_3(c_1, c) \geq c_3 = 0.7$.

Similarly, consider true cost vector $(c_1, 0.7, d)$; we have $p_2(c_1, d) \geq c_2 = 0.7$. Finally, consider cost vector (c_1, c, d). By the above two inequalities, both items 2 and 3 are the winners; this contradicts the budget feasibility. □

4.4.2 Lower Bound for Randomized Mechanisms

Theorem 14. *No randomized (universally) truthful budget feasible mechanism can achieve an approximation ratio better than* 2*, even in the case of two items.*

Proof. We use Yao's min-max principle, which is a typical tool used to prove lower bounds, where we need to design a distribution of instances and argue that any deterministic budget feasible mechanism cannot get an expected approximation ratio better than 2.

All the instances contain two items of value 1. Their costs (c_1, c_2) are drawn from the following distribution (see Fig. 4.1 for an example):

1. $(\frac{kB}{n}, \frac{(n-k)B}{n})$ with probability $\frac{1-\epsilon}{n-1}$, where $k = 1, 2, \ldots, n-1$,

2. $(\frac{iB}{n}, \frac{jB}{n})$ with probability $\frac{2\varepsilon}{(n-1)(n-2)}$, where $i, j \in \{1, \ldots, n-1\}$ and $i+j > n$,

where $1 > \varepsilon > 0$ and n is a large integer.

We first claim that for any deterministic truthful budget feasible mechanism with finite expected approximation ratio, there is at most one instance, for which both items win in the mechanism. Assume by contradiction that there are at least two such instances. Note that for the second distribution $(\frac{iB}{n}, \frac{jB}{n})$, where $i+j > n$, it cannot be the case that both items win due to the budget constraint. Hence, the two instances must be of the first type; denote them by $(\frac{k_1B}{n}, \frac{(n-k_1)B}{n})$ and $(\frac{k_2B}{n}, \frac{(n-k_2)B}{n})$, where $k_1 > k_2$. Consider the instance $(\frac{k_1B}{n}, \frac{(n-k_2)B}{n})$. Since

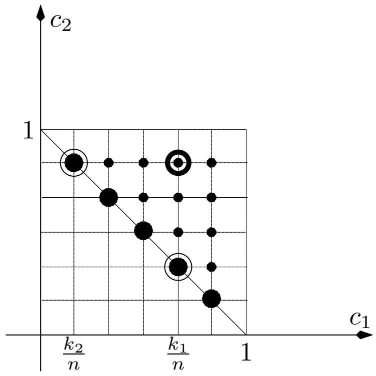

Figure 4.1: Distribution for $n = 6$. Diameter of a point emphasizes its probability.

$k_1 + n - k_2 > n$, this is the instance of the second type in our distribution. Therefore it has nonzero probability (see Fig. 4.1). The mechanism has a finite approximation ratio, thus it should have a finite approximation ratio on the instance $(\frac{k_1 B}{n}, \frac{(n-k_2)B}{n})$ as well. As a result, it cannot be the case that both items lose. We assume that item 1 wins (the proof for the other case is similar); the payment to him is at least $\frac{k_1 B}{n}$ due to individual rationality. Then consider the original instance $(\frac{k_2 B}{n}, \frac{(n-k_2)B}{n})$; item 1 should also win and get a threshold payment, which is equal to or greater than $\frac{k_1 B}{n}$. Therefore the payment to the second item is at most $B - \frac{k_1 B}{n} = \frac{(n-k_1)B}{n}$ because of the budget constraint. Since $\frac{(n-k_1)B}{n} < \frac{(n-k_2)B}{n}$, we arrive at a contradiction with either individual rationality or with the assumption that both items won in the instance $(\frac{k_2 B}{n}, \frac{(n-k_2)B}{n})$.

On the other hand, for all instances $(\frac{kB}{n}, \frac{(n-k)B}{n})$, both items win in the optimal solution with value 2. Hence, the expected approximation ratio of any deterministic truthful budget feasible mechanism is at least $\frac{1-\varepsilon}{n-1} \cdot 1 + (n-2) \cdot \frac{1-\varepsilon}{n-1} \cdot 2 + \varepsilon \cdot 1 = 2 - \varepsilon - \frac{1-\varepsilon}{n-1}$. The ratio approaches 2 as $\varepsilon \to 0$ and $n \to \infty$. $\square$

4.5 Submodular Valuations

For any given monotone submodular function, we denote the marginal contribution of an item i with respect to set S by $m_S(i) = v(S \cup \{i\}) - v(S)$. We assume that agents are sorted according to their non-increasing marginal contributions relative to their costs, recursively defined by: $i + 1 = \mathrm{argmax}_{j \in A \setminus S_i} \frac{m_{S_i}(j)}{c_j}$, where $S_i = \{1, \ldots, i\}$ and $S_0 = \emptyset$. To simplify notations we will denote this order by $[n]$ and write m_i instead of $m_{S_{i-1}}(i)$. This sorting, in the presence of submodularity, implies that

$$\frac{m_1}{c_1} \geq \frac{m_2}{c_2} \geq \cdots \geq \frac{m_n}{c_n}.$$

Notice that $v(S_k) = \sum_{i \leq k} m_i$ for all $k \in [n]$.

The following greedy scheme is the core of our mechanism (where the parameters denote the set of agents A and available budget $B/2$).

Greedy-SM$(A, B/2)$

1. Let $k = 1$ and $S = \emptyset$
2. While $k \leq |A|$ and $c_k \leq \frac{B}{2} \cdot \frac{m_k}{\sum_{i \in S \cup \{k\}} m_i}$
   - $S \leftarrow S \cup \{k\}$
   - $k \leftarrow k + 1$
3. Return winning set S

4.5.1 Randomized Mechanism

Our mechanism for general monotone submodular functions is as follows.[1]

[1]Our mechanism has a similar flavor to Singer's mechanism [63] for the greedy scheme and randomness between the greedy and the item with the largest value. Indeed, both arise from the algorithm that maximizes monotone submodular functions with weighted items [51]. Our mechanism, however, treats the greedy scheme and random selection in a slightly different way, which yields a much better approximation ratio.

Random-SM

1. `Let` $A = \{i \mid c_i \leq B\}$ `and` $i^* \in \text{argmax}_{i \in A} v(i)$
2. `with probability` 0.4, `return` i^*
3. `with probability` 0.6, `return` Greedy-SM$(A, B/2)$

In the above mechanism, if it returns i^*, the payment to i^* is B; if the mechanism returns Greedy-SM$(A, B/2)$, the payments are the threshold bids and are explicitly derived in [63]. Actually, in our analysis we do not need this explicit payment formula.

Theorem 15. Random-SM *is a budget feasible universally truthful mechanism for a submodular valuation function with an approximation ratio of* $\frac{5e}{e-1}(\approx 7.91)$.

Proof. The proof proceeds by verifying each property of the mechanism.

Universal Truthfulness. Our mechanism is a simple random combination of two mechanisms. To prove that the Random-SM is universally truthful, it suffices to prove that these two mechanisms are truthful respectively, i.e., the allocation rule is monotone.

The scheme where we simply return i^* is obviously truthful. Also it is easy to see that throwing away agents with a cost greater than B in the prior step does not affect truthfulness. The greedy scheme Greedy-SM$(A, B/2)$ is monotone as well, since any item out of a winning set cannot increase its bid to become a winner.

Budget Feasibility. While truthfulness is quite straightforward, the budget feasibility analysis turns out to be quite tricky. The difficulties arise while we compute the payment to each item. Indeed, it may happen that an agent changes his bid (while remaining in the winning set) and make the mechanism to change its output. In other words, computing the threshold bid of an item, we need to look at quite different outputs of the mechanism. Fortunately, in such a case no item can reduce the valuation of the output too much. That enables us to write down an upper bound on the bid of each item in the case of submodular valuation and obtain budget feasibility by summing up these bounds.

If the mechanism returns i^*, the agent's payment is B, which is clearly budget feasible. We are left to prove budget feasibility for $\mathsf{Greedy\text{-}SM}(A, B/2)$. A similar but a weaker result has been proven in [63], using the characterization of payments and arguing that the total payment is not larger than B. Here we directly show that the payment to any item i in the winning set S is bounded above by $\frac{m_i}{v(S)} \cdot B$; then the total payment will be bounded by B since $\frac{\sum_{i \in S} m_i}{v(S)} \cdot B = B$. Before doing that, we first prove a useful lemma.

Lemma 8. *Let $S \subset T \subseteq [n]$ and $t_0 = \text{argmax}_{t \in T \setminus S} \frac{m_S(t)}{c(t)}$. Then*

$$\frac{v(T) - v(S)}{c(T) - c(S)} \leq \frac{m_S(t_0)}{c(t_0)}.$$

Proof. Assume by contradiction that the lemma does not hold. Then

$$\frac{v(T) - v(S)}{c(T) - c(S)} > \frac{m_S(t)}{c(t)}, \qquad \text{for all} \quad t \in T \setminus S.$$

Summing up all the inequalities, each multiplied by $\frac{c(t)}{\sum_{t \in T \setminus S} c(t)}$, we have

$$\frac{v(T) - v(S)}{c(T) - c(S)} > \frac{\sum_{t \in T \setminus S} m_S(t)}{\sum_{t \in T \setminus S} c(t)} = \frac{\sum_{t \in T \setminus S} m_S(t)}{c(T) - c(S)}.$$

This implies that $v(T) - v(S) > \sum_{t \in T \setminus S} m_S(t)$, which contradicts the submodularity. □

Let $1, \ldots, k$ be the order of items in which we add them to the winning set. Let $\emptyset = S_0 \subset S_1 \subset \ldots \subset S_k \subseteq [n]$ be the sequence of winning sets that we pick at each step by applying our mechanism. Thus we have $S_j = \{1, \ldots, j\} = [j]$. Now, since $v(\cdot)$ is sumbodular, we can write the following chain of inequalities (note that marginal contribution is smaller for larger sets).

$$\frac{m_{S_0}(1)}{c_1} \geq \frac{m_{S_1}(2)}{c_2} \geq \ldots \geq \frac{m_{S_{k-1}}(k)}{c_k} \geq \frac{2v(S_k)}{B},$$

where the last inequality holds true due to Lemma 8 and the fact that $c(S_k) \le \frac{B}{2}$.

The following lemma concludes the proof of budget feasibility, since it shows that payment to each winning agent j does not exceed $m_{S_{j-1}}(j)\frac{B}{v(S)}$.

Lemma 9. *No agent $j \in S =$ Greedy-SM$(A, B/2)$ could bid more than $m_{S_{j-1}}(j)\frac{B}{v(S)}$ and be in the winning set.*

Proof. Assume that $S = S_k$ is the winning set and there is $j \in S_k$ such that it can bid $b_j > m_{S_{j-1}}(j)\frac{B}{v(S_k)}$ and still win (given fixed bids of others). In the following we will use $\mathbf{b}$ instead of $\mathbf{c}$ to denote that we are considering a new scenario where an agent j has increased his bid to b_j and bids of other agents remain the same ($b_i = c_i$ for $i \neq j$).

Note that

$$\frac{m_{S_0}(1)}{c_1} \ge \frac{m_{S_1}(2)}{c_2} \ge \ldots \ge \frac{m_{S_{j-1}}(j)}{c_j} \ge \frac{m_{S_{j-1}}(j)}{b_j}.$$

For a bid vector $\mathbf{b}$, we denote by $T \supseteq S_{j-1}$ the set of agents chosen earlier than j in the winning set. Thus, by the rule of the greedy mechanism, we have

$$j = \operatorname*{argmax}_{i \in [n] \setminus T} \frac{m_T(i)}{b_i}, \tag{4.1}$$

$$\frac{m_T(j)}{b_j} \ge \frac{2v(T \cup \{j\})}{B}. \tag{4.2}$$

We may assume $S_k \cup T \supsetneq T \cup \{j\}$. Indeed, otherwise $T \cup \{j\} = S_k \cup T$ and

$$\frac{m_{S_{j-1}}(j)}{b_j} \ge \frac{m_T(j)}{b_j} \ge \frac{2v(T \cup \{j\})}{B} \ge \frac{2v(S_k)}{B} \ge \frac{v(S_k)}{B}.$$

Thus $b_j \le m_{S_{j-1}}\frac{B}{v(S_k)}$ and we get a contradiction.

Let $R = S_k \setminus T$. Applying equation (4.1) and Lemma 8 to $S_k \cup T$ and $T \cup \{j\}$, we know that for some $r_0 \in R \setminus \{j\}$,

$$\frac{v(S_k \cup T) - v(T \cup \{j\})}{b(S_k \cup T) - b(T \cup \{j\})} \le \frac{m_{T \cup \{j\}}(r_0)}{b(r_0)} \le \frac{m_T(r_0)}{b(r_0)} \le \frac{m_T(j)}{b_j}.$$

On the other hand, since $b_j > m_{S_{j-1}}(j)\frac{B}{v(S_k)}$, we have

$$\frac{m_T(j)}{b_j} < \frac{m_T(j)}{m_{S_{j-1}}(j)}\frac{v(S_k)}{B} < \frac{v(S_k)}{B}.$$

Combining these inequalities, we get

$$\frac{v(S_k \cup T) - v(T \cup \{j\})}{b(S_k \cup T) - b(T \cup \{j\})} < \frac{v(S_k)}{B}.$$

We have

$$b(S_k \cup T) - b(T \cup \{j\}) = b(R \setminus \{j\}) = c(R \setminus \{j\}) \leq c(S_k).$$

Recall that $\frac{m_{S_{i-1}}(i)}{c_i} \geq \frac{2v(S_k)}{B}$ for $i \in [k]$. Thus $c_i \leq m_{S_{i-1}}(i)\frac{B}{2v(S_k)}$ and $c(S_k) = \sum_{i=1}^{k} c(i) \leq \frac{B}{2}$. We get

$$\begin{aligned} \frac{v(S_k) - v(T \cup \{j\})}{B/2} &\leq \frac{v(S_k) - v(T \cup \{j\})}{c(S_k)} \\ &\leq \frac{v(S_k \cup T) - v(T \cup \{j\})}{b(S_k \cup T) - b(T \cup \{j\})} \\ &< \frac{v(S_k)}{B}. \end{aligned}$$

Thus, $v(S_k) < 2v(T \cup \{j\})$. Finally, we derive

$$\frac{m_{S_{j-1}}(j)}{b_j} \geq \frac{m_T(j)}{b_j} \geq \frac{2v(T \cup \{j\})}{B} > \frac{v(S_k)}{B},$$

which contradicts with $b_j > m_{S_{j-1}}(j)\frac{B}{v(S_k)}$. □

Approximation Ratio. Throughout this part we need not bother with incentives issues and focus only on the approximation guarantees.

Before analyzing the performance of our mechanism, we consider the following simple greedy algorithm: order items according to their marginal contributions divided by costs and add as many items as possible (i.e., it stops when we cannot add the next item as the sum of c_i will exceed B). Moreover, we may consider the fractional variant of that scheme, i.e., on the all remaining budget we buy an affordable fraction of the last item. Let ℓ be the maximal index for which $\sum_{i=1,\dots,\ell} c_i \leq B$. Let $c'_{\ell+1} = B - \sum_{i=1,\dots,\ell} c_i$ and $m'_{\ell+1} = m_{\ell+1} \cdot \frac{c'_{\ell+1}}{c_{\ell+1}}$. Hence, the fractional greedy solution is defined as

$$f_{\text{gre}}(A) \triangleq \sum_{i=1}^{\ell} m_i + m'_{\ell+1}.$$

It is well-known that the greedy algorithm is a $1-1/e$ approximation for maximization of a monotone submodular functions with a cardinality constraint [58]. Also it was shown that the simple greedy algorithm may have an unbounded approximation ratio in case of weighted items with a capacity constraint. Nevertheless, a variant of greedy algorithm was suggested in [51] which gives the same $1-1/e$ approximation to the weighted case. The following lemma, which is fundamental in our analysis, establishes the same approximation ratio for the fractional greedy algorithm described above.

Lemma 10. *The fractional greedy solution has an approximation ratio of* $1-1/e$ *for the weighted submodular maximization problem. That is,*

$$f_{gre}(A) \geq (1-1/e) \cdot opt(A),$$

where $opt(A)$ *is the value of the optimal integral solution for the given instance* A*.*

Proof. Recall that c_i denotes the cost of each item $i \in [n]$. Our goal in the weighted problem is to pick a set S with total cost $\sum_{i\in S} c_i$ not exceeding given budget B of maximal possible value $v(S)$, where v is the given monotone submodular function. As the value $v_f(S_f)$ is for fractional problem we consider the expectation of $v(S)$, where each $i \in [n]$ is selected at random independently in S with probability equal to the fraction of item i in S_f.

Assume that all costs c_i are integers. We reduce our weighted problem with monotone submodular function v to the unweighted one as follows.

- For each item $i \in [n]$ we consider c_i new items of unit cost. Denote them as i_j for $j \in [c_i]$ and call i the type of the unit i_j.

- The new valuation function μ only depends on the amounts of unit items of each type.

- Let a set S contain a_i units of each type i. Independently for each type, pick at random in the set $\mathcal{R}$ with probability $\frac{a_i}{c_i}$ weighted item i. Define $\mu(S) = E(v(\mathcal{R}))$.

Therefore

$$\mu(S) = \frac{1}{c_1 \cdot \ldots \cdot c_n} \sum_{\pi} v\left(S \cdot \pi\right) \tag{4.3}$$

where π is a sampling of units, one for each type (there are $c_1 \cdot \ldots \cdot c_n$ variants for π); $S \cdot \pi$ is a vector of types for which sample π is an element of S.

Using (4.3) it is not hard to verify monotonicity and submodularity of μ. Indeed, e.g. to verify submodularity one only needs to check that the marginal contribution of any unit is smaller for a large set, i.e., for $S \subset T$ and $i_j \notin T$ verify inequality $\mu(S \cap \{i_j\}) - \mu(S) \geq \mu(T \cap \{i_j\}) - \mu(T)$.

For any $T \subseteq [n]$ if we consider the set of units $S = \{i_k | i \in T, 1 \leq k \leq c_i\}$. Then according to the definition $\mu(S) = v(T)$. Hence, the optimal solution to the unit costs problem is equal to or larger than the optimal solution to the original problem.

It remains to show that our fractional greedy scheme for an integer weighted instance gives us the same result as the greedy scheme for its unit weighted version. Note that once we have taken a unit of type i we will proceed to take units of type i until its supply gets exhausted completely (we break ties in favor of the last type we have picked). Indeed, let $i_k, i_{k+1} \notin S$ then

$$\begin{aligned}
\mu(S \cup \{i_k\}) - \mu(S) &= \mu(S \cup \{i_{k+1}\}) - \mu(S) \\
&= \frac{1}{\prod_{j=1}^{n} c_j} \sum_{\{\pi | i_{k+1} \in \pi\}} v\left(S \cup \{i_{k+1}\} \cdot \pi\right) - v\left(S \cdot \pi\right) \\
&= \frac{1}{\prod_{j=1}^{n} c_j} \sum_{\{\pi | i_{k+1} \in \pi\}} v\left(S \cup \{i_k, i_{k+1}\} \cdot \pi\right) - v\left(S \cup \{i_k\} \cdot \pi\right) \\
&= \mu(S \cup \{i_k, i_{k+1}\}) - \mu(S \cup \{i_k\})
\end{aligned}$$

Therefore, the marginal contribution of the type i does not decrease if we include units of type i in the solution. On the other hand, because μ is submodular, the marginal contribution of any other type cannot increase. So we will take unit i_{k+1} right after i_k.

Assume we already have picked set S and now are picking the first unit of a type i. Hence, S comprises all units of a type set T. Then we have

$$\begin{aligned}
\mu\left(S \cup \{i_1\}\right) - \mu\left(S\right) &= \frac{1}{\prod_{k=1}^{n} c_k} \sum_{\{\pi | i_1 \in \pi\}} v\left(S \cup \{i_1\} \cdot \pi\right) - v\left(S \cdot \pi\right) \\
&= \frac{\prod_{k \neq i} c_k}{\prod_{k=1}^{n} c_k} m_T(i) = \frac{m_T(i)}{c_i}
\end{aligned}$$

Thus $i = \operatorname{argmax}_{i \notin T} \frac{m_T(i)}{c_i}$ which coincides with the rule of our fractional greedy scheme.

In case of c_i being real costs the same approach can be applied, although in a more tedious way. □

Now we are ready to analyze the approximation ratio of the mechanism Random-SM. Let $S = \{1, \ldots, k\}$ be the subset returned by Greedy-SM$(A, \frac{B}{2})$. For any $j = k+1, \ldots, \ell$, we have

$$\frac{c_j}{m_j} \geq \frac{c_{k+1}}{m_{k+1}} > \frac{B}{2\sum_{i=1}^{k+1} m_i},$$

where the last inequality follows from the fact that the greedy strategy stops at item $k+1$. Hence, we have $c_j > B \cdot \frac{m_j}{2\sum_{i=1}^{k+1} m_i}$. The same analysis shows that $c'_{\ell+1} > B \cdot \frac{m'_{\ell+1}}{2\sum_{i=1}^{k+1} m_i}$. Therefore,

$$B \cdot \frac{\sum_{j=k+1}^{\ell} m_j + m'_{\ell+1}}{2\sum_{i=1}^{k+1} m_i} < \sum_{j=k+1}^{\ell} c_j + c'_{\ell+1} \leq B.$$

This implies that $2\sum_{i=1}^{k+1} m_i > \sum_{j=k+1}^{\ell} m_j + m'_{\ell+1}$ and $m_{k+1} + 2\sum_{i=1}^{k} m_i > \sum_{j=k+2}^{\ell} m_j + m'_{\ell+1}$. Hence,

$$\begin{aligned}
f_{\mathrm{gre}}(A) &= \sum_{i=1}^{\ell} m_i + m'_{\ell+1} = \sum_{i=1}^{k+1} m_i + \sum_{j=k+2}^{\ell} m_j + m'_{\ell+1} \\
&< 3\sum_{i \in S} m_i + 2m_{k+1} \leq 3\sum_{i \in S} m_i + 2v(i^*).
\end{aligned}$$

Together with Lemma 10, it bounds the optimal solution as

$$opt(A) \leq \frac{e}{e-1}\Big(3\mathsf{Greedy\text{-}SM}(A, B/2) + 2v(i^*)\Big). \tag{4.4}$$

The expected value of our randomized mechanism is $\frac{3}{5}\mathsf{Greedy\text{-}SM}(A, B/2) + \frac{2}{5}v(i^*) \geq \frac{e-1}{5e}opt$. □

4.5.2 Deterministic Mechanism.

Here we provide a deterministic truthful mechanism which is budget feasible and has a constant approximation ratio. In the following description $opt(A \setminus \{i^*\}, B)$ denotes the value of the optimal solution of the weighted submodular maximization problem for the set of agents $A \setminus \{i^*\}$ and budget B.

> Deterministic-SM
>
> 1. Let $A = \{i \mid c_i \leq B\}$ and $i^* \in \operatorname{argmax}_{i \in A} v(i)$
> 2. If $\frac{1+4e+\sqrt{1+24e^2}}{2(e-1)} \cdot v(i^*) \geq opt(A \setminus \{i^*\}, B)$, return i^*
> 3. Otherwise, return $\mathsf{Greedy\text{-}SM}(A, B/2)$

This deterministic mechanism does not work in polynomial time because of the computational hardness of submodular maximization problem. We note however that one cannot replace optimal solution by a greedy solution in the above mechanism as it may break monotonicity. Indeed, unlike the case of additive valuations a seller may change his bid and influence the outcome of a greedy algorithm, while remaining in the winning set. In contrast to frugality model for vertex cover set systems, where we have an efficient greedy algorithm with constant approximation ratio and monotone selection rule, we don't know if such a mechanism exists in the case of general submodular valuations.

Theorem 16. Deterministic-SM *is a deterministic budget feasible truthful mechanism for monotone submodular functions with an approximation ratio of* (≈ 8.34) $\frac{6e-1+\sqrt{1+24e^2}}{2(e-1)}$.

Proof. We note that the bid of i^* is independent of the value of $opt(A \setminus \{i^*\}, B)$. The similar argument as in the proof of Theorem 11 gives us truthfulness.

Budget feasibility follows from Lemma 9 and observation that Step 2 only sets additional thresholds on the payments of $\mathsf{Greedy\text{-}SM}(A, B/2)$.

In what follows, we prove the approximate ratio. Let

$$x = \frac{1 + 4e + \sqrt{1 + 24e^2}}{2(e-1)} (\approx 7.34).$$

We observe that

$$opt(A, B) - v(i^*) \leq opt(A \setminus \{i^*\}, B) \leq opt(A, B).$$

If the condition in Step 2 holds and the mechanism outputs i^*, then

$$opt(A, B) \leq opt(A \setminus \{i^*\}, B) + v(i^*) \leq (x + 1) \cdot v(i^*).$$

In the other case we run $\mathsf{Greedy\text{-}SM}(A, B/2)$. We apply (4.4) and get

$$\begin{aligned} x \cdot v(i^*) &< opt(A \setminus \{i^*\}, B) \leq opt(A, B) \\ &\leq \frac{e}{e-1}\Big(3\mathsf{Greedy\text{-}SM}(A, B/2) + 2v(i^*)\Big). \end{aligned}$$

This implies that

$$v(i^*) \leq \frac{3e}{x(e-1) - 2e}\mathsf{Greedy\text{-}SM}(A, B/2).$$

Hence,

$$\begin{aligned} opt &\leq \frac{e}{e-1}\Big(3\mathsf{Greedy\text{-}SM}(A, B/2) + 2v(i^*)\Big) \\ &\leq \frac{e}{e-1}\left(3 + \frac{6e}{x(e-1) - 2e}\right) \cdot \mathsf{Greedy\text{-}SM}(A, B/2). \end{aligned}$$

Simple calculations show that

$$\begin{aligned} 1 + x &= \frac{6e - 1 + \sqrt{1 + 24e^2}}{2(e-1)} \\ &= \frac{e}{e-1}\left(3 + \frac{6e}{x(e-1) - 2e}\right). \end{aligned}$$

Therefore, we have $opt \leq (x+1) \cdot \mathsf{Greedy\text{-}SM}(A, B/2)$ in the both cases, which concludes the proof of the claimed approximation ratio. □

4.6 XOS Valuations

We recall the definition of XOS valuation $v(\cdot)$: for any $S \subseteq A$

$$v(S) = \max\{f_1(S), f_2(S), \ldots, f_m(S)\},$$

where each $f_j(\cdot)$ is a nonnegative additive function, i.e., $f_j(S) = \sum_{i \in S} f_j(i)$.

We assume w.l.o.g. that $c_i \le B$ for any agent $i \in A$, since if $c_i > B$, then such an agent would never win in any budget feasible mechanism.

Below we establish the main component of our main XOS mechanism. We use randomized Additive-mechanism for additive valuations as an auxiliary procedure, where Additive-mechanism is an universally truthful mechanism Random-KS from Section 4.3 with an approximation factor of at most 3.

<u>XOS-random-sample</u>

1. Sample items independently at random with probability $\frac{1}{2}$ in group T.
2. Find optimal solution $\mathrm{opt}(T)$ for item set T and budget B.
3. Set a threshold $t = \frac{v(\mathrm{opt}(T))}{8B}$.
4. For items in $A \setminus T$ find a set $S^* \in \underset{S \subseteq A \setminus T}{\mathrm{argmax}} \big\{v(S) - t \cdot c(S)\big\}$.
5. Find additive $f \in \{f_1, \ldots, f_m\}$, s.t. $f(S^*) = v(S^*)$.
6. Run Additive-mechanism for $f(\cdot)$, item set S^*, budget B.
7. Output the result of Additive-mechanism.

In the above mechanism, we first sample half in expectation of the items to form a testing group T, and compute an optimal solution for T, given budget constraint B. In Lemma 11 below, we show $v(\mathrm{opt}(A)) \ge v(\mathrm{opt}(T)) \ge \Omega(v(\mathrm{opt}(A)))$ and $v(\mathrm{opt}(A \setminus T)) \ge \Omega(v(\mathrm{opt}(A)))$ with probability at least $\frac{1}{2}$, if all the values of single items in A are comparatively small w.r.t. $v(A)$. In other words, in the above mechanism we are able to learn the rough value of the optimal solution from a random sample, and still keep a nearly optimal solution formed by the remaining items. We then use the information from the sample to compute a proper threshold t for the rest of the items. Specifically, we find a subset $S^* \subseteq A \setminus T$ with the largest difference between its value and cost, multiplied by the threshold t (in the computation of S^*, if there are multiple choices, we break ties in arbitrary but given and fixed order). Finally, we use the property of XOS functions to find a additive representation of $v(S^*)$. Then we run truthful mechanism on the set S^* with respect to this additive valuation function.

We now establish the following technical lemma, which is useful in the analysis of our mechanisms for XOS and subadditive valuations.

Lemma 11. *For any subadditive function $v(\cdot)$ and for any given subset $S \subseteq A$ and a positive integer k, assume that $v(S) \geq k \cdot v(i)$ for any $i \in S$. Further, suppose that S is divided uniformly at random into two groups T_1 and T_2. Then, with probability at least $\frac{1}{2}$, we have $v(T_1) \geq \frac{k-1}{4k} v(S)$ and $v(T_2) \geq \frac{k-1}{4k} v(S)$.*

Proof. We first claim that there are disjoint subsets S_1 and S_2 with $S_1 \cup S_2 = S$ such that $v(S_1) \geq \frac{k-1}{2k} v(S)$ and $v(S_2) \geq \frac{k-1}{2k} v(S)$. This can be seen by the following recursive process: initially let $S_1 = \emptyset$ and $S_2 = S$; and we move items from S_2 to S_1 arbitrarily until the point when $v(S_1) \geq \frac{k-1}{2k} v(S)$. Consider the S_1, S_2 at the end of the process; we claim that at this point, we also have $v(S_2) \geq \frac{k-1}{2k} v(S)$. Note that $v(S) \leq v(S_1) + v(S_2)$. Let i be the last item moved from S_2 to S_1; therefore, $v(S_1 \setminus \{i\}) < \frac{k-1}{2k} v(S)$, which implies that $v(S_2 \cup \{i\}) > \frac{k+1}{2k} v(S)$. Thus, $v(S_2) + v(i) \geq v(S_2 \cup \{i\}) > \frac{k+1}{2k} v(S)$. As $v(i) \leq \frac{1}{k} v(S)$, we know that $v(S_2) > \frac{1}{2} v(S) > \frac{k-1}{2k} v(S)$.

Consider sets $X_1 = S_1 \cap T_1$, $Y_1 = S_1 \cap T_2$, $X_2 = S_2 \cap T_1$ and $Y_2 = S_2 \cap T_2$. Due to subadditivity we have $\frac{k-1}{2k} v(S) \leq v(S_1) \leq v(X_1) + v(Y_1)$; hence, either $v(X_1) \geq \frac{k-1}{4k} v(S)$ or $v(Y_1) \geq \frac{k-1}{4k} v(S)$. Similarly, we have that either $v(X_2) \geq \frac{k-1}{4k} v(S)$ or $v(Y_2) \geq \frac{k-1}{4k} v(S)$. Clearly, partitioning S_1 into X_1, Y_1 and partitioning S_2 into X_2, Y_2 are independent of each other. Therefore, with probability $\frac{1}{2}$ the most valuable parts of S_1's partition and S_2's partition will get into different sets T_1 and T_2, respectively. Thus the lemma follows. $\square$

The XOS-random-sample applicable to XOS functions only, although we use it as an auxiliary procedure for the more general subadditive functions in the next section.

We remark that the running time of the mechanism as described above is exponential in n. However, with an additional so called XOS oracle tailored to the complex XOS valuation function $v(\cdot)$ one can implement the mechanism in polynomial time. More detailed explanation is in order below.

- In the Step 2 of the mechanism, we can use any constant approximation poly-time solution (e.g., algorithm SA-alg-max established in Section 4.7.2), which suffices for our purpose.
- Step 4 can be done by asking a demand query.
- Step 5 is referred in the literature as the XOS oracle, which for any subset X of items returns a additive function f with $f(X) = v(X)$ and $f(S) \leq v(S)$ for each $S \subset X$.
- For some classic XOS problems like matching (the value of a subset of edges is the size of the largest matching induced by them), the XOS and demand oracles can be simulated by efficient algorithms. Therefore, our mechanism can be implemented in polynomial time.

Note that in Step 4, the function $v(S) - t \cdot c(S)$ we are maximizing is simply the Lagrangian function $v(S) - x \cdot c(S) + x \cdot B$ (where $x \cdot B$ is a fixed constant) of the original optimization problem $\max_S v(S)$ subject to $c(S) \leq B$. While we do not know the actual value of the variable x in the Lagrangian, a carefully chosen parameter t in the sampling step with a high probability ensures a constant factor approximation of $\max_S \{v(S) - t \cdot c(S) + t \cdot B\}$ to the actual value of the Lagrangian

$\max_S \big\{v(S) - x \cdot c(S) + x \cdot B\big\}$ and related optimum value $v(\mathrm{opt}(A))$ of the original problem $\max_S v(S)$ subjected to $c(S) \le B$.

The linearity of the Lagrangian, together with the subadditivity of $v(\cdot)$, is important for us in order to derive the following observations on the threshold t, subset S^*, and additive function f that were defined in the XOS-random-sample.

Claim 2. *For any $S \subseteq S^*$, $f(S) - t \cdot c(S) \ge 0$.*

Proof. Assume by contradiction that there exists a subset $S \subseteq S^*$ such that $f(S) - t \cdot c(S) < 0$. Let $S' = S^* \setminus S$. Since f is an additive function, we have $c(S') + c(S) = c(S^*)$ and $f(S') + f(S) = f(S' \cup S) = f(S^*) = v(S^*)$. Thus,

$$\begin{aligned}
v(S') - t \cdot c(S') &\ge f(S') - t \cdot c(S') \\
&= v(S^*) - t \cdot c(S^*) - \big(f(S) - t \cdot c(S)\big) \\
&> v(S^*) - t \cdot c(S^*),
\end{aligned}$$

which contradicts the definition of S^*. □

The following claim states that any item in S^* cannot change S^* by bidding smaller than his true cost. This fact is crucial for monotonicity and truthfulness of the mechanism.

Claim 3. *If any item $j \in S^*$ reports a smaller cost $b(j) < c(j)$, then set S^* remains the same.*

Proof. Let b be the bid vector where j reports $b(j)$ and other bids remain unchanged. First we notice that for any set S with $j \in S$, $\big(v(S) - t \cdot b(S)\big) - \big(v(S) - t \cdot c(S)\big) = t\big(c(j) - b(j)\big)$ is a fixed positive value. Hence,

$$\begin{aligned}
v(S^*) - t \cdot b(S^*) &= v(S^*) - t \cdot c(S^*) + t\big(c(j) - b(j)\big) \\
&\ge v(S) - t \cdot c(S) + t\big(c(j) - b(j)\big) \\
&= v(S) - t \cdot b(S).
\end{aligned}$$

Further, for any set S with $j \notin S$, we have

$$\begin{aligned} v(S^*) - t \cdot b(S^*) &> v(S^*) - t \cdot c(S^*) \\ &\geq v(S) - t \cdot c(S) \\ &= v(S) - t \cdot b(S). \end{aligned}$$

Therefore, we conclude that $S^* = \underset{S \subseteq A \setminus T}{\operatorname{argmax}} \big(v(S) - t \cdot b(S)\big)$; and by the fixed tie-breaking rule, S^* is selected as well. □

Our main mechanism for XOS functions is simply a uniform distribution of the mechanism XOS-random-sample and one that picks the most valuable item.

<u>XOS-mechanism-main</u>

- `With probability` 0.5 `run` XOS-random-sample.
- `With probability` 0.5 `pick item from` $\operatorname{argmax}_i v(i)$ `and pay` B.

Theorem 17. *The mechanism* XOS-mechanism-main *is budget feasible and truthful, and provides a constant approximation ratio for XOS valuation functions.*

Proof. The proof follows from the following three lemmas.

Lemma 12. XOS-mechanism-main *is universally truthful.*

Proof. At a high level point of view, our mechanism is built on the idea introduced in [1] of composing a few different selection rules. In particular, the Steps (1-4) comprise the first round of selection for winning agents and the Steps (5-7) further sift out survived agents in the second round. It was shown in [1] that if the first selection rule is composable (i.e., monotone and in addition to that any winner cannot manipulate the winning set without losing) and the second selection rule is just monotone, then their composition is monotone. In our case, Claim 3 implies the composability of the Steps (1-4) and the monotonicity of the Steps (5-7) simply follows from the truthfulness of Additive-mechanism. We note that in a single parameter domain monotone allocation rule with threshold payments implies truthfulness. Therefore, our mechanism is truthful. □

Lemma 13. XOS-mechanism-main *is budget feasible.*

Proof. It suffices to prove that both mechanisms are budget feasible. Clearly, picking the largest item is budget feasible. XOS-random-sample uses a budget feasible mechanism Additive-mechanism to decide the final winning set and payments. Therefore, the threshold payments in XOS-random-sample (the minimum over all middle steps) can be only smaller than those in Additive-mechanism; this implies that that XOS-random-sample is budget feasible as well. □

Lemma 14. XOS-mechanism-main *has a constant approximation ratio.*

Proof. Let $\mathrm{opt} = \mathrm{opt}(A)$ denote the optimal winning set given budget B, and let $k = \min_{i \in \mathrm{opt}} \frac{v(\mathrm{opt})}{v(i)}$. Thus $v(\mathrm{opt}) \geq k \cdot v(i)$ for each $i \in \mathrm{opt}$. By Lemma 11, we have $v(\mathrm{opt} \cap T) \geq \frac{k-1}{4k} v(\mathrm{opt})$ with probability at least $\frac{1}{2}$. Thus, we have $v(\mathrm{opt}(T)) \geq v(\mathrm{opt} \cap T) \geq \frac{k-1}{4k} v(\mathrm{opt})$ with probability at least $\frac{1}{2}$ (the first inequality holds $\mathrm{opt} \cap T$ is a particular solution and $\mathrm{opt}(T)$ is an optimal solution for set T with budget constraint).

We let $\mathrm{opt}^* = \mathrm{opt}_f(S^*)$ be the optimal solution with respect to the item set S^*, additive value-function f and budget B. Below we show that $f(\mathrm{opt}^*)$ is a good approximation to the actual social optimum $v(\mathrm{opt})$. Consider the following two cases:

- $c(S^*) > B$. With such assumption, we can always find a subset $S' \subseteq S^*$, such that $\frac{B}{2} \leq c(S') \leq B$. By Claim 2, we know $f(S') \geq t \cdot c(S') \geq \frac{v(\mathrm{opt}(T))}{8B} \cdot \frac{B}{2} \geq \frac{v(\mathrm{opt}(T))}{16}$. Then by the fact that opt^* is an optimal solution and S' is a particular solution with budget constraint B, we have $f(\mathrm{opt}^*) \geq f(S') \geq \frac{v(\mathrm{opt}(T))}{16} \geq \frac{k-1}{64k} v(\mathrm{opt})$ with probability at least $\frac{1}{2}$.

- $c(S^*) \leq B$. Then $\mathrm{opt}^* = S^*$. Let $S' = \mathrm{opt} \setminus T$; thus, $c(S') \leq c(\mathrm{opt}) \leq B$. By Lemma 11, we have $v(S') \geq \frac{k-1}{4k} v(\mathrm{opt})$ with probability at least $\frac{1}{2}$. Recall that $S^* = \operatorname*{argmax}_{S \subseteq A \setminus T}(v(S) - t \cdot c(S))$. Then with probability at least $\frac{1}{2}$, we have

$$\begin{aligned} f(\text{opt}^*) = f(S^*) &= v(S^*) \\ &\geq v(S^*) - t \cdot c(S^*) \\ &\geq v(S') - t \cdot c(S') \\ &\geq \frac{k-1}{4k} v(\text{opt}) - \frac{v(\text{opt}(T))}{8B} \cdot B \\ &\geq \frac{k-1}{4k} v(\text{opt}) - \frac{v(\text{opt})}{8} \\ &= \frac{k-2}{8k} v(\text{opt}). \end{aligned}$$

In either case, we get

$$f(\text{opt}^*) \geq \min\left\{\frac{k-1}{64k} v(\text{opt}), \frac{k-2}{8k} v(\text{opt})\right\} \geq \frac{k-2}{64k} v(\text{opt})$$

with probability at least $\frac{1}{2}$. At the end we output the result of Additive-mechanism(f, S^*, B) in the last step of XOS-random-sample. We recall that Additive-mechanism has approximation factor of at most 3 with respect to the optimal solution $f(\text{opt}^*)$. Thus the value of the solution given by XOS-random-sample is at least $\frac{1}{3} \cdot f(\text{opt}^*) \geq \frac{1}{3} \cdot \frac{1}{2} \cdot \frac{k-2}{64k} v(\text{opt}) = \frac{k-2}{384k} v(\text{opt})$.

On the other hand, since $k = \min_{i \in \text{opt}} \frac{v(\text{opt})}{v(i)}$, the solution given by picking the largest item satisfies $\max_i v(i) \geq \frac{1}{k} v(\text{opt})$. Combining the two mechanisms together, our main mechanism XOS-mechanism-main has performance at least

$$\left(\frac{1}{2} \cdot \frac{k-2}{384k} + \frac{1}{2} \cdot \frac{1}{k}\right) v(\text{opt}) = \frac{k+382}{768k} v(\text{opt}) \geq \frac{1}{768} v(\text{opt}).$$

□

This completes the proof of the theorem. □

4.7 Subadditive Valuations

We first apply the following straightforward approach, where one simply substitutes a given general subadditive valuation by its approximation with XOS function. Noticeably, that change in the objective valuation does not cause any problems with agents' incentives and thus can be addressed from purely approximation point of view.

4.7.1 Mechanism via Reduction to XOS

XOS valuations may be equivalently characterized as the class of fractionally subadditive functions, as was shown in [34]. Let $S_1, \ldots, S_N$ be the tuple of all possible subsets of A, where $N = |2^A|$ is the size of the power set 2^A. In particular, we may consider the following linear program LP(S) for a subset $S \subseteq A$, where every subset S_j is associated with a variable α_j.

$$
\begin{array}{lll}
LP(S): & \min & \sum_{j=1}^{N} \alpha_j \cdot v(S_j) \qquad\qquad (\Diamond) \\
& s.t. & \alpha_j \geq 0, \quad 1 \leq j \leq N \\
& & \sum_{j:\ i \in S_j} \alpha_j \geq 1, \quad \forall\, i \in S
\end{array}
$$

In $(\Diamond)$ the minimum is taken over all possible non-negative values of $\alpha = (\alpha_1, \ldots, \alpha_N)$. If we consider each α_j as the fraction of the cover by subset S_j, the last constraint implies that all items in S are fractionally covered. Hence, LP(S) describes a linear program for the set cover of S. By definition, any fractionally subadditive function $v(\cdot)$ must obey the inequality $v(A') \leq LP(A')$ for each $A' \subset A$.

The above LP has a strong connection to the core of cost sharing games (considering $v(\cdot)$ instead as a cost function), which is a central notion in cooperative game theory [60]. Roughly speaking, the core of a game is a stable cooperation among all agents to share $v(A)$ where no subset of agents can benefit by breaking away from the grand coalition. It is well known that the cores of many cost sharing games are empty. This motivates the notion of α-approximate core, which requires all the agents to share only an α-fraction of $v(A)$. Bondareva-Shapley Theorem [13, 62] says that for subadditive functions, the largest value α for which the α-approximate core is nonempty is equal to the integrality gap of the LP. Further, the integrality gap of the LP equals to one (i.e., $v(A)$ is also an optimal fractional solution) if and only if the valuation function is XOS, which is also equivalent to the non-emptiness of the core.

For any subadditive function $v(\cdot)$, it can be seen that the value of the optimal integral solution to the above LP(S) is always $v(S)$. Indeed, one has $S \subseteq \bigcup_{j\colon \alpha_j \geq 1} S_j$ and $\sum_j \alpha_j \cdot v(S_j) \geq \sum_{j\colon \alpha_j \geq 1} v(S_j) \geq v\big(\bigcup_{j\colon \alpha_j \geq 1} S_j\big) \geq v(S)$.

Definition 6. *Let $\widetilde{v}(S)$ be the value of the optimal fractional solution to LP(S), and $\mathcal{I}(S) = {}^{v(S)}/_{\widetilde{v}(S)}$ be the integrality gap of LP(S). Let $\mathcal{I} = \max_{S \subseteq A} \mathcal{I}(S)$.*

The integrality gap $\mathcal{I}$ gives a worst-case upper bound on the integrality gap of all subsets. Hence, we have $\frac{v(S)}{\mathcal{I}} \leq \widetilde{v}(S) \leq v(S)$ for any $S \subseteq A$.

The classic Bondareva-Shapley Theorem [13, 62] says that the integrality gap $\mathcal{I}(S)$ is one (i.e., $v(S)$ is also an optimal fractional solution to the LP) if and only if $v(\cdot)$ is an XOS function.

Lemma 15. *$\widetilde{v}(\cdot)$ is an XOS function.*

Proof. For any subset $S \subseteq A$, consider any non-negative vector $\gamma = (\gamma_1, \ldots, \gamma_N) \geq 0$ that satisfies $\sum_{j\colon i \in S_j} \gamma_j \geq 1$ for any $i \in S$. Then we have

$$\begin{aligned}
\sum_{j=1}^{N} \gamma_j \cdot \widetilde{v}(S_j) &= \sum_{j=1}^{N} \gamma_j \cdot \min_{\beta_{j,\cdot} \geq 0} \left(\sum_{k=1}^{N} \beta_{j,k} \cdot v(S_k) \;\middle|\; \forall\, i \in S_j,\ \sum_{k\colon i \in S_k} \beta_{j,k} \geq 1 \right) \\
&= \min_{\beta \geq 0} \left(\sum_{j=1}^{N} \gamma_j \sum_{k=1}^{N} \beta_{j,k} \cdot v(S_k) \;\middle|\; \forall\, j, \forall\, i \in S_j,\ \sum_{k\colon i \in S_k} \beta_{j,k} \geq 1 \right) \\
&= \min_{\beta \geq 0} \left(\sum_{k=1}^{N} \left(\sum_{j=1}^{N} \gamma_j \beta_{j,k} \right) \cdot v(S_k) \;\middle|\; \forall\, j, \forall\, i \in S_j,\ \sum_{k\colon i \in S_k} \beta_{j,k} \geq 1 \right) \\
&\geq \min_{\alpha \geq 0} \left(\sum_{k=1}^{N} \alpha_k \cdot v(S_k) \;\middle|\; \forall\, i \in S,\ \sum_{k\colon i \in S_k} \alpha_k \geq 1 \right) \\
&= \widetilde{v}(S).
\end{aligned}$$

The inequality above follows from the fact that for any $i \in S$,

$$\begin{aligned}\sum_{k:\ i\in S_k}\sum_j \gamma_j\beta_{j,k} &= \sum_j \gamma_j \sum_{k:\ i\in S_k} \beta_{j,k} \\ &\geq \sum_j \gamma_j \geq \sum_{j:\ i\in S_j} \gamma_j \geq 1.\end{aligned}$$

Hence, $\widetilde{v}(\cdot)$ is fractionally subadditive, which is equivalent to XOS. □

We are now ready to present our first mechanism for subadditive functions.

<u>SA-mechanism-main</u>

1. For each subset $S \subseteq A$, compute $\widetilde{v}(S)$.
2. Run XOS-mechanism-main on the same instance w.r.t. $\widetilde{v}(\cdot)$.
3. Output the result of XOS-mechanism-main.

Theorem 18. *The mechanism* SA-mechanism-main *is budget feasible, truthful, and provides an approximation ratio of $O(\mathcal{I})$ for subadditive functions, where $\mathcal{I}$ is the largest integrality gap of LP(S) for all $S \subset A$.*

Proof. Note that the valuation $v(\cdot)$ is a public knowledge and utilities of agents do not depend on $v(\cdot)$; thus computing $\widetilde{v}(\cdot)$ and running XOS-mechanism-main with respect to $\widetilde{v}(\cdot)$ do not affect truthfulness. The claim then follows from Theorem 17 and the fact that $\frac{v(S)}{\mathcal{I}} \leq \widetilde{v}(S) \leq v(S)$ for any $S \subseteq A$ (i.e., using $\widetilde{v}(\cdot)$ instead of $v(\cdot)$ we lose at most factor of $\mathcal{I}$ in the approximation ratio). □

In general, the approximation ratio of the mechanism can be as large as $\Theta(\log n)$ [28, 12]. But for those instances when the integrality gap of ($\Diamond$) is bounded by a constant (e.g., facility location [60]), our mechanism gives a constant approximation.

4.7.2 Sub-Logarithmic Approximation

Subadditive Maximization with Budget We first give an algorithm that approximates $\max_{S\subseteq A} v(S)$ given that $c(S) \leq B$. That is, we ignore for a while strategic behaviors of the agents and consider a pure optimization problem. Dobzinski et al. [30] considered the same question and gave a 4-approximation algorithm for the unweighted case (i.e., the restriction is on the size of selected subset). Our next algorithm extends this result to the weighted case and runs in polynomial time using the demand oracle. Later, Badanidiyuru et al. [6] gave a $2 + \epsilon$ approximation polynomial time algorithm to the same weighted problem that also exploits demand oracle. To make the exposition of this thesis complete we present below our algorithm, although we admit that in terms of the approximation ratio it is better to use the algorithm from [6].

<u>SA-alg-max</u>

- `Let` $v^* = \max_{i\in A} v(i)$ `and` $\mathcal{V} = \{v^*, 2v^*, \ldots, nv^*\}$
- `For each` $v \in \mathcal{V}$
  - $\forall i \in A$ `set` $p(i) = \frac{v\cdot c(i)}{2B}$`, find` $T \in \underset{S\subseteq A}{\operatorname{argmax}} \left(v(S) - \sum_{i\in S} p(i)\right)$.
  - `Let` $S_v = \emptyset$.
  - `If` $v(T) < \frac{v}{2}$`, then continue to next` v.
  - `Else, in decreasing order of` $c(i)$ `put items from` T `in` S_v `while preserving budget constraint.`
- `Output:` S_v `with the largest value` $v(S_v)$ `for all` $v \in \mathcal{V}$.

Lemma 16. SA-alg-max *is an* 8*-approximation algorithm for subadditive maximization with budget working in polynomial time with a demand oracle.*

Proof. Let S^* be an optimal solution. Note that $v(S^*) \geq v^* = \max_{i\in A} v(i)$ and $c(S^*) \leq B$. For all $v \leq v(S^*)$, we first prove that the algorithm will generate a non-empty set S_v with $v(S_v) \geq \frac{v}{4}$. Since T is the maximum set returned by the oracle, we have

$$v(T) - \frac{v}{2B}c(T) \geq v(S^*) - \frac{v}{2B}c(S^*) \geq v - \frac{v}{2B} \cdot B \geq \frac{v}{2}.$$

Hence, $v(T) \geq \frac{v}{2}$. If $c(T) \leq B$, then $S_v = T$ and we are done. Otherwise, by the greedy procedure of picking items from T to S_v, we are guaranteed that $c(S_v) \geq \frac{B}{2}$. Assume for contradiction that $v(S_v) < \frac{v}{4}$. Then

$$\begin{aligned} v(T \setminus S_v) - \frac{v}{2B}c(T \setminus S_v) &\geq v(T) - v(S_v) - \frac{v}{2B}\big(c(T) - c(S_v)\big) \\ &> v(T) - \frac{v}{4} - \frac{v}{2B}c(T) + \frac{v}{2B} \cdot \frac{B}{2} \\ &= v(T) - \frac{v}{2B}c(T). \end{aligned}$$

The latter contradicts the definition of T, since $T \setminus S_v$ is then better than T. Thus, we always have $v(S_v) \geq \frac{v}{4}$ for each $v \leq v(S^*)$. Since the algorithm tries all possible $v \in \mathcal{V}$ (including one with $\frac{v(S^*)}{2} < v \leq v(S^*)$) and outputs the largest $v(S_v)$, the output is guaranteed to be within a factor of 8 to the optimal value $v(S^*)$. □

Remark 2. *Note that we can actually modify the algorithm to get a $(4+\epsilon)$-approximation with running time polynomial in n and $\frac{1}{\epsilon}$. To do so one may simply replace $\mathcal{V}$ by a larger set $\left\{\epsilon v^*, 2\epsilon v^*, \ldots, \lceil \frac{n}{\epsilon} \rceil \epsilon v^*\right\}$. Both algorithms suffice for our purpose; for the rest of the paper, for simplicity we will use the 8-approximation algorithm, to avoid the extra parameter ϵ in the description.*

We will use SA-alg-max further as a subroutine in the mechanism SA-random-sample. When there are different sets maximizing $v(S) - \sum_{i \in S} p(i)$, we require that the demand query oracle always returns a fixed set T. This property is important for truthfulness of our mechanism. To implement this, we set some fixed order on the items $i_1 \prec i_2 \prec \cdots \prec i_n$. We first compute $T_1 \in \text{argmax}_{S \subseteq A}\left(v(S) - \sum\limits_{i \in S} p(i)\right)$ and $T_2 \in \text{argmax}_{S \subseteq A \setminus \{i_1\}}\left(v(S) - \sum\limits_{i \in S} p(i)\right)$. If $v(T_1) - \sum_{i \in T_1} p(i) = v(T_2) - \sum_{i \in T_2} p(i)$, we know that there is a subset without i_1

that gives us the maximum; thus, we ignore i_1 for further consideration. If $v(T_1) - \sum_{i \in T_1} p(i) > v(T_2) - \sum_{i \in T_2} p(i)$, we know that i_1 should be in any optimal solution; hence, we keep i_1 and proceed the process iteratively for $i_2, i_3, \ldots, i_n$. This process clearly gives a fixed outcome that maximizes $v(S) - \sum_{i \in S} p(i)$.

We present now another mechanism for subadditive valuation based on the ideas of random sampling and cost sharing. In contrast to the previous SA-mechanism-main this procedure runs in polynomial time and has an $o(\log n)$ approximation ratio, which improves upon previously best known ratio [30] of $O(\log^2 n)$.

Let us first consider the following mechanism, based on random sampling and cost sharing.

SA-random-sample

1. Sample items independently at random with probability $\frac{1}{2}$ in group T.
2. Run SA-alg-max on the item set T, let v be the value of the returned set.
3. For $k = 1$ to $|A \setminus T|$
   - Run SA-alg-max on the item set $\left\{ i \in A \setminus T \mid c(i) \leq \frac{B}{k} \right\}$, where each item has cost $\frac{B}{k}$, denote output by X.
   - If $v(X) \geq \frac{\log \log n}{80 \log n} \cdot v$
     - Output X, pay $\frac{B}{k}$ to each item in X.
     - Halt.
4. Output $\emptyset$.

In the above mechanism, we again first sample half in expectation of the items to form a testing group T, and then use SA-alg-max to compute an approximate solution for the items in T given the budget constraint B. As it can be seen in the analysis of the mechanism, the computed value v is (in expectation) within a constant factor of the optimal value of the whole set A. That is, we are able to learn the rough value of the optimal solution by random sampling. Next we consider the remaining items $A \setminus T$ and try to find a subset X with a relatively big value in which every item is willing to "share" the budget B at a fixed share $\frac{B}{k}$. (This part of our mechanism can be viewed as a reversion of the classic cost sharing mechanism.) Finally, we use the information v from the sample as a benchmark to determine whether X should be a winning set or not.

Finally, the mechanism for subadditive functions is as follows.

SA-mechanism-main-2

- `With probability` 0.5, `run` SA-random-sample.
- `With probability` 0.5, `pick item from` $\text{argmax}_i\, v(i)$ `and pay` B

Theorem 19. SA-mechanism-main-2 *runs in polynomial time, given a demand oracle, and it is a truthful budget feasible mechanism for subadditive functions with an approximation ratio of* $O(\frac{\log n}{\log\log n})$.

Proof. Let $S = A \setminus T$. It is obvious that the mechanism runs in polynomial time since SA-alg-max works in polynomial time. When the mechanism picks the largest item, it is certainly budget feasible, as the total payment is precisely B. If it chooses SA-random-sample, either no item is a winner or X is selected as the winning set. Note that $|X| \le k$ and each item in X gets a payment of $\frac{B}{k}$. It is therefore budget feasible as well.

(Truthfulness.) Truthfulness for the part of picking the largest item is obvious, as the outcome is irrelevant to the submitted bids. Next we prove that SA-random-sample is truthful as well. The random sampling step does not depend on the item's bids, and items in T have no incentive to lie as they cannot win anyway. Hence, it suffices to consider only items in S. We observe that every agent becomes a candidate to win only if $c(i) \le \frac{B}{k}$. Consider any item $i \in S$ and

fix bids of other items. If i reports his true cost $c(i)$, there are the following three possibilities.

- Item i wins and is paid $\frac{B}{k}$. Then $c(i) \leq \frac{B}{k}$ and his utility is $\frac{B}{k} - c(i) \geq 0$. If i reports a bid which is still less than or equal to $\frac{B}{k}$, the output and all the payments do not change. If i reports a bid which is larger than $\frac{B}{k}$, he still could not win for a share larger than $\frac{B}{k}$ and will not be considered for all smaller shares. Therefore, he derives 0 utility. Thus for either case, i does not have incentive to lie.

- Item i loses, and the payment to each winner is $\frac{B}{k} \geq c(i)$. In this case, if i reduces or increases his bid, he cannot change the output of the mechanism. Thus i always has zero utility.

- Item i loses, and the payment to each winner is $\frac{B}{k} < c(i)$, or the winning set is empty. In this case, if i reduces his bid, he will not change the process of the mechanism until the payment offered by the mechanism is less than $c(i)$. Thus, even if i could win for some value k, the payment he gets would be less than $c(i)$, in which case his utility is negative. If i increases his bid, he loses and thus derives zero utility.

Therefore, SA-random-sample is a universally truthful mechanism.

(Approximation Ratio.) It remains to estimate the approximation ratio. Let $\mathrm{opt} = \mathrm{opt}(A)$ denote the optimal solution for the whole set.

If there exists an item $i \in A$ such that $v(i) \geq \frac{1}{2}v(\mathrm{opt})$, then picking the largest item will generate the value at least $\frac{1}{2}v(\mathrm{opt})$ and we are done. Below we assume that $v(i) < \frac{1}{2}v(\mathrm{opt})$ for all $i \in A$. Then, by Lemma 11, with probability at least $\frac{1}{2}$ we have $v(\mathrm{opt}(T)) \geq \frac{1}{8}v(\mathrm{opt})$ and $v(\mathrm{opt}(S)) \geq \frac{1}{8}v(\mathrm{opt})$. Hence, with probability at least $\frac{1}{4}$ we have

$$v(\mathrm{opt}(S)) \geq v(\mathrm{opt}(T)) \geq \frac{1}{8}v(\mathrm{opt}). \tag{4.5}$$

Therefore, it suffices to prove that the main mechanism has an approximation ratio of $O(\frac{\log n}{\log\log n})$ given the inequalities (4.5).

Since SA-alg-max is an 8-approximation of $v(\mathrm{opt}(T))$, we have $v \geq \frac{1}{8}v(\mathrm{opt}(T)) \geq \frac{1}{64}v(\mathrm{opt})$. Clearly, if SA-random-sample outputs a non-empty set, then its value is at least $\frac{\log\log n}{80\log n} \cdot v \geq \frac{\log\log n}{5120\log n} \cdot v(\mathrm{opt})$. Hence, it remains to prove that the mechanism will always output a non-empty set given formula (4.5).

Let $S^* = \{1, 2, 3, \ldots, m\} \subseteq S$ be an optimal solution of S, given the budget constraint B and $c_1 \geq c_2 \geq \cdots \geq c_m$. We recursively divide the agents in S^* into different groups as follows:

- Let α_1 be the largest integer such that $c_1 \leq \frac{B}{\alpha_1}$. Put the first $\min\{\alpha_1, m\}$ agents into group Z_1.

- Let $\beta_r = \alpha_1 + \cdots + \alpha_r$. If $\beta_r < m$ let α_{r+1} be the largest integer such that $c_{\beta_r+1} \leq \frac{B}{\alpha_{r+1}}$; put the next $\min\{\alpha_{r+1}, m-\beta_r\}$ agents into group Z_{r+1}.

Let us denote by $x+1$ the number of groups. Since the items in S^* are ordered by $c_1 \geq c_2 \geq \cdots \geq c_m$, we have $\alpha_{r+1} \geq \alpha_r$ for any r. If there exists a set Z_j such that $v(Z_j) \geq \frac{\log\log n}{10\log n} \cdot v$, then the mechanism outputs a non-empty set, as it could buy α_j items at price $\frac{B}{\alpha_j}$ given that SA-alg-max is an 8-approximation and the threshold we set is $v(Z_j) \geq \frac{\log\log n}{80\log n} \cdot v$. Thus, we may assume that $v(Z_j) < \frac{\log\log n}{10\log n} \cdot v$ for each $j = 1, 2, \ldots, x+1$. On the other hand, by subadditivity, we have

$$\sum_{j=1}^{x+1} v(Z_j) \geq v(S^*) = v(\mathrm{opt}(S)) \geq v(\mathrm{opt}(T)) \geq v.$$

Putting the two inequalities together, we can conclude that $(x+1) \cdot \frac{\log\log n}{10\log n} \cdot v > v$, which implies that

$$x > \frac{5\log n}{\log\log n} \geq \frac{5\log m}{\log\log m}.$$

On the other hand, since $S^* = \{1, 2, 3, \ldots, m\}$ is a solution for S within the budget constraint, we have that $\sum_{j=1}^{m} c_j \leq B$. Further, since $c_1 > \frac{B}{\alpha_1+1}, c_{\beta_1+1} >$

$\frac{B}{\alpha_2+1}, \ldots, c_{\beta_x+1} > \frac{B}{\alpha_{x+1}+1}$, we have

$$\begin{aligned} B &\geq \sum_{j=1}^{m} c_j \geq c_1 + \alpha_1 c_{\beta_1+1} + \cdots + \alpha_x c_{\beta_x+1} \\ &> \frac{B}{\alpha_1+1} + \frac{\alpha_1 B}{\alpha_2+1} + \cdots + \frac{\alpha_x B}{\alpha_{x+1}+1}. \end{aligned}$$

Hence,

$$1 \geq \frac{1}{\alpha_1+1} + \frac{\alpha_1}{\alpha_2+1} + \cdots + \frac{\alpha_x}{\alpha_{x+1}+1} \geq \frac{1}{2\alpha_1} + \frac{\alpha_1}{2\alpha_2} + \cdots + \frac{\alpha_x}{2\alpha_{x+1}}.$$

In particular, we get

$$2 \geq \frac{1}{\alpha_1} + \frac{\alpha_1}{\alpha_2} + \cdots + \frac{\alpha_{x-1}}{\alpha_x} \geq x \sqrt[x]{\frac{1}{\alpha_1}\frac{\alpha_1}{\alpha_2}\cdots\frac{\alpha_{x-1}}{\alpha_x}},$$

where the last inequality is simply the inequality of arithmetic and geometric means. Hence, we get $2 \geq x\sqrt[x]{\frac{1}{\alpha_x}}$, which is equivalent to $\alpha_x \geq (\frac{x}{2})^x$. Now plugging in the fact that $m \geq \alpha_x$ and $x \geq \frac{5\log m}{\log\log m}$, we come to a contradiction. This concludes the proof. □

Chapter 5

Budget Feasible Mechanism Design 2: Extensions

In this chapter we introduce some natural extensions of the budget feasible model. First, we analyze the original setup in a more relaxed classical Bayesian framework as opposed to the former worst-case prior-free framework. Second, we suggest a generalization of the basic model to the case, where each agent is allowed to have more than one item for sale.

Bayesian mechanism design. As a standard game theoretic model for incomplete information, Bayesian mechanism design assumes that agents' private information (i.e., $(c(i))_{i \in A}$ in the basic model) is drawn from a known distribution. In contrast to the previous chapter, if we have prior knowledge about the cost's distribution, then we can obtain more positive results in the form of constant approximation truthful mechanisms. It turns out that the question

> *"A fundamental question is whether, regardless of computational constraints, a constant-factor budget feasible mechanism exists for subadditive functions,"*

posed by Dobzinski, Papadimitriou, Singer in [30] has an affirmative answer.

Techniques. The major approach in the design of budget feasible mechanisms for additive and submodular valuations is based on a simple idea of adding agents one by one greedily and carefully ensuring that the budget constraint is not violated. All mechanisms in this thesis for XOS and subadditive valuations, from a high level structural point of view, use a different approach of random sampling.

In the Bayesian setting, random sampling is often deemed to be unnecessary, because, when we have knowledge of the distribution, it is tempting to use a 'prior sampling' approach to generate random virtual instances, and based on them get an estimate for the real instance. While this works well when the private costs $c(i)$ of every agent are drawn independently, interestingly (and surprisingly), it fails when costs $c(i)$'s are correlated in the distribution. We therefore still have to use "real" random sample to get a rough estimate based on the sampled test set; the collected information from random sampling correctly reflects the structure of the private costs (with a high probability) even for correlated distributions.

Random sampling appears to be a powerful tool in mechanism design and has been used successfully in other domains such as digital goods auctions [39], secretary problem [5, 4], social welfare maximization [28], and mechanism design without money [20].

Multi-parameter. The design of mechanisms for agents having a rich multi-parameter bidding language is known to be much harder than for the scenarios where each agent provides only a single real number. Indeed, the restrictions on what auctioneer can choose as an outcome and payments become much more complicated than the threshold rule we were using so far in the scenario with single-item sellers. Generally speaking, in a multi-parameter case there is no such simple characteristic property for a mechanism being incentive compatible as the one of monotonicity in the single parameter domains [52, 3]. Not surprisingly, there are considerably fewer positive results in the multi-parameter settings than in the single-parameter settings. Thus it is quite unexpected to see a truthful mechanism with good performance guarantees in the multi-parameter extension of the original budget feasible model.

5.1 Bayesian Framework

In this section we study budget feasible mechanisms for subadditive functions from a standard economic viewpoint, where the costs of all agents $(c(i))_{i\in A}$ are drawn from given distribution $\mathcal{D}$. More specifically, the mechanism designer and all participants know in advance $\mathcal{D}$, from which the real cost vector $(c(i))_{i\in A}$ is drawn. However, each $c(i)$ is the private information of agent i. Distribution $\mathcal{D}$ is given on the probability space Ω with the corresponding density function $\rho(\cdot)$ on $\mathbb{R}^{|A|}$. We allow dependencies on the agents' costs in $\mathcal{D}$. We need some mild technical restriction on $\mathcal{D}$ in order to sample a conditioned random variable. We assume that the density function $\rho(\cdot)$ of $\mathcal{D}$ is integrable over each subset $S \subseteq A$ of its variables for any choice of the rest parameters, i.e., $\rho(c_{A\setminus S}) = \int_\Omega \rho(c)\, dx_S$ is bounded. This condition is reminiscent of integrability of marginal density functions (see, e.g., page 331 of [65]), though in our case we require a slightly stronger condition. This includes independent distributions as a special case. We note that any distribution with discrete and finite support satisfies this requirement.

Every agent submits a bid $b(i)$ and seeks to maximize his own utility. We again consider universally truthful mechanisms, i.e., for every coin flip of the mechanism and each cost vector, truth-telling is a dominant strategy for every agent.

- The performance of a mechanism $\mathcal{M}$ is measured now in expectation by $\mathbf{E}[\mathcal{M}] = \mathbf{E}_{c\sim\mathcal{D}}[\mathcal{M}(c)]$.
- It is compared to the expected value of the optimal solution, given by $\mathbf{E}[\text{opt}] = \mathbf{E}_{c\sim\mathcal{D}}\big[v(\text{opt}(c))\big]$.
- We say mechanism $\mathcal{M}$ is a (Bayesian) α-approximation if $\frac{\mathbf{E}[\text{opt}]}{\mathbf{E}[\mathcal{M}]} \le \alpha$.

Interestingly, in the budget feasible model one does not need to rely on Bayesian analysis in the following aspects:

Truthfulness. In most of the previous works in Bayesian mechanism design regarding social welfare maximization, e.g. [45, 10, 44, 16], the considered solution concept is Bayesian truthfulness, i.e., truth-telling is in expectation an equilibrium strategy when other agents' profiles are drawn from the known distribution. In budget feasible model it is still possible to achieve *universal truthfulness*, meaning that truth-telling is dominant strategy of each agent for any coin flips of the mechanism and any instance of the costs. Thus universal truthfulness is a stronger solution concept than Bayesian truthfulness. Universal truthfulness in Bayesian mechanism design has also been used in, e.g. [15], but with a different focus on profit maximization.

Distribution. Regarding the prior knowledge of the distribution, most of the previous related works consider independent distributions, e.g. [46, 45, 10, 44]. In the budget feasible model one may still allow general distributions with a correlation of costs. Dependency on private information is a natural phenomenon arising in practice and it has been considered for, e.g., auctions [56]. In our model, although costs are private parameters, correlation among them appears to be common occurrence. For example, if the price on crude oil goes up, the incurred costs of *all* agents for producing their items will go up as well.

In this section we: let $\mathrm{opt}_v(c, S)$ denote the winning set in an optimal solution with the valuation function $v(\cdot)$, the cost vector is c, and the agent set is S (the parameters are omitted if they are clear from the context); let $v(\mathrm{opt}_v(c, S))$ denote the value of $\mathrm{opt}_v(c, S)$.

We present below a Bayesian constant approximation universally truthful mechanism for a general subadditive valuation $v(\cdot)$. We note that this mechanism does not always work in polynomial time.

SA-Bayesian-mechanism

- `With probability` 0.5`, pick item from` $\mathrm{argmax}_i\, v(i)$ `and pay` B.
- `With probability` 0.5`, run the following:`
  1. `Sample items independently at random in group` T.
  2. `Find` $\mathrm{opt}_v(c,T)$ `for item set` T `and budget` B.
  3. `Set a threshold` $t = \frac{v(\mathrm{opt}(c,T))}{8B}$.
  4. `For items in` $A \setminus T$ `find a set` $S^* \in \underset{S \subseteq A \setminus T}{\mathrm{argmax}} \big\{v(S) - t \cdot c(S)\big\}$.
  5. `Sample a cost vector` $d \sim \mathcal{D}$ `conditioned on`
     (a) $d(i) = c(i)$ `for each` $i \in T$`, and`
     (b) $S^* \in \underset{S \subseteq A \setminus T}{\mathrm{argmax}} \big\{v(S) - t \cdot d(S)\big\}$.
  6. `If` $d(S^*) < B$`, let the set` $\{i \in S^* \mid c(i) \leq d(i)\}$ `win.`
  7. `If` $d(S^*) \geq B$
     – `run` XOS-mechanism-main `on the valuation` $\widetilde{v}(\cdot)$`, item set` S^*`, cost vector` $c(\cdot)$`, and budget` B.
     – `Output the result of` XOS-mechanism-main.

Remark 3. *In the mechanism,*

- *Steps (1-3) are the same as* XOS-random-sample *where we sample randomly a test group T and generate a threshold value t.*
- *In Steps (4-7), we consider a specific subset $S^* \subseteq A \setminus T$, and further select winners from S^* only.*
- *Step (5) provides a guidance on the threshold payments to the winners (see discussion below).*
- *Step (7) runs* XOS-mechanism-main *on the function $\widetilde{v}(\cdot)$ (defined as the optimal value of the LP ($\Diamond$)), which is XOS function, according to Lemma 15.*

Further remarks about the mechanism are in order.

- It is tempting to remove the random sampling part, as given $\mathcal{D}$ one may consider a 'prior sampling' approach: Generate some virtual instances according to $\mathcal{D}$ and compute a threshold t based on them; then apply this threshold to all agents in A. Interestingly, the prior sampling approach works well in our mechanism for, e.g., the case when all $c(i)$'s are independent, but it does not work for the case when variables are dependent.

  For instance, consider additive valuation $v(S) = |S|$, budget $B = 2^k$ for a large k, and a set of $N = 2^k$ agents with the following discrete distribution over costs ($c = \ell$ means that $c(i) = \ell$ for all i):

$$\mathbf{Pr}[c = 1] = \frac{1}{2^{k+1}}, \mathbf{Pr}[c = 2] = \frac{1}{2^k}, \ldots,$$
$$\mathbf{Pr}\big[c = 2^k\big] = \frac{1}{2}, \mathbf{Pr}\big[c = 2^{k+1}\big] = \frac{1}{2^{k+1}}.$$

  Note that

$$v(\mathrm{opt}(c = 1)) = 2^k, v(\mathrm{opt}(c = 2)) = 2^{k-1}, \ldots\ldots,$$
$$v\big(\mathrm{opt}\left(c = 2^k\right)\big) = 1, v\big(\mathrm{opt}\left(c = 2^{k+1}\right)\big) = 0.$$

  The expected optimal value is $\mathbf{E}[\mathrm{opt}] = \frac{k+1}{2}$ and it is equally spread over all possible costs except the last one $c = 2^{k+1}$. Roughly speaking, on a given instance c, any prior estimate on $v(\mathrm{opt}(c))$ that gives a constant approximation only applies to a constant number of distinct costs (the contribution of these cases to $\mathbf{E}[\mathrm{opt}]$ is negligible). Hence for the rest almost all possible costs, we get a meaningless estimate for $\mathrm{opt}(c)$. Therefore, the prior sampling will lead to a bad approximation ratio.

- Why do we generate another cost vector d in Step (5)? Recall that our target winner set is S^*, whose value $v(S^*)$ in expectation gives a constant approximation to $\mathbf{E}[\mathrm{opt}]$. However, we face the problems of selecting a winning set in S^* of a sufficiently large value and distributing the budget among the winners. These two problems together are closely related to cooperative game theory and the notion of approximate core. For subadditive valuations, a constant approximate core may not exist [60] (e.g., set cover provides a logarithmic lower bound [12]). Thus we might not be able to pick a winning set with a constant approximation and set threshold payments in accordance with the valuation function. The question then is: is there any other guidance we can take to bound budget feasible threshold payments and give a constant approximation?

Our solution is to use another random vector d that serves as such a guidance. (Conditions in Steps (5a) and (5b), from a high level point of view, guarantee that the vector d is not too 'far' from c for the agents in S^*, in the sense that both vectors are derived from the same distribution. Thus, cost vectors c and d are distributed symmetrically and can be switched while preserving some important parameters such as t and S^* in expectation.) If $d(S^*) \leq B$ (Step (6)), then we set $d(i)$ as an upper bound on the payment of each agent $i \in S^*$; this guarantees that we are always within the budget constraint. If $d(S^*) > B$, setting $d(i)$ as an upper bound is not sufficient to ensure budget feasibility; then we adopt our approach for XOS functions with inputs subset S^* and XOS valuation $\widetilde{v}(\cdot)$ defined by ($\Diamond$).

Theorem 20. SA-Bayesian-mechanism *is a universally truthful budget feasible mechanism for subadditive functions and gives in expectation a constant approximation.*

Proof. Budget feasibility follows simply from the description of the mechanism and the fact that XOS-mechanism-main is budget feasible.

For universal truthfulness, we note that in the mechanism, the sampled vector d comes from a distribution that depends on actual bid vector c. To see why our mechanism takes a distribution over deterministic truthful mechanisms, we can describe all possible samples d for (i) all possible cost vectors on T and (ii) all possible choices $S \subseteq A \setminus T$ of S^*; then we tell all flipped d's to the agents before looking at the costs of T. (Practically, we can provide all our randomness as a black box accessible by all agents.) Note that the selection rule of S^* is monotone, and, similarly to Claim 3, each agent in S^* cannot manipulate (i) the composition of S^* given c and T, and (ii) the choice of d, as long as he stays in S^*. Therefore, composing the first part choosing S^* (Step (4)) with the next monotone rule picking winners in S^* (Steps (6-7)), we again get a monotone winner selection rule. Hence, the mechanism is universally truthful.

Next we give a sketch of the idea of proving the constant approximation.

Approximation analysis (sketch). We sketch the proof idea of the approximation ratio of the mechanism. Later in section 5.1.1 we give a complete argument. First, similar to our analysis in Section 4.6, the optimal solution $v(\mathrm{opt}(c,T))$ obtained from random sampling in expectation gives a constant approximation to the optimal solution $\mathbf{E}[\mathrm{opt}]$. Further, we observe the following facts (which are reminiscent of Claim 2):

$$\widetilde{v}(S) - t \cdot c(S) \geq 0 \quad \text{and} \quad \widetilde{v}(S) - t \cdot d(S) \geq 0, \quad \forall S \subseteq S^*$$

where the second inequality is based on the conditional distribution we choose for d.

If $c(S^*) \geq B$ and $d(S^*) \geq B$ (i.e., the mechanism runs Step (7)), we can pick a subset $S^0 \subseteq S^*$ with $B \geq c(S^0) \geq \frac{B}{2}$. By Theorem 17 XOS-mechanism-main gives a constant approximation to the optimum of $\widetilde{v}(\cdot)$ on S^*. (This is the reason why in Step (7) of the mechanism, we run the whole XOS-mechanism-main on the input instance $\widetilde{v}(\cdot)$ and S^*.) Hence,

$$\widetilde{v}(\mathrm{opt}_{\widetilde{v}}(c,S^*)) \geq \widetilde{v}(S^0) \geq t \cdot c(S^0) \geq t \cdot \frac{B}{2} \geq \frac{v(\mathrm{opt}(c,T))}{16},$$

where the first inequality follows from the fact that $S^0 \subseteq S^*$ is a budget feasible set. Thus, the optimum of $\widetilde{v}(\cdot)$ on S^* is within a constant factor of $v(\mathrm{opt}(c,T))$, as well as the benchmark $\mathbf{E}[\mathrm{opt}]$.

If $c(S^*) < B$ and $d(S^*) \geq B$, we have $\widetilde{v}(S^*) \geq t \cdot d(S^*) \geq \frac{v(\mathrm{opt}(c,T))}{8}$. Further, we notice that S^* is budget feasible with respect to cost vector c; thus, XOS-mechanism-main gives a constant approximation to $\widetilde{v}(S^*)$, which in turn is within a constant factor of $v(\mathrm{opt}(c,T))$ and $\mathbf{E}[\mathrm{opt}]$.

We observe that the vectors d and c restricted to the agents in S^* and conditioned on

$$S^* \in \operatorname*{argmax}_{S\subseteq A\setminus T}\big\{v(S) - t\cdot d(S)\big\}, \qquad S^* \in \operatorname*{argmax}_{S\subseteq A\setminus T}\big\{v(S) - t\cdot c(S)\big\}$$

have exactly the same distributions. Therefore, due to such a symmetry between d and c, in a run of our mechanism in expectation we will have the outcome T, t, S^* and a pair of vectors (c,d) equally often as the outcome T,t,S^* and the pair (d,c). This implies that in the case when $d(S^*) < B$ and $c(S^*) < B$, we get on average the value of at least $\frac{1}{2}v(S^*)$, since the winning sets on the two instances where c (resp., d) is the private cost and d (resp., c) is the sampled cost altogether cover S^*, and v is a subadditive function. By the choice of threshold t, we also know that

$$\begin{aligned} v(S^*) &\geq v(S^*) - t\cdot c(S^*) \\ &\geq v(\mathrm{opt}(c, A\setminus T)) - t\cdot c(\mathrm{opt}(c, A\setminus T)) \\ &\geq v(\mathrm{opt}(c, A\setminus T)) - t\cdot B. \end{aligned}$$

Thus, our mechanism gives a constant approximation to $v(\mathrm{opt}(c))$ with some constant probability.

The last case is when $c(S^*) \geq B$ and $d(S^*) < B$. Again due to the symmetry between c and d, intuitively, we can treat this case as the above one when c and d are switched; thus we also get a constant approximation to $\mathbf{E}[\mathrm{opt}]$. (The formal argument, however, due to multiple randomness used in the mechanism, is much more complicated.) Therefore, the mechanism SA-Bayesian-mechanism on average has a constant approximation to the expected socially optimal value $\mathbf{E}[\mathrm{opt}]$. □

5.1.1 Approximation Guarantees

Below we provide a formal proof for the constant approximation of SA-Bayesian-mechanism. We first have the following observation.

Claim 4. *For any $S \subseteq S^*$, $\widetilde{v}(S) - t \cdot c(S) \geq 0$.*

Proof. Indeed, recall that $S^* = \arg\max_{S \subseteq A \setminus T} \{v(S) - t \cdot c(S)\}$. Then $v(S) - t \cdot c(S) \geq 0$ for any $S \subseteq S^*$, since otherwise, we have $v(S) - t \cdot c(S) < 0$ for some $S \subseteq S^*$ and we get

$$
\begin{aligned}
v(S^* \setminus S) - t \cdot c(S^* \setminus S) &\geq v(S^*) - v(S) - t \cdot c(S^* \setminus S) \\
&= v(S^*) - t \cdot c(S^*) - \big(v(S) - t \cdot c(S)\big) \\
&> v(S^*) - t \cdot c(S^*),
\end{aligned}
$$

a contradiction.

Thus, for any $S \subseteq S^*$, $v(S) \geq t \cdot c(S)$. Therefore, in the description of $LP(S)$, we have $v(S_j) \geq t \cdot c(S_j)$ for each $j \in [1, N]$. Now substituting $v(S_j)$ for $c(S_j)$ in $LP(S)$, we get the desired inequality $\widetilde{v}(S) = LP(S) \geq t \cdot c(S)$. □

We assume that no single item can have cost more than B in the cost vector c. We need extra notation. We assume that the distribution $\mathcal{D}$ is given on the probability space Ω with corresponding density function $\rho(x)$ on $\mathbb{R}^{|A|}$. For a set $T \subseteq A$, let x_T be a point in $\mathbb{R}^{|T|}$. We will denote by $\rho(x_T)$ the distribution's density we get on the corresponding space $\mathbb{R}^{|T|}$ by sampling $x \in \mathbb{R}^{|A|}$ from $\mathcal{D}$ and restricting it to T-coordinates of x. By $\rho(x|x_T)$ we denote the conditional density obtained by fixing T-coordinates of x to be the same as in x_T. By $\mathrm{opt}(x, S)$ we denote the optimal value we can get from the set $S \subseteq A$ on cost vector x with budget B. For brevity sometimes we will omit x or S, in the latter case $S = A$; sometimes we take the optimum over valuation $\widetilde{v}$ instead of v, so to emphasis this we employ notation $\mathrm{opt}_{\widetilde{v}}(x, S)$. In the mechanism we compute the set $S^*(c, T)$, which depends on the sampled set T and cost vector c. By $XOS(c, S^*)$ we denote the value we get from XOS-mechanism-main(c, S^*) run on the set S^*, cost vector c and XOS valuation $\widetilde{v}(\cdot)$ that alone depends on $v(\cdot)$.

In the following we write explicitly the expected value of the second part of our mechanism.

$$\int_{\Omega} \frac{1}{2^{|A|}} \sum_{T \subset A} \int_{\Omega} f(x, y, T) \rho\Big(y \,\Big|\, y = x\,|_{T} ; S^*(x, T) = S^*(y, T)\Big) \,\mathrm{d}y \rho(x) \mathrm{d}x, \quad (5.1)$$

where

$$f(c, d, T) = \begin{cases} XOS(c, S^*) & \text{if } d(S^*) \geq B \\ v(S^* \cap \{i : c(i) \leq d(i)\}) & \text{if } d(S^*) < B, \end{cases}$$

Swapping in (5.1) the sum and integral we get

$$\begin{aligned}
&\frac{1}{2^{|A|}} \sum_{T \subset A} \int_{\Omega} \int_{\Omega} f(x, y, T) \rho\Big(y \,\Big|\, y = x\,|_{T} ; S^*(y, T) = S^*(x, T)\Big) \,\mathrm{d}y \rho\Big(x\Big) \mathrm{d}x = \\
&\qquad \frac{1}{2^{|A|}} \sum_{T \subset A} \int_{\Omega(T)} \int_{\Omega} \int_{\Omega} f(x, y, T) \rho\Big(y \,\Big|\, x_T ; S^*(y, T) = S^*(x, T)\Big) \,\mathrm{d}y \\
&\qquad\qquad \rho\Big(x \,\Big|\, x_T\Big) \,\mathrm{d}x\, \rho\Big(x_T\Big) \mathrm{d}x_T = \\
&\frac{1}{2^{|A|}} \sum_{T \subset A} \int_{\Omega(T)} \int_{\Omega} \sum_{S \subset A \setminus T} \left[\int_{\Omega} f(x, y, T) \rho\Big(y \,\Big|\, x_T ; S^*(y, T) = S^*(x, T)\Big) \,\mathrm{d}y \right] \\
&\qquad \rho\Big(x \,\Big|\, x_T ; S^*(x, T) = S\Big) \cdot Pr\Big(S^*(x, T) = S \,\Big|\, x_T\Big) \,\mathrm{d}x\, \rho\Big(x_T\Big) \mathrm{d}x_T =
\end{aligned}$$

$$\frac{1}{2^{|A|}}\sum_{T\subset A}\int_{\Omega(T)}\sum_{S\subset A\setminus T}\left[\int_{\Omega}\int_{\Omega} f(x,y,T)\rho\Big(y\,\Big|x_{_T};S^*(y,T)=S\Big)\,\mathrm{d}y\cdot\right.$$

$$\left.\rho\Big(x\,\Big|x_{_T};S^*(x,T)=S\Big)\,\mathrm{d}x\right] Pr\Big(S^*(\cdot,T)=S\,\Big|x_{_T}\Big)\;\;\rho\Big(x_{_T}\Big)\cdot\mathrm{d}x_{_T}=$$

$$\frac{1}{2^{|A|}}\sum_{T\subset A}\int_{\Omega(T)}\sum_{S\subset A\setminus T}\int_{\Omega}\int_{\Omega}\frac{f(x,y,T)+f(y,x,T)}{2}\rho\Big(y\,\Big|x_{_T};S^*(y,T)=S\Big)\,\mathrm{d}y$$

$$\rho\Big(x\,\Big|x_{_T};S^*(x,T)=S\Big)\,\mathrm{d}x\;Pr\Big(S^*(\cdot,T)=S\,\Big|x_{_T}\Big)\;\;\rho\Big(x_{_T}\Big)\cdot\mathrm{d}x_{_T}.$$

To get the first equality we split the integral w.r.t. variable x into two integrals w.r.t. variables $x_{_T}\in\mathbb{R}^{|T|}$ and x conditioned on $x=x_{_T}$ on T; the second equality follows from the law of total expectation applied to the variable in the square brackets with conditioning on all possible values of random variable $\{S^*(\cdot,T)|x_{_T}\}$; to get the third equality we swap integral with the sum; to get the last equality we have used the symmetry between x and y for the expression in square brackets. Next, the formula (5.1) can be written as

$$\int_{\Omega}\frac{1}{2^{|A|}}\sum_{T\subset A}\int_{\Omega(x,T)}\frac{f(x,y,T)+f(y,x,T)}{2}$$

$$\rho\Big(y\,\Big|y=x\,|_{_T};S^*(x,T)=S^*(y,T)\;\Big)\,\mathrm{d}y\rho(x)\mathrm{d}x,\qquad(5.2)$$

Now we estimate the value of $XOS(c,S^*)$ in the case when $d(S^*)\geq B$. Recall that $\widetilde{v}(S)\leq v(S)$ for any set $S\subseteq A$. According to the results of previous section for the XOS functions we have a constant approximation to the optimum: $XOS(c,S^*)\geq\alpha\cdot\widetilde{v}(\mathrm{opt}_{\widetilde{v}}(c,S^*))$, for a positive constant α. Now we consider two cases for $c(S^*)$.

1. $c(S^*)<B$. Since one can buy the whole set S^*, we have $\mathrm{opt}_{\widetilde{v}}(c,S^*)=S^*$. We recall that by definition vector d is sampled so that $d(i)=c(i)$ on

$i \in T$, and $S^*(c) = S^*(d)$. We have $\widetilde{v}(S^*) \geq t \cdot d(S^*)$ by Claim 4 applied to the cost vector d. Hence, recalling the definition of t we get $\widetilde{v}(\mathrm{opt}_{\widetilde{v}}(c, S^*)) \geq \frac{v(\mathrm{opt}(c,T))}{8B} B = \frac{v(\mathrm{opt}(c,T))}{8}$.

2. $c(S^*) \geq B$. Since any single item in S^* has cost less than B, we can find a set $S_0 \subset S^*$, such that $B \geq c(S_0) \geq \frac{B}{2}$. We observe that $\widetilde{v}(\mathrm{opt}_{\widetilde{v}}(c, S^*)) \geq \widetilde{v}(S^0)$, as S_0 is a budget feasible set for the cost vector c. Then we have $\widetilde{v}(S^0) \geq t \cdot c(S^0)$ by Claim 4 and $\widetilde{v}(\mathrm{opt}_{\widetilde{v}}(c, S^*)) \geq \widetilde{v}(S^0) \geq tc(S^0) \geq \frac{v(\mathrm{opt}(c,T))}{16}$.

Thus

$$f(c, d, T) = XOS(c, S^*) \geq \frac{\alpha}{16} v(\mathrm{opt}(c, T)),$$

if $d(S^*) \geq B$. By symmetry between c and d we have $XOS(d, S^*) \geq \frac{\alpha}{16} v(\mathrm{opt}(d, T))$, if $c(S^*) \geq B$. We note that c coincides with d on T, and hence $\mathrm{opt}(c, T) = \mathrm{opt}(d, T)$. Therefore,

$$f(d, c, T) = XOS(d, S^*) \geq \frac{\alpha}{16} v(\mathrm{opt}(c, T)),$$

if $c(S^*) \geq B$.

Next we estimate $f(c, d, T) + f(d, c, T)$.

1. If either $c(S^*) \geq B$ or $d(S^*) \geq B$, then we observe that due to the last inequalities for $f(c, d, T)$ and $f(d, c, T)$

$$f(c, d, T) + f(d, c, T) \geq \frac{\alpha}{16} v(\mathrm{opt}(c, T)).$$

2. If $c(S^*) < B$ and $d(S^*) < B$, then by subadditivity of v we get

$$f(c,d,T) + f(d,c,T) = v(S^* \cap \{i : c(i) \leq d(i)\}) + v(S^* \cap \{i : c(i) \geq d(i)\}) \geq v(S^*).$$

Hence we can write a lower bound on $f(c, d, T) + f(d, c, T)$

$$f(c, d, T) + f(d, c, T) \geq \min\left(v(S^*), \frac{\alpha}{16} v(\mathrm{opt}(c, T))\right),$$

which does not depend on d. We plug in this lower bound, instead of $f(c,d,T)+f(d,c,T)$, into the formula (5.2). We obtain

$$\int_{\Omega} \frac{1}{2^{|A|}} \sum_{T \subset A} \int_{\Omega(x,T)} \frac{\min\left(v(S^*), \frac{\alpha}{16} v(\mathrm{opt}(x,T))\right)}{2} \rho\Big(y \,\Big|\, y = x\,|_T; S^*(x,T) = S^*(y,T)\Big)$$
$$\mathrm{d}y\rho(x)\mathrm{d}x = \int_{\Omega} \frac{1}{2^{|A|}} \sum_{T \subset A} \min\left(\frac{v(S^*(x,T))}{2}, \frac{\alpha}{32} v(\mathrm{opt}(x,T))\right) \rho(x)\mathrm{d}x.$$

The equality holds, since y comes from probability distribution and the function under integral does not depend on y.

Let $i^*(x)$ be the most valuable item in A with $c(i^*) \leq B$. We can write the following lower bound on the total valuation of SA-Bayesian-mechanism.

$$\int_{\Omega} \left(0.5 \times v(i^*(x)) + 0.5 \times \frac{1}{2^{|A|}} \sum_{T \subset A} \min\left(\frac{v(S^*(x,T))}{2}, \frac{\alpha}{32} v(\mathrm{opt}(x,T))\right)\right) \rho(x)\mathrm{d}x.$$

The optimal expected value is

$$\int_{\Omega} v(\mathrm{opt}(x))\rho(x)\mathrm{d}x.$$

Next we argue that the function

$$g(x) := v(i^*(x)) + \frac{1}{2^{|A|}} \sum_{T \subset A} \min\left(\frac{v(S^*(x,T))}{2}, \frac{\alpha}{32} v(\mathrm{opt}(x,T))\right)$$

approximates $v(\mathrm{opt}(x))$ within a constant factor for any cost vector x.

We fix a cost vector x. Let $v(i^*) = \frac{1}{k} v(\mathrm{opt})$, for some $k \geq 1$. We know that due to the Lemma 11 $\min\left(v(\mathrm{opt}(A \setminus T), v(\mathrm{opt}(T))\right) \geq \frac{k-1}{4k} v(\mathrm{opt})$ with probability at least $\frac{1}{2}$. For each "good" T, i.e., such that $\min\left(v(\mathrm{opt}(A \setminus T), v(\mathrm{opt}(T))\right) \geq \frac{k-1}{4k} v(\mathrm{opt})$, we can write:

$$v(\mathrm{opt}(T)) \geq \frac{k-1}{4k} v(\mathrm{opt})$$

and

$$\begin{aligned} v(S^*) &\geq v(S^*) - t \cdot x(S^*) \geq v(\text{opt}(A \setminus T)) - t \cdot x(\text{opt}(A \setminus T)) \\ &\geq \frac{k-1}{4k} v(\text{opt}) - t \cdot B \geq \frac{k-1}{4k} v(\text{opt}) - \frac{v(\text{opt})}{8B} \cdot B \\ &\geq \frac{k-2}{8k} v(\text{opt}). \end{aligned}$$

The second inequality holds because $S^* \in \arg\max_{S \subseteq A \setminus T} \{v(S) - t \cdot x(S)\}$; the third inequality holds because the cost of feasible solution $\text{opt}(A \setminus T)$ is within the budget; in the forth inequality we plugged in the definition of t and used the fact that $v(\text{opt}) \geq v(\text{opt}(T))$.

Therefore, combining these two lower bounds for all "good" T we get

$$\begin{aligned} g(x) &\geq \left(\frac{1}{k} + \frac{1}{2} \min\left(\frac{k-2}{16k}, \frac{(k-1)\alpha}{128k}\right)\right) v(\text{opt}(x)) \\ &\geq \frac{1}{2}\left(\min\left(\frac{k+14}{16k}, \frac{k\alpha}{128k}\right)\right) v(\text{opt}(x)) \\ &\geq \frac{\alpha}{256} v(\text{opt}(x)). \end{aligned}$$

Hence, we have shown that the expected value of SA-Bayesian-mechanism is within a constant factor of $\frac{\alpha}{512}$ from the expected value of the optimal solution.

5.1.2 Back to Prior-free

Finally, we turn back to the original prior-free worst-case framework and based on the results from Bayesian framework show the existence of budget feasible incentive compatible mechanism with constant approximation to the optimum for any subadditive valuation and any bid vector. This settles in the affirmative the "fundamental question"' posed by Dobzinski, Papadimitriou, Singer in [30].

Theorem 21. *For any given subadditive valuation $v(\cdot)$ and budget B there exists a budget feasible incentive compatible mechanism with a constant approximation to the optimum.*

Proof. Let $\mathcal{A}$ be the space of all universally truthful budget feasible mechanisms. Note that $\mathcal{A}$ forms a convex set, since for any two mechanisms $A_1, A_2 \in \mathcal{A}$ one may define another universally truthful budget feasible mechanism $A = \lambda \cdot A_1 + (1-\lambda) \cdot A_2$, which with probability λ runs A_1 and with probability $1-\lambda$ runs A_2.

Without loss of generality, we may consider only those situations where every agent could submit finitely many different numbers as a bid (say all integer multiples of $\cdot\frac{B}{2^n}$ not exceeding $B+1$). Note that there are only finitely many possible allocation rules over the finite space of feasible bids. Therefore, we may also assume that there are only finitely many deterministic truthful mechanisms in $\mathcal{A}$. We notice as well that all our integrability assumptions on the prior distribution in Bayesian framework trivially hold true for any distribution over a finite set.

We now recall our constant ($\lambda_0 > 0$) approximation results from Theorem 20 in the Bayesian framework: for any given distribution $\mathcal{D}$ of cost vector there is a mechanism $A \in \mathcal{A}$ s.t.

$$\mathbf{E}_{\mathbf{c}\sim\mathcal{D}}\big[v(A(\mathbf{c}))\big] \geq \lambda_0 \cdot \mathbf{E}_{\mathbf{c}\sim\mathcal{D}}\big[v(\mathrm{opt}(\mathbf{c}))\big].$$

Let us consider a 2 player game with the first player deciding on a feasible cost vector $\mathbf{c}$ and the second player choosing a deterministic truthful budget feasible mechanism $A \in \mathcal{A}$. Note that each player has a finite number of pure strategies in this game. We define each entry of the payoff matrix (i.e., the amount that player 1 pays to player 2) for the pair $(\mathbf{c}, A)$ as $v(A(\mathbf{c})) - \lambda_0 v(\mathrm{opt}(\mathbf{c}))$ in the normal form of our game. Next we apply Yao's min-max principle:

$$\min_{\mathcal{D}} \max_{A\in\mathcal{A}} \mathop{\mathbf{E}}_{\mathbf{c}\sim\mathcal{D}} \left[v(A(\mathbf{c})) - \lambda_0 \cdot v(\mathrm{opt}(\mathbf{c}))\right] = \max_{\mathfrak{D}} \min_{\mathbf{c}} \mathop{\mathbf{E}}_{A\sim\mathfrak{D}} \left[v(A(\mathbf{c})) - \lambda_0 \cdot v(\mathrm{opt}(\mathbf{c}))\right].$$

From our results for Bayesian setting the LHS is non negative. Hence there is a distribution $\mathfrak{D}$ of deterministic truthful budget feasible mechanisms in the RHS, such that for any cost vector $\mathbf{c}$

$$\mathbf{E}_{A\sim\mathfrak{D}}\big[v(A(\mathbf{c}))\big] \geq \lambda_0 \cdot v(\mathrm{opt}(\mathbf{c})).$$

This concludes the proof as we can take a randomized universally truthful budget feasible mechanism that simply runs $A \sim \mathfrak{D}$ and achieves λ_0 factor of approximation. $\square$

5.1.3 Discussion and Open Questions

In the previous two chapters we considered budget feasible mechanism incentive compatible design in two analysis frameworks: prior-free and Bayesian. For XOS functions, there is a prior-free constant approximation mechanism. For subadditive functions, we have two prior-free mechanisms with integrality-gap and sub-logarithmic worst-case approximations to the optimum, as well as a Bayesian constant approximation mechanism. In addition to that we have shown the existence of constant approximation prior-free mechanism for subadditive valuations.

One can adapt these mechanisms to the extension where the valuation function is non-monotone, i.e., $v(S)$ is not necessarily less than $v(T)$ for $S \subset T \subseteq A$. For instance, the cut function studied in [30] is non-monotone. For such valuations, we can define $\hat{v}(S) = \max_{T\subseteq S} v(T)$ for any $S \subseteq A$. It is easy to see that $\hat{v}(\cdot)$ can be evaluated given a demand oracle, is a monotone function, and inherits the hierarchy of classes of $v(\cdot)$, i.e., if $v(\cdot)$ is submodular/XOS/subadditive function, so does $\hat{v}(\cdot)$. Further, any solution maximizing the valuation $v(\cdot)$ is also an optimal solution for $\hat{v}(\cdot)$ as well. Hence one can apply the above mechanisms to $\hat{v}(\cdot)$ directly and obtain the same order of approximation.

We have constant approximation results in the Bayesian framework provided by SA-Bayesian-mechanism for a known distribution. Unfortunately, our results provide approximation guarantees only for relatively large constants. This suggests a natural further research direction of getting smaller and if possibly tight bounds on the approximation in both prior-independent and Bayesian frameworks. An important and interesting question within these two frameworks is the prior-independent mechanism design with certain assumptions on the underlying distribution of costs. In this framework we know that there is some specific type of the distribution from which the bid vector is drawn, although we and the mechanism designer are not aware of which distribution it is exactly. For example one may look at the case when agent's true costs are i.i.d. but with unknown underlying distribution. One possible strategy is still to use random sampling approach, as by sampling randomly part of the agent's population we observe many independent samples from the underlying distribution and basically can learn it. However, it would be interesting to see mechanisms with a small constant factors of approximation for such a scenario.

For those mechanisms with exponential running time, it is natural to ask if there are truthful designs with the same approximations that can be implemented in polynomial time. Further, all of the aforementioned mechanisms are randomized; thus the question on approximability of deterministic truthful mechanisms remains open. We leave these as a future work. It also would be interesting to see a stronger lower bound than that presented in this thesis.

In the last two chapter we have seen a noticeable difference between designing budget feasible mechanism for XOS and for subadditive valuations. Another point of view for such a distinction between the two classes is from exponential concentration: in the XOS case the valuation of a randomly selected subset obeys an exponential concentration around its expected value, whereas in the case of general subadditive valuations it does not. We note that the exponential concentration may be used to improve the approximation ratio of XOS functions.

5.2 Multi Item Sellers

5.2.1 The Model

We now turn to another possible extension of the original model with multi-item bidders in the prior-free worst-case framework. In this extension we allow every agent i to have more than one item for sale and submit bids on a range R_i of items. To distinguish this extension from the previous single parameter case we will mostly refer to agents as shops.

Formally, in the multi-item model there are n shops and every shop i has a range R_i of offered goods, where all different ranges R_i are assumed to be disjoint. Further, each shop has a true private cost c_j for every item $j \in R_i$ and places a bid $b_i(j)$ on it. The auctioneer in his turn upon receiving the bids $(b_i(\cdot))_{i\in[n]}$ decides which set $S = \bigcup_{i=1}^{n} S_i$ of items to buy and on the price vector $\mathbf{p} = (p_1, \ldots, p_n)$ to pay to each shop. Every agent i has a quasi-linear utility, i.e., given the set S_i of sold items and received payment p_i the derived utility is $u_i = p_i - \sum_{j\in S_i} c_j$. The property of incentive compatibility is the same as before, i.e., it should be of the best interest for every agent to bid his true costs on offered items.

As before we assume that the auctioneer has a complex valuation function over the different bundles of items. Our goal is to design an incentive compatible mechanism, that has the total payment below the budget and approximates the optimal solution to the optimization problem with sharp budget constraint on every possible bid vector.

Unfortunately, this direct extension fails even for additive valuation in the case of a single shop. Indeed, let the shop have a large number of identical items for sale. Then it might raise the price of every item up to the budget, therefore, enforcing auctioneer to buy only one item at the highest possible price. On the other hand, in the optimal solution auctioneer might have bought all the items and obtain much higher value. The similar problem occurs even if the shop has just two items: an expensive item of a high value and a cheap item of a low value.

Again auctioneer would not have any choice but to buy the cheapest item and pay his budget, while in the optimal solution he could get much more valuable item.

An inquisitive reader might have noticed that in the single parameter setting we avoided all these problems. The reason is that we could safely remove from the auction the agents bidding above the budget on their items. In fact, such a reduction, if not stated explicitly, was being done silently from the very beginning in all the previous work on the budget feasible mechanism design including this thesis.

Let us consider in more detail the case of a single shop (the monopolist) offering many items. We observe first that if auctioneer gets nothing, he pays nothing. Thus the shop has incentive to sell at least one item if its incurred cost is smaller than the payment of auctioneer. Auctioneer on the other hand prefers to derive any positive value for buying a single item and spending all his budget to buying and paying nothing. Therefore, the shop will sell its cheapest item for the price of auctioneer's budget (of course only if payment covers the item true cost). Thereby, almost in any situation with monopoly seller our auctioneer derives too small value with respect to the optimal solution.

One way to resolve aforementioned problems is to restrict the possible bidding (equivalently true costs) domain. Indeed, in many scenarios (e.g. when government runs a procurement auction), it is the case that auctioneer has a relatively big budget compared to the total bid of every single supplier. However, this approach has a few drawbacks. First, we would need to describe explicitly what are these restrictions and that might vary from one scenario to another. Second, if these restrictions forbid big bids (e.g. bids above the budget) bidders might have incentive not to put some of their goods on the auction at all.

We take another approach with no restrictions on the bidding language of shops. In this approach the value of a truthful mechanism is compared to a benchmark opt* that is similar but slightly weaker than the optimal solution to the original optimization problem opt. The difference between opt* and opt is reminiscent to the difference between benchmarks $\mathcal{F}$ and $\mathcal{F}^{(2)}$ in profit (revenue) maximization, that were introduced in [40].

For each set of agents A, cost vector $\mathbf{c}$ and budget B we denote by $\mathrm{opt}_v(A, B)$ the optimal solution to the optimization problem

$$\max_{S \subset \bigcup_{i \in A} R_i} v(S), \quad \text{s.t.} \sum_{j \in S} c_j \leq B.$$

To obviate the problems with monopolies we compare our mechanisms to a slightly weaker than that of $\mathrm{opt}([n], B)$ benchmark opt*.

Definition 7. *For each cost vector* $\mathbf{c}$

$$\mathrm{opt}^* = \min_{i^* \in [n]} \quad v\left(\mathrm{opt}_v([n] \setminus \{i^*\}, B)\right).$$

Remark 4. *In the single parameter setting, this relaxed benchmark is not too different from the optimum. Indeed, with a constant probability we can run simple truthful mechanism that buys the most valuable single item. Notice that the value obtained by this mechanism together with* opt* *gives an upper bound on the optimal solution.*

We denote by $m_S(R) = v(S \cup R) - v(S)$ the marginal contribution of items in R. Below we describe an incentive compatible mechanism designated for submodular valuation $v(\cdot)$.

<u>Shops-random-sample</u>

1. `Sample shops independently at random in group` T.
2. `Find optimal solution` $\mathrm{opt}_v(T,B)$.
3. `Set payment-per-value conversion rate` $t = \frac{2\cdot B}{v(\mathrm{opt}_v(T,B))}$.
4. `Initialize: purchased items` $S=\emptyset$, `total payment` $p=0$.
5. `In a fixed order visit each shop` $i \in [n] \setminus T$
   (a) `Buy the set` $S_i = \underset{\substack{R\subset R_i:\\ p+t\cdot m_S(R)\le B}}{\mathrm{argmax}} \left(m_S(R) - \frac{b_i(R)}{t}\right)$.
   (b) `Set` $p_i = t\cdot m_S(S_i)$.
   (c) `Set` $S := S\cup S_i$, $p := p + p_i$.
6. `Buy` S, `pay` p_i `to each shop` i

The above mechanism has an intuitive interpretation. First, a customer makes the market study by sampling half of the shops and surveying for the appropriate price-per-value rate t. Next, he shops around by visiting remaining shops one after another. In each shop the customer tells the retailer his valuation, payment-per-value rate and the remaining budget. Then he opts for the best offer that seller can make to him given these parameters and pays the payment-per-value rate multiplied by his value gain. Such a payment rule clearly guarantees budget feasibility.

Claim 5. Shops-random-sample *mechanism is budget feasible.*

Claim 6. Shops-random-sample *mechanism is universally truthful.*

Proof. We analyze truthfulness of the deterministic mechanism for a fixed set T. The agents in group T always derive 0 utility and, therefore, have no reason to change their bids. The order in which the remaining shops are attended does not depend on the bids. The payment to each shop i is proportional to the marginal value that auctioneer gains from items of i, i.e., $p_i = t \cdot m_S(S_i)$. Further, the utility of agent i is

$$u_i = p_i - \sum_{j \in S_i} c_j = t \cdot \left(m_S(S_i) - \frac{\mathbf{c}(S_i)}{t} \right).$$

Given that t is a constant, in Step 5(a) the auctioneer picks the set $R = S_i$, which maximizes the above utility u_i given bid b_i and budget constraint $p + p_i \leq B$.

$$S_i = \operatorname*{argmax}_{\substack{R \subset R_i: \\ p + t \cdot m_S(R) \leq B}} \left(m_S(R) - \frac{b_i(R)}{t} \right).$$

Thereby, the agent i maximizes his utility when reporting truthfully. $\square$

Unfortunately, Shops-random-sample mechanism may not always approximate benchmark opt* well. A bad instance contains only two shops, where the first shop sells a single item X and the second shop sells single item Y, each item at arbitrary price smaller than budget B; the auctioneer's valuation v is additive function with $v(X) = 1, v(Y) = 100$. The benchmark has value opt$^* = 1$. The only possibility for the auctioneer to buy anything and derive non zero value is when the first shop gets sampled in T and the second shop does not. In this case $t = \frac{2 \cdot B}{1}$ and then the auctioneer estimates item Y too high, so that he simply cannot afford to pay $t \cdot m_{\emptyset}(Y) = 200 \cdot B$ for the item. Thus the expected value of the mechanism is 0.

The problem we encounter here is due to the integrality issues in knapsack packing. Indeed, if we would allow to buy a fraction of an item then mechanism would have bought $\frac{1}{200}$ fraction of Y in the latter case and would obtain the value of $\frac{1}{2}$ and thus total expected value of $\frac{1}{8}$. To circumvent this integral issues we employ a complementary mechanism, which is given below.

<u>Exclusive-supplier</u>

1. `Sample shops independently at random in group` T.
2. `Find optimal solution` $\mathrm{opt}_v(T, B)$.
3. `Set value threshold` $t_v = \frac{v(\mathrm{opt}_v(T,B))}{4}$.
4. `In a fixed order visit each shop` $i \in [n] \setminus T$
   (a) `Set` $S = \underset{\substack{R \subset R_i:\\ v(R) \geq t_v}}{\mathrm{argmin}}\, b_i(R)$.
   (b) `If` $S \neq \emptyset$ `and` $b_i(S) \leq B$ `Then buy` S`; pay` $p_i = B$`; halt.`
5. `Buy` $\emptyset$`, pay` 0.

As before the above mechanism has an intuitive interpretation. At this time our customer does market study again by sampling half of the shops. Then he makes a survey and decides what should be his targeted value t_v. Further, the customer consequently visits the rest of the shops aiming for an exclusive supplier. In other words, conditioned on the fact that a shop can supply his demanded value t_v, he takes any offer that shop can make and spends all his budget in that shop.

Each run of the mechanism the payment is either 0 or B, which ensures budget feasibility.

Claim 7. Exclusive-supplier *is budget feasible mechanism.*

Claim 8. Exclusive-supplier *is universally truthful mechanism.*

Proof. The proof is similar to the proof of Claim 6. This time the auctioneer sets the rule that the value of any purchased set S should meet the threshold t_v. Then he agrees with any offer of the shop that meets the condition $v(S) \geq t_v$ and pays all his budget. The potential utility of a shop i in case of such deal is $u_i = B - c_i(S)$. Clearly, this utility will be maximized if agent i bids his true costs $b_i(j) = c_j$ on each $j \in R_i$. $\square$

Finally our main universally truthful and budget feasible mechanism, that runs uniformly at random one of Exclusive-supplier and Shops-random-sample mechanism, has a constant approximation ratio w.r.t. the benchmark opt*.

> Shops-mechanism-main
>
> - With probability 0.5 run Exclusive-supplier.
> - With probability 0.5 run Shops-random-sample.

Theorem 22. Shops-mechanism-main *in universally truthful budget feasible mechanism that in expectation gives a constant approximation to benchmark* opt* *for any submodular valuation* $v(\cdot)$.

Proof. Budget feasibility and universal truthfulness hold due to the Claims 5, 6, 7, 8. We next sketch the idea of constant approximation guarantee. The full argument is given in the next subsection.

Approximation analysis (idea). For each subset $A \subset [n]$ of shops and cost vector $\mathbf{c}$ we define a function $f(A)$ as the maximum value one can achieve on the item set $\cup_{i \in A} R_i$ under the budget constraint. It turns out that $f(\cdot)$ is subadditive function.

Using subadditivity of $f(\cdot)$ we show that $f([n] \setminus T) \geq f(T) \geq \text{opt}^*$ happens with some constant probability for selecting T. For these situations using the properties of submodular function such as in Lemma 8, we show that either Shops-random-sample or Exclusive-supplier mechanism must have a value of at least $\frac{f(T)}{4}$. Thus we get a lower bound constant fraction of opt* on the expected value of Shops-mechanism-main. □

5.2.2 Approximation Analysis.

Here we describe in full details the proof of a constant approximation guarantee on the performance of Shops-mechanism-main. We assume that all shops bid truthfully, i.e., $b_i(j) = c_j$.

Similarly to the previous chapter we first analyze how well the sampling part approximates an optimal solution. We want first to warn reader that in the multi-item case sampling agents is different from sampling items. Now every agent represents a cluster of items that altogether are sampled or not in the set T.

For each subset $A \subset [n]$ of shops we define a function $f(A)$ as a maximal value one can get on the items $\cup_{i\in A} R_i$ under the budget constraint. We note that $f(T) = \text{opt}_v(T, B)$ for any $T \subset [n]$. We show below that $f(\cdot)$ is a subadditive function provided that $v(\cdot)$ is subadditive valuation. We further show that $f(\cdot)$ is XOS function provided that $v(\cdot)$ is XOS.

Interestingly, $f(\cdot)$ is not necessarily submodular function even if $v(\cdot)$ is additive. Here is an example with 3 shops, 4 items and additive valuation $v(\cdot)$: $v(1) = 1, v(2) = 2, v(3) = 3, v(4) = 5$; $c(1) = 1, c(2) = 2, c(3) = 3, c(4) = 4$; seller's budget is 5; first shop offers only item 1, second shop sells items 2 and 3, third shop offers item 4. Then $f(\{2\}) = 5$, $f(\{23\}) = 5$, $f(\{123\}) = 6$ and, therefore, $f(\cdot)$ is not submodular.

Lemma 17. *If $v(\cdot)$ is subadditive, then $f(\cdot)$ is subadditive. If $v(\cdot)$ is XOS, then $f(\cdot)$ is XOS.*

Proof. To verify the subadditivity we need to show $f(X \cup Y) \le f(X) + f(Y)$. Let us assume that in an optimal budget feasible solution for $f(X \cup Y)$ we buy items S_X spending B_X from the agents in X and buy S_Y spending B_Y from the agents in Y. Then $f(X) \ge v(S_X)$, since we can afford to buy S_X with budget B. Similarly, $f(Y) \ge v(S_Y)$. Therefore, $f(X) + f(Y) \ge v(S_X) + v(S_Y) \ge v(S_X \cup S_Y) = f(X \cup Y)$.

In order to show that $f(\cdot)$ is XOS we may specify for each agent $i \in [n]$ a part of the budget we can spend on the items from R_i. Then f can be described as the maximum taken over all possible additive functions f_j in the representation of

$v = \max_{j=1}^{N} f_j(\cdot)$ with any possible set of budget constraints B_i on every bundle R_i ($B_i \geq 0$, $\sum_{i=1}^{|A|} B_i = B$). Clearly, each such function

$$f_j^{B_1,\ldots,B_n}(A) = \sum_{i \in A} \max_{\substack{Y_i \subset R_i: \\ c(Y_i) \leq B_i}} f_j(Y_i)$$

is additive as a function of $A \subset [n]$ and

$$f(A) = \max_{\substack{j, B_1,\ldots,B_n \\ B_1+\cdots+B_n=B}} f_j^{B_1,\ldots,B_n}(A).$$

We notice that in such description there are infinitely many (continuum) of functions $f_j^{B_1,\ldots,B_n}(\cdot)$ and we are allowed to use only finitely many of them. However, for each B_i there are only finitely many different quantities that make any difference for the value of $\max_{\substack{Y_i \subset R_i: \\ c(Y_i) \leq B_i}} f_j(Y_i)$. Thus we need to use only finitely many sets of different B_i and get the representation of f as the maximum over finitely many additive functions. $\square$

Our next lemma is reminiscent of Lemma 11.

Lemma 18. $\Pr_T \left[f([n] \setminus T) \geq f(T) \geq \frac{1}{6} \cdot \mathrm{opt}^* \right] \geq \frac{1}{4}$.

Proof. Recall that $\mathrm{opt}^* = \min_{i^* \in [n]} f([n] \setminus \{i^*\})$. First, we argue that there is a partition of $[n]$ into two disjoint sets S_1 and S_2, such that $f(S_1), f(S_2) \geq \frac{1}{3}\,\mathrm{opt}^*$.

We can start with $S_1 = [n]$, $S_2 = \emptyset$ and move agents one by one from S_1 to S_2. Unavoidably, there will be a moment when $f(S_2)$ is still less than $\frac{1}{3}\,\mathrm{opt}^*$, but right after we move an agent i from S_1 to S_2 we get $f(S_2 \cup \{i\}) \geq \frac{1}{3}\,\mathrm{opt}^*$. We notice that at the same time $f(S_1) > \frac{2}{3}\,\mathrm{opt}^*$, since f is subadditive. If $f(S_1 \setminus \{i\}) \geq \frac{1}{3}\,\mathrm{opt}^*$, then we have found the required partition $S_1 \setminus \{i\}$ and $S_2 \cup \{i\}$. On the other hand, if $f(S_1 \setminus \{i\}) < \frac{1}{3}\,\mathrm{opt}^*$, then $f(\{i\}) \geq f(S_1) - f(S_1 \setminus \{i\}) > \frac{1}{3}\,\mathrm{opt}^*$. In such a case we can simply take $S_1 = \{i\}$ and $S_2 = [n] \setminus \{i\}$. Note that $f(S_1) \geq \frac{1}{3}\,\mathrm{opt}^*$ and $f(S_2) = f([n] \setminus \{i\}) \geq \mathrm{opt}^*$ by definition of *opt**.

Thus, we have a partition of $[n]$ into S_1 and S_2 with $f(S_1) \geq \frac{1}{3}\,\mathrm{opt}^*$ and $f(S_2) \geq \frac{1}{3}\,\mathrm{opt}^*$. Let us denote $[n] \setminus T$ by $\overline{T}$. Notice that when agents from S_1 are sampled in T, either $f(S_1 \cap T) \geq \frac{1}{2} f(S_1) \geq \frac{1}{6}\,\mathrm{opt}^*$, or $f(S_1 \cap \overline{T}) \geq \frac{1}{2} f(S_1) \geq \frac{1}{6}\,\mathrm{opt}^*$. Moreover, each of these events happens with probability at least 0.5. We have similar bounds for agents in S_2, which are drawn in T independently from S_1. Hence, with probability at least 0.5 we have $f(T) \geq \frac{1}{6}\,\mathrm{opt}^*$ and $f(\overline{T}) \geq \frac{1}{6}\,\mathrm{opt}^*$ simultaneously.

Therefore, $\Pr_T\left[\min(f([n]\setminus T), f(T)) \geq \frac{1}{6}\cdot \mathrm{opt}^*\right] \geq \frac{1}{2}$. Due to the symmetry between T and $[n]\setminus T$ we get that $f([n]\setminus T) \geq f(T)$ with probability 0.5 in the previous inequality. Therefore, $\Pr_T\left[f([n]\setminus T) > f(T) \geq \frac{1}{6}\cdot \mathrm{opt}^*\right] \geq \frac{1}{4}$. □

Now we are ready to estimate the expected value of Shops-mechanism-main. Let us fix a sample set T with $f(T) \leq f([n]\setminus T)$ same for both Shops-random-sample and Exclusive-supplier mechanisms. Let S be the output set of items of Shops-random-sample mechanism and U be the optimal set of items in $[n]\setminus T$ under the budget constraint, i.e., $v(U) = f([n]\setminus T) \geq f(T)$ and $\mathbf{c}(U) \leq B$; payment-per-value conversion rate t is $\frac{2B}{f(T)}$.

The total payment of Shops-random-sample mechanism is $t \cdot v(S)$, which is below the budget B. Therefore, $\frac{2B\cdot v(S)}{f(T)} \leq B$ and $v(S) \leq 0.5 \cdot f(T)$. We apply Lemma 8 from the previous chapter to the sets $U \cup S$ and S. It tells us that there is item $j_0 \in U \setminus S$, s.t.

$$\frac{m_S(j_0)}{\mathbf{c}(j_0)} \geq \frac{v(U\cup S) - v(S)}{\mathbf{c}(U \cup S) - \mathbf{c}(S)}.$$

We observe that $\frac{v(U\cup S)-v(S)}{\mathbf{c}(U\cup S)-\mathbf{c}(S)} \geq \frac{f(T)-0.5\cdot f(T)}{\mathbf{c}(U\setminus S)}$, since $v(U\cup S) \geq v(U) \geq f([n]\setminus T) \geq f(T)$ and $v(S) \leq 0.5 f(T)$. Further, $\frac{v(U\cup S)-v(S)}{\mathbf{c}(U\cup S)-\mathbf{c}(S)} \geq \frac{0.5\cdot f(T)}{B}$, because $\mathbf{c}(U\setminus S) \leq \mathbf{c}(U) \leq B$. Since $\frac{0.5\cdot f(T)}{B} = \frac{1}{t}$, we have the following observation

Claim 9. *There is an item $j_0 \in U \setminus S$ s.t. $m_S(j_0) \geq t \cdot \mathbf{c}(j_0)$.*

Let us examine the question why the item j_0 did not appear in the final set S. At certain point the shop i with $j_0 \in R_i$ was considered in Step 5 of Shops-random-sample mechanism and item j_0 was not included in S_i. We recall that

$$S_i = \operatorname*{argmax}_{\substack{R \subset R_i: \\ p+t\cdot m_{S(i)}(R)\le B}} \left(m_{S(i)}(R) - \frac{\mathbf{c}(R)}{t} \right),$$

where $S(i)$ is the current set of winners S selected right before the visit to shop i in the Shops-random-sample mechanism. Note that at this moment $m_{S(i)\cup S_i}(j_0)$ could be only larger then the final quantity $m_S(j_0)$ due to submodularity of $v(\cdot)$. Hence, if in the mechanism we added j_0 to the set S_i, then the value of $m_{S(i)}(S_i) - \frac{\mathbf{c}(S_i)}{t}$ would only increase. Therefore, the only reason why we did not include j_0 in S_i is because of the budget constraint. That is, the payment of the mechanism would be larger than B. Equivalently, $v(S \cup \{j_0\}) = v(S) + m_S(j_0) \ge \frac{f(T)}{2}$, because the total payment is proportional to the total value with the coefficient $\frac{2B}{f(T)}$. In particular, we can make the following observation.

Claim 10. *If $v(S) \le \frac{f(T)}{4}$, then $m_S(j_0) \ge \frac{f(T)}{4}$.*

First we notice that if $v(S) > \frac{f(T)}{4}$ then Shops-random-sample mechanism provides a constant approximation to $f(T)$.

Second, if $v(S) \le \frac{f(T)}{4}$, then $m_S(j_0) \ge \frac{f(T)}{4}$. We notice that then $v(\{j_0\}) \ge m_S(j_0) \ge t_v$, where $t_v = \frac{f(T)}{4}$ in Exclusive-supplier mechanism. Therefore, Exclusive-supplier mechanism must output a nonempty set with total value of at least $t_v = \frac{f(T)}{4}$.

Combining the above two lower bounds, we obtain that the expected value of Shops-mechanism-main in case $f([n] \setminus T) \ge f(T)$ must be at least $0.5 \cdot \frac{f(T)}{4}$. Using this lower bound and Lemma 18 we obtain that expected value of Shops-mechanism-main must be at least

$$\Pr_T \left[f([n] \setminus T) \ge f(T) \ge \frac{1}{6} \cdot \mathrm{opt}^* \right] \cdot \left(0.5 \cdot \frac{f(T)}{4} \right) \ge \frac{\mathrm{opt}^*}{192},$$

which is a constant approximation to the benchmark opt^*.

5.2.3 Computational Issues

The main mechanism as it is described does not work in polynomial time, however it may be implemented "almost" efficiently. Remarks on the efficient implementation of mechanisms Shops-random-sample and Exclusive-supplier are in order below.

1. Instead of using an optimal solution $\text{opt}_v(T, B)$, which is NP-hard to find for a general submodular function v, we may efficiently find a solution that approximates the optimum by a constant factor. Indeed, an approximate solution to the budgeted submodular maximization problem can be found in polynomially many value queries [51]. We notice that substituting optimum with an approximate solution in the sampling part does not affect truthfulness, but may decrease the approximation guarantees by a constant factor.

2. Step 5 of Shops-random-sample and Step 4 of Exclusive-supplier mechanisms in general are hard to implement, even if we are granted access to a more powerful demand oracle for the valuation $v(\cdot)$. However, if every shop is selling only a few items (number is bounded by a constant), then we can implement Shops-mechanism-main in polynomial time. We note that substituting $S_i = \text{argmax}(\cdot)$ or $S = \text{argmin}(\cdot)$ in each Step 5 or Step 4 with an approximate solution would change the value of the final mechanism only by a constant factor. Unfortunately, such a substitution is not incentive compatible.

3. In general case our main mechanism, formally speaking, cannot be implemented in polynomial time, but speaking practically, it actually may be efficiently implemented. Let us consider, for instance, a problematic Step 5 of Shops-random-sample mechanism. We already have convinced ourselves that the selection rule in Step 5(a) and the payment rule in Step 5(b) are aligned with the utility maximization objective of the seller i. Thereby, the objectives of the shop and the auctioneer at this point do coincide and there should not be a problem for both parties to find an agreement. In particular, using only value queries, the auctioneer may efficiently find a good candidate set S_i and suggest it to shop

i. The shop in its turn, having the access to valuation function $v(\cdot)$ and observing the remaining auctioneer's budget $B - p$, may try to find a better solution S_i'. Remarkably, the shop, in order to increase its utility, is only interested to propose a better set S_i' than that of S_i. Auctioneer will agree on every such S_i' as long as it meets all of his easy-to-verify criteria. In this "dialog" between the auctioneer and the seller i, the shop i does not have any incentive to misreport its costs. Indeed, the computational task of figuring out a beneficial bid b_i different from shop's true costs may be only harder than the task of proposing a better set S_i' to the auctioneer. A similar "efficient dialog" between auctioneer and a shop can be implemented in Exclusive-supplier mechanism as well. To sum up, our auctioneer gets in polynomial time a solution that is a constant approximation to the benchmark, while all the shops report truthfully.

Remark 5. *In the field of algorithmic mechanism design it is commonly required that a mechanism, upon receiving the bids from agents, should decide in polynomially many steps on the set of winners and payments, while ensuring incentive compatibility of these decision rules. The latter observation raises a critical point against this classic definition of efficient implementation of a mechanism. It also suggest an extra round of communication between agents and the mechanism designer.*

We propose below a relaxed definition, where an extra round of communication is allowed between auctioneer and agents.

Definition 8. *First, mechanism $\mathcal{M}$ receives the bid vector $\mathbf{b}$ from each of n agents, which may be different from the true cost vector $\mathbf{c}$. Then $\mathcal{M}$ applies a communication protocol exchanging messages with the agents, each time sending to and receiving reply from an agent $i \in [n]$. The mechanism keeps the record r_i on all the replies from every agent i. The total length of the replies $\mathbf{r} = (r_i)_{i\in[n]}$ and the messages sent in the protocol must be bounded by a polynomial in n. Moreover, we assume that $\mathcal{M}$ can only perform computations in polynomial in* *n time. Therefore, each subsequent massage it sends to an agent must be easy to compute for a given bid vector and previous agent's replies. Based on the bid vector* $\mathbf{b}$ *and replies* $\mathbf{r} = (r_i)_{i\in[n]}$*, the final decision on the winning set* $S(\mathbf{b}, \mathbf{r})$ *and payments* $\mathbf{p}(\mathbf{b}, \mathbf{r})$ *is made. We call the mechanism* $\mathcal{M}$ *truthful, if for each agent* i *and fixed bids* $\mathbf{b}_{-i}$ *and replies* $\mathbf{r}_{-i}$ *of other agents there exists efficiently computable function* f_i*, s.t.*

$$\forall b_i', r_i' \quad u_i\left(b_i', \mathbf{b}_{-i}, r_i', \mathbf{r}_{-i}\right) \leq u_i(c_i, \mathbf{b}_{-i}, f_i(b_i', r_i'), \mathbf{r}_{-i}).$$

In other words, mechanism is truthful if and only if any misreported bid of an agent can be effectively turned into at least as good true bid with an appropriate reply message.

The performance of such a mechanism should be measured in the case of worst possible replies provided by the agents.

Chapter 6

Conclusions and Open Problems

In this work we examined two important classes of optimization problems in algorithmic mechanism design. For the first class, we proposed a unified scheme for designing frugal incentive compatible mechanisms. We showed that several existing mechanisms fall under our scheme, and described its applications to k-path systems and vertex cover systems. We demonstrated that our scheme produces mechanisms with good ν-frugality ratios for k-path systems and a large subclass of vertex cover systems; for k-path systems, we showed that our mechanism has the optimal μ-frugality ratio and can be implemented in polynomial time.

For the second class, we proposed a number of constant approximation incentive compatible mechanisms for most of the valuation classes in the hierarchy of complement free functions. We gave the positive answer to the "fundamental question" raised in [30] by presenting a constant approximation mechanism in the Bayesian framework and showing via Yao's min-max principle the existence of constant approximation mechanism in the prior-free framework. We presented two prior-free truthful mechanisms of integrality-gap and sub-logarithmic worst-case approximations to the optimum, as well as a Bayesian constant approximation incentive compatible mechanism, and provided an efficient implementation of our sub-logarithmic mechanism. We proposed a new extension of the original model to the scenario with multi-item sellers and gave a constant approximation incentive compatible mechanism for an arbitrary submodular valuation.

Certainly, much work is left to be done in the area of frugal mechanism design. Beyond k-paths and vertex cover auctions, there are many other set systems in which frugal mechanism design can be applied, e.g., Matching, Independent Set etc. While there exist preliminary studies on some of these settings, their optimal designs still remain a mystery. Thereby, the important question "what are the (nearly) optimal designs of frugal mechanisms in different procurement auctions?" remains widely open.

In the budget feasible model many open questions remain as well. In particular, for the model with multi-item sellers it is not known whether constant approximation mechanisms exist for XOS and subadditive valuaitons. In the unit-item seller model, for those mechanisms with exponential runtime, it is natural to ask if there are truthful designs with the same approximations that can be implemented in polynomial time. Further, all of our mechanisms are randomized; it is intriguing to consider the approximability of deterministic mechanisms. We are also looking for other, not complement free, classes of valuations that admit budget feasible mechanisms with good approximation ratios.

List of Publications

1. *Computing approximate pure Nash equilibria in congestion games*, I. Caragiannis, A. Fanelli, N. Gravin, A. Skopalik, SIGecom Exchanges 2012, p. 26-29.

2. *Approximate Pure Nash Equilibria in Weighted Congestion Games:Existence, Efficient Computation, and Structure*, I. Caragiannis, A. Fanelli, N. Gravin, A. Skopalik, EC 2012, p. 284-301.

3. *Budget Feasible Mechanism Design: From Prior-Free to Bayesian* X. Bei, N. Chen, N. Gravin, P. Lu, STOC 2012, p. 449-458.

4. *Efficient computation of approximate pure Nash equilibria in congestion games*, I. Caragiannis, A. Fanelli, N. Gravin, A. Skopalik, FOCS 2011, p. 532-541.

5. *Dynamics of Profit-Sharing Games*, J. Augustine, N. Chen, E. Elkind, A. Fanelli, N. Gravin, D. Shiryaev, IJCAI 2011, p. 37-42.

6. *On the Approximability of Budget Feasible Mechanisms*, N. Chen, N. Gravin, P. Lu, SODA 2011, p. 685-699.

7. *Note on Shortest k-Paths Problem,* N. Chen, N. Gravin, Journal of Graph Theory. V. 67(1), 2011 p. 34-37.

8. *Frugal Mechanism Design via Spectral Techniques*, N. Chen, E. Elkind, N. Gravin, F. Petrov, FOCS 2010, p. 755-764.

9. *Refining the Cost of Cheap Labor in Set System Auctions*, N. Chen, E. Elkind and N. Gravin, WINE 2009, p. 447-454.

Bibliography

[1] G. Aggarwal and J. D. Hartline. Knapsack auctions. In *SODA*, pages 1083–1092, 2006.

[2] A. Archer and É. Tardos. Truthful mechanisms for one-parameter agents. In *FOCS*, pages 482–491, 2001.

[3] A. Archer and É. Tardos. Frugal path mechanisms. In *SODA*, pages 991–999, 2002.

[4] M. Babaioff, M. Dinitz, A. Gupta, N. Immorlica, and K. Talwar. Secretary problems: weights and discounts. In *SODA*, pages 1245–1254, 2009.

[5] M. Babaioff, N. Immorlica, and R. Kleinberg. Matroids, secretary problems, and online mechanisms. In *SODA*, pages 434–443, 2007.

[6] A. Badanidiyuru, S. Dobzinski, and S. Oren. Optimization with demand oracles. In *ACM Conference on Electronic Commerce*, pages 110–127, 2012.

[7] A. Badanidiyuru, R. Kleinberg, and Y. Singer. Learning on a budget: posted price mechanisms for online procurement. In *ACM Conference on Electronic Commerce*, pages 128–145, 2012.

[8] R. Bar-Yehuda and S. Even. A local-ratio theorem for approximating the weighted vertex cover problem. *North-Holland Mathematics Studies*, 109(C):27–45, 1985.

[9] X. Bei, N. Chen, N. Gravin, and P. Lu. Budget feasible mechanism design: from prior-free to bayesian. In *STOC*, pages 449–458, 2012.

[10] X. Bei and Z. Huang. Bayesian incentive compatibility via fractional assignments. In *SODA*, pages 720–733, 2011.

[11] S. Bhattacharya, G. Goel, S. Gollapudi, and K. Munagala. Budget constrained auctions with heterogeneous items. In *STOC*, pages 379–388, 2010.

[12] K. Bhawalkar and T. Roughgarden. Welfare guarantees for combinatorial auctions with item bidding. In *SODA*, pages 700–709, 2011.

[13] O. Bondareva. Some applications of linear programming to cooperative games. *Problemy Kibernetiki*, 10:119–139, 1963.

[14] G. Călinescu. Bounding the payment of approximate truthful mechanisms. In *ISAAC*, pages 221–233, 2004.

[15] S. Chawla, J. D. Hartline, D. L. Malec, and B. Sivan. Multi-parameter mechanism design and sequential posted pricing. In *STOC*, pages 311–320, 2010.

[16] S. Chawla, D. L. Malec, and A. Malekian. Bayesian mechanism design for budget-constrained agents. In *ACM Conference on Electronic Commerce*, pages 253–262, 2011.

[17] S. Chawla, D. L. Malec, and B. Sivan. The power of randomness in bayesian optimal mechanism design. In *ACM Conference on Electronic Commerce*, pages 149–158, 2010.

[18] N. Chen, E. Elkind, N. Gravin, and F. Petrov. Frugal mechanism design via spectral techniques. In *FOCS*, pages 755–764, 2010.

[19] N. Chen and N. Gravin. A note on k-shortest paths problem. *Journal of Graph Theory*, 67(1):34–37, 2011.

[20] N. Chen, N. Gravin, and P. Lu. Mechanism design without money via stable matching. *CoRR*, abs/1104.2872, 2011.

[21] N. Chen, N. Gravin, and P. Lu. On the approximability of budget feasible mechanisms. In *SODA*, pages 685–699, 2011.

[22] N. Chen and A. R. Karlin. Cheap labor can be expensive. In *SODA*, pages 707–715, 2007.

[23] E. H. Clarke. Multipart pricing of public goods. *Public Choice*, 11:17–33, 1971.

[24] A. Czumaj and A. Ronen. On the expected payment of mechanisms for task allocation. In *PODC*, pages 98–106, 2004.

[25] K. David. personal communication.

[26] N. R. Devanur, J. D. Hartline, A. R. Karlin, and C. T. Nguyen. Prior-independent multi-parameter mechanism design. In *WINE*, pages 122–133, 2011.

[27] P. Dhangwatnotai, T. Roughgarden, and Q. Yan. Revenue maximization with a single sample. In *ACM Conference on Electronic Commerce*, pages 129–138, 2010.

[28] S. Dobzinski. Two randomized mechanisms for combinatorial auctions. In *APPROX-RANDOM*, pages 89–103, 2007.

[29] S. Dobzinski, R. Lavi, and N. Nisan. Multi-unit auctions with budget limits. In *FOCS*, pages 260–269, 2008.

[30] S. Dobzinski, C. H. Papadimitriou, and Y. Singer. Mechanisms for complement-free procurement. In *ACM Conference on Electronic Commerce*, pages 273–282, 2011.

[31] E. Elkind. True costs of cheap labor are hard to measure: edge deletion and vcg payments in graphs. In *ACM Conference on Electronic Commerce*, pages 108–116, 2005.

[32] E. Elkind, L. A. Goldberg, and P. W. Goldberg. Frugality ratios and improved truthful mechanisms for vertex cover. In *ACM Conference on Electronic Commerce*, pages 336–345, 2007.

[33] E. Elkind, A. Sahai, and K. Steiglitz. Frugality in path auctions. In *SODA*, pages 701–709, 2004.

[34] U. Feige. On maximizing welfare when utility functions are subadditive. *SIAM J. Comput.*, 39(1):122–142, 2009.

[35] U. Feige, V. S. Mirrokni, and J. Vondrák. Maximizing non-monotone submodular functions. In *FOCS*, pages 461–471, 2007.

[36] J. Feigenbaum, C. H. Papadimitriou, R. Sami, and S. Shenker. A bgp-based mechanism for lowest-cost routing. In *PODC*, pages 173–182, 2002.

[37] A. Ghosh and A. Roth. Selling privacy at auction. In *ACM Conference on Electronic Commerce*, pages 199–208, 2011.

[38] C. Godsil and G. Royle. *Algebraic graph theory*, volume 207 of *Graduate Texts in Mathematics*. Springer-Verlag, New York, 2001.

[39] A. Goldberg, J. Hartline, A. Karlin, M. Saks, and A. Wright. Competitive auctions. *Games and Economic Behavior*, 55(2):242–269, 2006.

[40] A. V. Goldberg, J. D. Hartline, and A. Wright. Competitive auctions and digital goods. In *SODA*, pages 735–744, 2001.

[41] A. V. Goldberg and R. E. Tarjan. Finding minimum-cost circulations by canceling negative cycles. *J. ACM*, 36(4):873–886, 1989.

[42] T. Groves. incentives in teams. *Econometrica*, 41(4):617–631, 1973.

[43] G. H. Hardy, J. E. Littlewood, and G. Polya. *Inequalities*. Cambridge university press, 1952.

[44] J. D. Hartline, R. Kleinberg, and A. Malekian. Bayesian incentive compatibility via matchings. In *SODA*, pages 734–747, 2011.

[45] J. D. Hartline and B. Lucier. Bayesian algorithmic mechanism design. In *STOC*, pages 301–310, 2010.

[46] J. D. Hartline and T. Roughgarden. Optimal mechanism design and money burning. In *STOC*, pages 75–84, 2008.

[47] N. Immorlica, D. R. Karger, E. Nikolova, and R. Sami. First-price path auctions. In *ACM Conference on Electronic Commerce*, pages 203–212, 2005.

[48] A. Iwasaki, D. Kempe, Y. Saito, M. Salek, and M. Yokoo. False-name-proof mechanisms for hiring a team. In *WINE*, pages 245–256, 2007.

[49] A. R. Karlin, D. Kempe, and T. Tamir. Beyond vcg: Frugality of truthful mechanisms. In *FOCS*, pages 615–626, 2005.

[50] D. Kempe, M. Salek, and C. Moore. Frugal and truthful auctions for vertex covers, flows and cuts. In *FOCS*, pages 745–754, 2010.

[51] A. Krause and C. Guestrin. A note on the budgeted maximization of submodular functions. Technical Report CMU-CALD-05-103, School of Computer Science Carnegie Mellon University, 2005.

[52] V. Krishna. *Auction Theory.* Academic Press, San Diego, 2003.

[53] J. Lee, V. S. Mirrokni, V. Nagarajan, and M. Sviridenko. Non-monotone submodular maximization under matroid and knapsack constraints. In *STOC*, pages 323–332, 2009.

[54] B. Lehmann, D. J. Lehmann, and N. Nisan. Combinatorial auctions with decreasing marginal utilities. In *ACM Conference on Electronic Commerce*, pages 18–28, 2001.

[55] K. Menger. Zur allgemeinen kurventheorie. *Fundamenta Mathematicae*, 10:96–115, 1927.

[56] P. Milgrom. Putting auction theory to work: The simultaneous ascending auction. *Journal of Political Economy*, 108(2):245–272, 2000.

[57] R. B. Myerson. Optimal auction design. *Math. Oper. Res.*, 6(1):58–73, 1981.

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

[59] N. Nisan and A. Ronen. Algorithmic mechanism design (extended abstract). In *STOC*, pages 129–140, 1999.

[60] N. Nisan, T. Roughgarden, E. Tardos, and V. V. Vazirani. *Algorithmic Game Theory.* Cambridge University Press, New York, NY, USA, 2007.

[61] A. D. Procaccia and M. Tennenholtz. Approximate mechanism design without money. In *ACM Conference on Electronic Commerce*, pages 177–186, 2009.

[62] L. S. Shapley. On balanced sets and cores. *Naval Research Logistics Quarterly*, 14(4):453–460, 1967.

[63] Y. Singer. Budget feasible mechanisms. In *FOCS*, pages 765–774, 2010.

[64] Y. Singer. *Incentives, Computation, and Networks: Limitations and Possibilities of Algorithmic Mechanism Design.* PhD thesis, UC Berkeley, 2011.

[65] C. Stone. *A Course in Probability and Statistics.* Duxbury Press, 1995.

[66] M. Sviridenko. A note on maximizing a submodular set function subject to a knapsack constraint. *Oper. Res. Lett.*, 32(1):41–43, 2004.

[67] K. Talwar. The price of truth: Frugality in truthful mechanisms. In *STACS*, pages 608–619, 2003.

[68] W. Vickrey. Counterspeculation, auctions, and competitive sealed tenders. *journal of finance*, 16(1):8–37, 1961.

[69] Q. Yan. On the price of truthfulness in path auctions. In *WINE*, pages 584–589, 2007.