Frugality Ratios And Improved Truthful Mechanisms for Vertex Cover *

Edith Elkind
Hebrew University of Jerusalem, Israel, and University of Southampton, Southampton, SO17 1BJ, U.K.

Leslie Ann Goldberg
University of Liverpool
Liverpool L69 3BX, U.K.

Paul Goldberg
University of Liverpool
Liverpool L69 3BX, U.K.

ABSTRACT

In *set-system auctions*, there are several overlapping teams of agents, and a task that can be completed by any of these teams. The auctioneer's goal is to hire a team and pay as little as possible. Examples of this setting include shortest-path auctions and vertex-cover auctions. Recently, Karlin, Kempe and Tamir introduced a new definition of *frugality ratio* for this problem. Informally, the "frugality ratio" is the ratio of the total payment of a mechanism to a desired payment bound. The ratio captures the extent to which the mechanism overpays, relative to perceived fair cost in a truthful auction. In this paper, we propose a new truthful polynomial-time auction for the vertex cover problem and bound its frugality ratio. We show that the solution quality is with a constant factor of optimal and the frugality ratio is within a constant factor of the best possible worst-case bound; this is the first auction for this problem to have these properties. Moreover, we show how to transform any truthful auction into a frugal one while preserving the approximation ratio. Also, we consider two natural modifications of the definition of Karlin et al., and we analyse the properties of the resulting payment bounds, such as monotonicity, computational hardness, and robustness with respect to the draw-resolution rule. We study the relationships between the different payment bounds, both for general set systems and for specific set-system auctions, such as path auctions and vertex-cover auctions. We use these new definitions in the proof of our main result for vertex-cover auctions via a bootstrapping technique, which may be of independent interest.

Categories and Subject Descriptors

F.2 [**Theory of Computation**]: Analysis of Algorithms and Problem Complexity; J.4 [**Computer Applications**]: Social and Behavioral Sciences—*economics*

General Terms

Algorithms, Economics, Theory

*This research is supported by the EPSRC research grants "Algorithmics of Network-sharing Games" and "Discontinuous Behaviour in the Complexity of randomized Algorithms".

EC'07, June 13–16, 2007, San Diego, California, USA.

Keywords

Auctions, Frugality, Vertex Cover

1. INTRODUCTION

In a *set system auction* there is a single buyer and many vendors that can provide various services. It is assumed that the buyer's requirements can be satisfied by various subsets of the vendors; these subsets are called the *feasible sets*. A widely-studied class of set-system auctions is *path auctions*, where each vendor is able to sell access to a link in a network, and the feasible sets are those sets whose links contain a path from a given source to a given destination; the study of these auctions has been initiated in the seminal paper by Nisan and Ronen [19] (see also [1, 10, 9, 6, 15, 7, 20]).

We assume that each vendor has a cost of providing his services, but submits a possibly larger *bid* to the auctioneer. Based on these bids, the auctioneer selects a feasible subset of vendors, and makes payments to the vendors in this subset. Each selected vendor enjoys a profit of payment minus cost. Vendors want to maximise profit, while the buyer wants to minimise the amount he pays. A natural goal in this setting is to design a *truthful* auction, in which vendors have an incentive to bid their true cost. This can be achieved by paying each selected vendor a premium above her bid in such a way that the vendor has no incentive to overbid. An interesting question in mechanism design is how much the auctioneer will have to overpay in order to ensure truthful bids.

In the context of path auctions this topic was first addressed by Archer and Tardos [1]. They define the *frugality ratio* of a mechanism as the ratio between its total payment and the cost of the cheapest path disjoint from the path selected by the mechanism. They show that, for a large class of truthful mechanisms for this problem, the frugality ratio is as large as the number of edges in the shortest path. Talwar [21] extends this definition of frugality ratio to general set systems, and studies the frugality ratio of the classical VCG mechanism [22, 4, 14] for many specific set systems, such as minimum spanning trees and set covers.

While the definition of frugality ratio proposed by [1] is well-motivated and has been instrumental in studying truthful mechanisms for set systems, it is not completely satisfactory. Consider, for example, the graph of Figure 1 with the costs $c_{AB} = c_{BC} =$

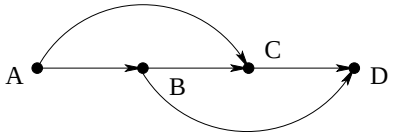

Figure 1: The diamond graph

$c_{CD} = 0$, $c_{AC} = c_{BD} = 1$. This graph is 2-connected and the VCG payment to the winning path ABCD is bounded. However, the graph contains no A–D path that is disjoint from ABCD, and hence the frugality ratio of VCG on this graph remains undefined. At the same time, there is no *monopoly*, that is, there is no vendor that appears in all feasible sets. In auctions for other types of set systems, the requirement that there exist a feasible solution disjoint from the selected one is even more severe: for example, for vertex-cover auctions (where vendors correspond to the vertices of some underlying graph, and the feasible sets are vertex covers) the requirement means that the graph must be bipartite. To deal with this problem, Karlin et al. [16] suggest a better benchmark, which is defined for any monopoly-free set system. This quantity, which they denote by ν, intuitively corresponds to the value of a cheapest Nash equilibrium. Based on this new definition, the authors construct new mechanisms for the shortest path problem and show that the overpayment of these mechanisms is within a constant factor of optimal.

1.1 Our results

Vertex cover auctions We propose a truthful polynomial-time auction for vertex cover that outputs a solution whose cost is within a factor of 2 of optimal, and whose frugality ratio is at most 2Δ, where Δ is the maximum degree of the graph (Theorem 4). We complement this result by proving (Theorem 5) that for any Δ and n, there are graphs of maximum degree Δ and size $\Theta(n)$ for which *any* truthful mechanism has frugality ratio at least $\Delta/2$. This means that the solution quality of our auction is with a factor of 2 of optimal and the frugality ratio is within a factor of 4 of the best possible bound for worst-case inputs. To the best of our knowledge, this is the first auction for this problem that enjoys these properties. Moreover, we show how to transform any truthful mechanism for the vertex-cover problem into a frugal one while preserving the approximation ratio.

Frugality ratios Our vertex cover results naturally suggest two modifications of the definition of ν in [16]. These modifications can be made independently of each other, resulting in four different payment bounds TUmax, TUmin, NTUmax, and NTUmin, where NTUmin is equal to the original payment bound ν of in [16]. All four payment bounds arise as Nash equilibria of certain games (see the full version of this paper [8]); the differences between them can be seen as "the price of initiative" and "the price of cooperation" (see Section 3). While our main result about vertex cover auctions (Theorem 4) is with respect to $\mathrm{NTUmin} = \nu$, we make use of the new definitions by first comparing the payment of our mechanism to a weaker bound NTUmax, and then bootstrapping from this result to obtain the desired bound.

Inspired by this application, we embark on a further study of these payment bounds. Our results here are as follows:

1. We observe (Proposition 1) that the four payment bounds always obey a particular order that is independent of the choice of the set system and the cost vector, namely, $\mathrm{TUmin} \leq \mathrm{NTUmin} \leq \mathrm{NTUmax} \leq \mathrm{TUmax}$. We provide examples (Proposition 5 and Corollaries 1 and 2) showing that for the vertex cover problem any two consecutive bounds can differ by a factor of $n-2$, where n is the number of agents. We then show (Theorem 2) that this separation is almost best possible for general set systems by proving that for any set system $\mathrm{TUmax}/\mathrm{TUmin} \leq n$. In contrast, we demonstrate (Theorem 3) that for path auctions $\mathrm{TUmax}/\mathrm{TUmin} \leq 2$. We provide examples (Propositions 2, 3 and 4) showing that this bound is tight. We see this as an argument for the study of vertex-cover auctions, as they appear to be more representative of the general team -selection problem than the widely studied path auctions.

2. We show (Theorem 1) that for any set system, if there is a cost vector for which TUmin and NTUmin differ by a factor of α, there is another cost vector that separates NTUmin and NTUmax by the same factor and vice versa; the same is true for the pairs $(\mathrm{NTUmin}, \mathrm{NTUmax})$ and $(\mathrm{NTUmax}, \mathrm{TUmax})$. This symmetry is quite surprising, since, e.g., TUmin and NTUmax are obtained from NTUmin by two very different transformations. This observation suggests that the four payment bounds should be studied in a unified framework; moreover, it leads us to believe that the bootstrapping technique of Theorem 4 may have other applications.

3. We evaluate the payment bounds introduced here with respect to a checklist of desirable features. In particular, we note that the payment bound $\nu = \mathrm{NTUmin}$ of [16] exhibits some counterintuitive properties, such as nonmonotonicity with respect to adding a new feasible set (Proposition 7), and is NP-hard to compute (Theorem 6), while some of the other payment bounds do not suffer from these problems. This can be seen as an argument in favour of using weaker but efficiently computable bounds NTUmax and TUmax.

Related work

Vertex-cover auctions have been studied in the past by Talwar [21] and Calinescu [5]. Both of these papers are based on the definition of frugality ratio used in [1]; as mentioned before, this means that their results only apply to bipartite graphs. Talwar [21] shows that the frugality ratio of VCG is at most Δ. However, since finding the cheapest vertex cover is an NP-hard problem, the VCG mechanism is computationally infeasible. The first (and, to the best of our knowledge, only) paper to investigate polynomial-time truthful mechanisms for vertex cover is [5]. This paper studies an auction that is based on the greedy allocation algorithm, which has an approximation ratio of $\log n$. While the main focus of [5] is the more general set cover problem, the results of [5] imply a frugality ratio of $2\Delta^2$ for vertex cover. Our results improve on those of [21] as our mechanism is polynomial-time computable, as well as on those of [5], as our mechanism has a better approximation ratio, and we prove a stronger bound on the frugality ratio; moreover, this bound also applies to the mechanism of [5].

2. PRELIMINARIES

In most of this paper, we discuss auctions for set systems. A *set system* is a pair $(\mathcal{E}, \mathcal{F})$, where $\mathcal{E}$ is the *ground set*, $|\mathcal{E}| = n$, and $\mathcal{F}$ is a collection of *feasible sets*, which are subsets of $\mathcal{E}$. Two particular types of set systems are of interest to us — *shortest path* systems, in which the ground set consists of all edges of a network, and the feasible sets are paths between two specified vertices s and t, and *vertex cover* systems, in which the elements of the ground set are the vertices of a graph, and the feasible sets are vertex covers of this graph.

In set system auctions, each element e of the ground set is owned by an independent agent and has an associated non-negative cost c_e. The goal of the centre is to select (purchase) a feasible set. Each element e in the selected set incurs a cost of c_e. The elements that are not selected incur no costs.

The auction proceeds as follows: all elements of the ground set make their bids, the centre selects a feasible set based on the bids and makes payments to the agents. Formally, an auction is defined by an *allocation rule* $A : \mathbf{R}^n \mapsto \mathcal{F}$ and a *payment rule* $P : \mathbf{R}^n \mapsto \mathbf{R}^n$. The allocation rule takes as input a vector of bids and decides which of the sets in $\mathcal{F}$ should be selected. The payment rule also takes as input a vector of bids and decides how much to pay to each agent. The standard requirements are *individual rationality*, i.e., the payment to each agent should be at least as high as his incurred cost (0 for agents not in the selected set and c_e for agents in the

selected set) and *incentive compatibility*, or *truthfulness*, i.e., each agent's dominant strategy is to bid his true cost.

An allocation rule is *monotone* if an agent cannot increase his chance of getting selected by raising his bid. Formally, for any bid vector $\mathbf{b}$ and any $e \in \mathcal{E}$, if $e \notin A(\mathbf{b})$ then $e \notin A(b_1, \ldots, b'_e, \ldots, b_n)$ for any $b'_e > b_e$. Given a monotone allocation rule A and a bid vector $\mathbf{b}$, the *threshold bid* t_e of an agent $e \in A(\mathbf{b})$ is the highest bid of this agent that still wins the auction, given that the bids of other participants remain the same. Formally, $t_e = \sup\{b'_e \in \mathbf{R} \mid e \in A(b_1, \ldots, b'_e, \ldots, b_n)\}$. It is well known (see, e.g. [19, 13]) that any auction that has a monotone allocation rule and pays each agent his threshold bid is truthful; conversely, any truthful auction has a monotone allocation rule.

The VCG mechanism is a truthful mechanism that maximises the "social welfare" and pays 0 to the losing agents. For set system auctions, this simply means picking a cheapest feasible set, paying each agent in the selected set his threshold bid, and paying 0 to all other agents. Note, however, that the VCG mechanism may be difficult to implement, since finding a cheapest feasible set may be intractable.

If U is a set of agents, $c(U)$ denotes $\sum_{w \in U} c_w$. Similarly, $b(U)$ denotes $\sum_{w \in U} b_w$.

3. FRUGALITY RATIOS

We start by reproducing the definition of the quantity ν from [16, Definition 4].

Let $(\mathcal{E}, \mathcal{F})$ be a set system and let S be a cheapest feasible set with respect to the true costs c_e. Then $\nu(\mathbf{c}, S)$ is the solution to the following optimisation problem.

Minimise $B = \sum_{e \in S} b_e$ subject to

(1) $b_e \geq c_e$ for all $e \in \mathcal{E}$

(2) $\sum_{e \in S \setminus T} b_e \leq \sum_{e \in T \setminus S} c_e$ for all $T \in \mathcal{F}$

(3) for every $e \in S$, there is a $T_e \in \mathcal{F}$ such that $e \notin T_e$ and $\sum_{e' \in S \setminus T_e} b_{e'} = \sum_{e' \in T_e \setminus S} c_{e'}$

The bound $\nu(\mathbf{c}, S)$ can be seen as an outcome of a two-stage process, where first each agent $e \in S$ makes a bid b_e stating how much it wants to be paid, and then the centre decides whether to accept these bids. The behaviour of both parties is affected by the following considerations. From the centre's point of view, the set S must remain the most attractive choice, i.e., it must be among the cheapest feasible sets under the new costs $c'_e = c_e$ for $e \notin S$, $c'_e = b_e$ for $e \in S$ (condition (2)). The reason for that is that if (2) is violated for some set T, the centre would prefer T to S. On the other hand, no agent would agree to a payment that does not cover his costs (condition (1)), and moreover, each agent tries to maximise his profit by bidding as high as possible, i.e., none of the agents can increase his bid without violating condition (2) (condition (3)). The centre wants to minimise the total payout, so $\nu(\mathbf{c}, S)$ corresponds to the best possible outcome from the centre's point of view.

This definition captures many important aspects of our intuition about 'fair' payments. However, it can be modified in two ways, both of which are still quite natural, but result in different payment bounds.

First, we can consider the worst rather than the best possible outcome for the centre. That is, we can consider the maximum total payment that the agents can extract by jointly selecting their bids subject to (1), (2), and (3). Such a bound corresponds to maximising B subject to (1), (2), and (3) rather than minimising it. If it is the agents who make the original bids (rather than the centre), this kind of bidding behaviour is plausible. On the other hand, in a game in which the centre proposes payments to the agents in S and the agents accept them as long as (1), (2) and (3) are satisfied, we would be likely to observe a total payment of $\nu(\mathbf{c}, S)$. Hence, the difference between these two definitions can be seen as "the price of initiative".

Second, the agents may be able to make payments to each other. In this case, if they can extract more money from the centre by agreeing on a vector of bids that violates individual rationality (i.e., condition (1)) for some bidders, they might be willing to do so, as the agents who are paid below their costs will be compensated by other members of the group. The bids must still be realistic, i.e., they have to satisfy $b_e \geq 0$. The resulting change in payments can be seen as "the price of co-operation" and corresponds to replacing condition (1) with the following weaker condition (1^*):

$$b_e \geq 0 \text{ for all } e \in \mathcal{E}. \qquad (1^*)$$

By considering all possible combinations of these modifications, we obtain four different payment bounds, namely

- $\mathrm{TUmin}(\mathbf{c}, S)$, which is the solution to the optimisation problem "Minimise B" subject to (1^*), (2), and (3).
- $\mathrm{TUmax}(\mathbf{c}, S)$, which is the solution to the optimisation problem "Maximise B" subject to (1^*), (2), and (3).
- $\mathrm{NTUmin}(\mathbf{c}, S)$, which is the solution to the optimisation problem "Minimise B" subject to (1), (2), and (3).
- $\mathrm{NTUmax}(\mathbf{c}, S)$, which is the solution to the optimisation problem "Maximise B" subject to (1), (2), (3).

The abbreviations TU and NTU correspond, respectively, to transferable utility and non-transferable utility, i.e., the agents' ability/inability to make payments to each other. For concreteness, we will take $\mathrm{TUmin}(\mathbf{c})$ to be $\mathrm{TUmin}(\mathbf{c}, S)$ where S is the lexicographically least amongst the cheapest feasible sets. We define $\mathrm{TUmax}(\mathbf{c})$, $\mathrm{NTUmin}(\mathbf{c})$, $\mathrm{NTUmax}(\mathbf{c})$ and $\nu(\mathbf{c})$ similarly, though we will see in Section 6.3 that, in fact, $\mathrm{NTUmin}(\mathbf{c}, S)$ and $\mathrm{NTUmax}(\mathbf{c}, S)$ are independent of the choice of S. Note that the quantity $\nu(\mathbf{c})$ from [16] is $\mathrm{NTUmin}(\mathbf{c})$.

The second modification (transferable utility) is more intuitively appealing in the context of the maximisation problem, as both assume some degree of co-operation between the agents. While the second modification can be made without the first, the resulting payment bound $\mathrm{TUmin}(\mathbf{c}, S)$ is too strong to be a realistic benchmark, at least for general set systems. In particular, it can be smaller than the total cost of the cheapest feasible set S (see Section 6). Nevertheless, we provide the definition as well as some results about $\mathrm{TUmin}(\mathbf{c}, S)$ in the paper, both for completeness and because we believe that it may help to understand which properties of the payment bounds are important for our proofs. Another possibility would be to introduce an additional constraint $\sum_{e \in S} b_e \geq \sum_{e \in S} c_e$ in the definition of $\mathrm{TUmin}(\mathbf{c}, S)$ (note that this condition holds automatically for $\mathrm{TUmax}(\mathbf{c}, S)$, as $\mathrm{TUmax}(\mathbf{c}, S) \geq \mathrm{NTUmax}(\mathbf{c}, S)$); however, such a definition would have no direct game-theoretic interpretation, and some of our results (in particular, the ones in Section 4) would no longer be true.

REMARK 1. *For the payment bounds that are derived from maximisation problems, (i.e.,* $\mathrm{TUmax}(\mathbf{c}, S)$ *and* $\mathrm{NTUmax}(\mathbf{c}, S)$*), constraints of type (3) are redundant and can be dropped. Hence,* $\mathrm{TUmax}(\mathbf{c}, S)$ *and* $\mathrm{NTUmax}(\mathbf{c}, S)$ *are solutions to linear programs, and therefore can be computed in polynomial time as long as we have a separation oracle for constraints in (2). In contrast,*

$\mathrm{NTUmin}(\mathbf{c}, S)$ *can be NP-hard to compute even if the size of* $\mathcal{F}$ *is polynomial (see Section 6).*

The first and third inequalities in the following observation follow from the fact that condition (1^*) is strictly weaker than condition (1).

PROPOSITION 1.

$$\mathrm{TUmin}(\mathbf{c}, S) \leq \mathrm{NTUmin}(\mathbf{c}, S) \leq \mathrm{NTUmax}(\mathbf{c}, S) \leq \mathrm{TUmax}(\mathbf{c}, S).$$

Let $\mathcal{M}$ be a truthful mechanism for $(\mathcal{E}, \mathcal{F})$. Let $p_{\mathcal{M}}(\mathbf{c})$ denote the total payments of $\mathcal{M}$ when the actual costs are $\mathbf{c}$. A *frugality ratio* of $\mathcal{M}$ with respect to a payment bound is the ratio between the payment of $\mathcal{M}$ and this payment bound. In particular,

$$\phi_{\mathrm{TUmin}}(\mathcal{M}) = \sup_{\mathbf{c}} p_{\mathcal{M}}(\mathbf{c})/\mathrm{TUmin}(\mathbf{c}),$$
$$\phi_{\mathrm{TUmax}}(\mathcal{M}) = \sup_{\mathbf{c}} p_{\mathcal{M}}(\mathbf{c})/\mathrm{TUmax}(\mathbf{c}),$$
$$\phi_{\mathrm{NTUmin}}(\mathcal{M}) = \sup_{\mathbf{c}} p_{\mathcal{M}}(\mathbf{c})/\mathrm{NTUmin}(\mathbf{c}),$$
$$\phi_{\mathrm{NTUmax}}(\mathcal{M}) = \sup_{\mathbf{c}} p_{\mathcal{M}}(\mathbf{c})/\mathrm{NTUmax}(\mathbf{c}).$$

We conclude this section by showing that there exist set systems and respective cost vectors for which all four payment bounds are different. In the next section, we quantify this difference, both for general set systems, and for specific types of set systems, such as path auctions or vertex cover auctions.

EXAMPLE 1. *Consider the shortest-path auction on the graph of Figure 1. The cheapest feasible sets are all paths from* A *to* D*. It can be verified, using the reasoning of Propositions 2 and 3 below, that for the cost vector* $c_{AB} = c_{CD} = 2$, $c_{BC} = 1$, $c_{AC} = c_{BD} = 5$*, we have*

- $\mathrm{TUmax}(\mathbf{c}) = 10$ *(with* $b_{AB} = b_{CD} = 5$, $b_{BC} = 0$*),*
- $\mathrm{NTUmax}(\mathbf{c}) = 9$ *(with* $b_{AB} = b_{CD} = 4$, $b_{BC} = 1$*),*
- $\mathrm{NTUmin}(\mathbf{c}) = 7$ *(with* $b_{AB} = b_{CD} = 2$, $b_{BC} = 3$*),*
- $\mathrm{TUmin}(\mathbf{c}) = 5$ *(with* $b_{AB} = b_{CD} = 0$, $b_{BC} = 5$*).*

4. COMPARING PAYMENT BOUNDS

4.1 Path auctions

We start by showing that for path auctions any two consecutive payment bounds can differ by at least a factor of 2.

PROPOSITION 2. *There is an instance of the shortest-path problem for which we have* $\mathrm{NTUmax}(\mathbf{c})/\mathrm{NTUmin}(\mathbf{c}) \geq 2$.

PROOF. This construction is due to David Kempe [17]. Consider the graph of Figure 1 with the edge costs $c_{AB} = c_{BC} = c_{CD} = 0$, $c_{AC} = c_{BD} = 1$. Under these costs, ABCD is the cheapest path. The inequalities in (2) are $b_{AB} + b_{BC} \leq c_{AC} = 1$, $b_{BC} + b_{CD} \leq c_{BD} = 1$. By condition (3), both of these inequalities must be tight (the former one is the only inequality involving b_{AB}, and the latter one is the only inequality involving b_{CD}). The inequalities in (1) are $b_{AB} \geq 0$, $b_{BC} \geq 0$, $b_{CD} \geq 0$. Now, if the goal is to maximise $b_{AB} + b_{BC} + b_{CD}$, the best choice is $b_{AB} = b_{CD} = 1$, $b_{BC} = 0$, so $\mathrm{NTUmax}(\mathbf{c}) = 2$. On the other hand, if the goal is to minimise $b_{AB} + b_{BC} + b_{CD}$, one should set $b_{AB} = b_{CD} = 0$, $b_{BC} = 1$, so $\mathrm{NTUmin}(\mathbf{c}) = 1$. □

PROPOSITION 3. *There is an instance of the shortest-path problem for which we have* $\mathrm{TUmax}(\mathbf{c})/\mathrm{NTUmax}(\mathbf{c}) \geq 2$.

PROOF. Again, consider the graph of Figure 1. Let the edge costs be $c_{AB} = c_{CD} = 0$, $c_{BC} = 1$, $c_{AC} = c_{BD} = 1$. ABCD is the lexicographically-least cheapest path, so we can assume that $S = \{AB, BC, CD\}$. The inequalities in (2) are the same as in the previous example, and by the same argument both of them are, in fact, equalities. The inequalities in (1) are $b_{AB} \geq 0$, $b_{BC} \geq 1$, $b_{CD} \geq 0$. Our goal is to maximise $b_{AB} + b_{BC} + b_{CD}$. If we have to respect the inequalities in (1), we have to set $b_{AB} = b_{CD} = 0$, $b_{BC} = 1$, so $\mathrm{NTUmax}(\mathbf{c}) = 1$. Otherwise, we can set $b_{AB} = b_{CD} = 1$, $b_{BC} = 0$, so $\mathrm{TUmax}(\mathbf{c}) \geq 2$. □

PROPOSITION 4. *There is an instance of the shortest-path problem for which we have* $\mathrm{NTUmin}(\mathbf{c})/\mathrm{TUmin}(\mathbf{c}) \geq 2$.

PROOF. This construction is also based on the graph of Figure 1. The edge costs are $c_{AB} = c_{CD} = 1$, $c_{BC} = 0$, $c_{AC} = c_{BD} = 1$. ABCD is the lexicographically least cheapest path, so we can assume that $S = \{AB, BC, CD\}$. Again, the inequalities in (2) are the same, and both are, in fact, equalities. The inequalities in (1) are $b_{AB} \geq 1$, $b_{BC} \geq 0$, $b_{CD} \geq 1$. Our goal is to minimise $b_{AB} + b_{BC} + b_{CD}$. If we have to respect the inequalities in (1), we have to set $b_{AB} = b_{CD} = 1$, $b_{BC} = 0$, so $\mathrm{NTUmin}(\mathbf{c}) = 2$. Otherwise, we can set $b_{AB} = b_{CD} = 0$, $b_{BC} = 1$, so $\mathrm{TUmin}(\mathbf{c}) \leq 1$. □

In Section 4.4 (Theorem 3), we show that the separation results in Propositions 2, 3, and 4 are optimal.

4.2 Connections between separation results

The separation results for path auctions are obtained on the same graph using very similar cost vectors. It turns out that this is not coincidental. Namely, we can prove the following theorem.

THEOREM 1. *For any set system* $(\mathcal{E}, \mathcal{F})$*, and any feasible set* S*,*

$$\max_{\mathbf{c}} \frac{\mathrm{TUmax}(\mathbf{c}, S)}{\mathrm{NTUmax}(\mathbf{c}, S)} = \max_{\mathbf{c}} \frac{\mathrm{NTUmax}(\mathbf{c}, S)}{\mathrm{NTUmin}(\mathbf{c}, S)},$$
$$\max_{\mathbf{c}} \frac{\mathrm{NTUmax}(\mathbf{c}, S)}{\mathrm{NTUmin}(\mathbf{c}, S)} = \max_{\mathbf{c}} \frac{\mathrm{NTUmin}(\mathbf{c}, S)}{\mathrm{TUmin}(\mathbf{c}, S)},$$

where the maximum is over all cost vectors c *for which* S *is a cheapest feasible set.*

The proof of the theorem follows directly from the four lemmas proved below; more precisely, the first equality in Theorem 1 is obtained by combining Lemmas 1 and 2, and the second equality is obtained by combining Lemmas 3 and 4. We prove Lemma 1 here; the proofs of Lemmas 2– 4 are similar and can be found in the full version of this paper [8].

LEMMA 1. *Suppose that* $\mathbf{c}$ *is a cost vector for* $(\mathcal{E}, \mathcal{F})$ *such that* S *is a cheapest feasible set and* $\mathrm{TUmax}(\mathbf{c}, S)/\mathrm{NTUmax}(\mathbf{c}, S) = \alpha$*. Then there is a cost vector* $\mathbf{c}'$ *such that* S *is a cheapest feasible set and* $\mathrm{NTUmax}(\mathbf{c}', S)/\mathrm{NTUmin}(\mathbf{c}', S) \geq \alpha$.

PROOF. Suppose that $\mathrm{TUmax}(\mathbf{c}, S) = X$ and $\mathrm{NTUmax}(\mathbf{c}, S) = Y$ where $X/Y = \alpha$. Assume without loss of generality that S consists of elements $1, \ldots, k$, and let $\mathbf{b}^1 = (b_1^1, \ldots, b_k^1)$ and $\mathbf{b}^2 = (b_1^2, \ldots, b_k^2)$ be the bid vectors that correspond to $\mathrm{TUmax}(\mathbf{c}, S)$ and $\mathrm{NTUmax}(\mathbf{c}, S)$, respectively.

Construct the cost vector $\mathbf{c}'$ by setting $c_i' = c_i$ for $i \notin S$, $c_i' = \min\{c_i, b_i^1\}$ for $i \in S$. Clearly, S is a cheapest set under $\mathbf{c}'$. Moreover, as the costs of elements outside of S remained the same, the right-hand sides of all constraints in (2) did not change, so any bid vector that satisfies (2) and (3) with respect to $\mathbf{c}$, also satisfies them with respect to $\mathbf{c}'$. We will construct two bid vectors $\mathbf{b}^3$ and $\mathbf{b}^4$ that satisfy conditions (1), (2), and (3) for the cost vector $\mathbf{c}'$, and

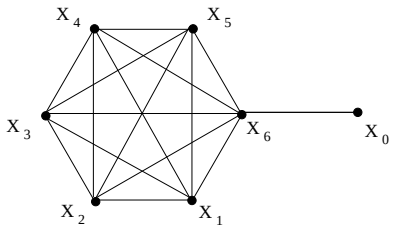

Figure 2: Graph that separates payment bounds for vertex cover, $n = 7$

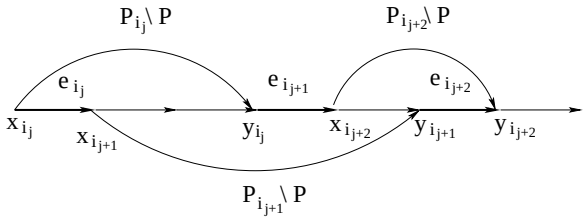

Figure 3: Proof of Theorem 3: constraints for $\hat{P}_{i_j}$ and $\hat{P}_{i_{j+2}}$ do not overlap

have $\sum_{i\in S} b_i^3 = X$, $\sum_{i\in S} b_i^4 = Y$. As $\mathrm{NTUmax}(\mathbf{c}', S) \geq X$ and $\mathrm{NTUmin}(\mathbf{c}', S) \leq Y$, this implies the lemma.

We can set $b_i^3 = b_i^1$: this bid vector satisfies conditions (2) and (3) since $\mathbf{b}^1$ does, and we have $b_i^1 \geq \min\{c_i, b_i^1\} = c_i'$, which means that $\mathbf{b}^3$ satisfies condition (1). Furthermore, we can set $b_i^4 = b_i^2$. Again, $\mathbf{b}^4$ satisfies conditions (2) and (3) since $\mathbf{b}^2$ does, and since $\mathbf{b}^2$ satisfies condition (1), we have $b_i^2 \geq c_i \geq c_i'$, which means that $\mathbf{b}^4$ satisfies condition (1). □

LEMMA 2. *Suppose* $\mathbf{c}$ *is a cost vector for* $(\mathcal{E}, \mathcal{F})$ *such that* S *is a cheapest feasible set and* $\mathrm{NTUmax}(\mathbf{c}, S)/\mathrm{NTUmin}(\mathbf{c}, S) = \alpha$. *Then there is a cost vector* $\mathbf{c}'$ *such that* S *is a cheapest feasible set and* $\mathrm{TUmax}(\mathbf{c}', S)/\mathrm{NTUmax}(\mathbf{c}', S) \geq \alpha$.

LEMMA 3. *Suppose that* $\mathbf{c}$ *is a cost vector for* $(\mathcal{E}, \mathcal{F})$ *such that* S *is a cheapest feasible set and* $\mathrm{NTUmax}(\mathbf{c}, S)/\mathrm{NTUmin}(\mathbf{c}, S) = \alpha$. *Then there is a cost vector* $\mathbf{c}'$ *such that* S *is a cheapest feasible set and* $\mathrm{NTUmin}(\mathbf{c}', S)/\mathrm{TUmin}(\mathbf{c}', S) \geq \alpha$.

LEMMA 4. *Suppose that* $\mathbf{c}$ *is a cost vector for* $(\mathcal{E}, \mathcal{F})$ *such that* S *is a cheapest feasible set and* $\mathrm{NTUmin}(\mathbf{c}, S)/\mathrm{TUmin}(\mathbf{c}, S) = \alpha$. *Then there is a cost vector* $\mathbf{c}'$ *such that* S *is a cheapest feasible set and* $\mathrm{NTUmax}(\mathbf{c}', S)/\mathrm{NTUmin}(\mathbf{c}', S) \geq \alpha$.

4.3 Vertex-cover auctions

In contrast to the case of path auctions, for vertex-cover auctions the gap between $\mathrm{NTUmin}(\mathbf{c})$ and $\mathrm{NTUmax}(\mathbf{c})$ (and hence between $\mathrm{NTUmax}(\mathbf{c})$ and $\mathrm{TUmax}(\mathbf{c})$, and between $\mathrm{TUmin}(\mathbf{c})$ and $\mathrm{NTUmin}(\mathbf{c})$) can be proportional to the size of the graph.

PROPOSITION 5. *For any* $n \geq 3$, *there is a an* n*-vertex graph and a cost vector* $\mathbf{c}$ *for which* $\mathrm{TUmax}(\mathbf{c})/\mathrm{NTUmax}(\mathbf{c}) \geq n - 2$.

PROOF. The underlying graph consists of an $(n-1)$-clique on the vertices $X_1, \ldots, X_{n-1}$, and an extra vertex X_0 adjacent to X_{n-1}. The costs are $c_{X_1} = c_{X_2} = \cdots = c_{X_{n-2}} = 0$, $c_{X_0} = c_{X_{n-1}} = 1$. We can assume that $S = \{X_0, X_1, \ldots, X_{n-2}\}$ (this is the lexicographically first vertex cover of cost 1). For this set system, the constraints in (2) are $b_{X_i} + b_{X_0} \leq c_{X_{n-1}} = 1$ for $i = 1, \ldots, n-2$. Clearly, we can satisfy conditions (2) and (3) by setting $b_{X_i} = 1$ for $i = 1, \ldots, n-2$, $b_{X_0} = 0$. Hence, $\mathrm{TUmax}(\mathbf{c}) \geq n - 2$. For $\mathrm{NTUmax}(\mathbf{c})$, there is an additional constraint $b_{X_0} \geq 1$, so the best we can do is to set $b_{X_i} = 0$ for $i = 1, \ldots, n-2$, $b_{X_0} = 1$, which implies $\mathrm{NTUmax}(\mathbf{c}) = 1$. □

Combining Proposition 5 with Lemmas 1 and 3, we derive the following corollaries.

COROLLARY 1. *For any* $n \geq 3$, *we can construct an instance of the vertex cover problem on a graph of size* n *that satisfies* $\mathrm{NTUmax}(\mathbf{c})/\mathrm{NTUmin}(\mathbf{c}) \geq n - 2$.

COROLLARY 2. *For any* $n \geq 3$, *we can construct an instance of the vertex cover problem on a graph of size* n *that satisfies* $\mathrm{NTUmin}(\mathbf{c})/\mathrm{TUmin}(\mathbf{c}) \geq n - 2$.

4.4 Upper bounds

It turns out that the lower bound proved in the previous subsection is almost tight. More precisely, the following theorem shows that no two payment bounds can differ by more than a factor of n; moreover, this is the case not just for the vertex cover problem, but for general set systems. We bound the gap between $\mathrm{TUmax}(\mathbf{c})$ and $\mathrm{TUmin}(\mathbf{c})$. Since $\mathrm{TUmin}(\mathbf{c}) \leq \mathrm{NTUmin}(\mathbf{c}) \leq \mathrm{NTUmax}(\mathbf{c}) \leq \mathrm{TUmax}(\mathbf{c})$, this bound applies to any pair of payment bounds.

THEOREM 2. *For any set system* $(\mathcal{E}, \mathcal{F})$ *and any cost vector* $\mathbf{c}$, *we have* $\mathrm{TUmax}(\mathbf{c})/\mathrm{TUmin}(\mathbf{c}) \leq n$.

PROOF. Assume wlog that the winning set S consists of elements $1, \ldots, k$. Let $c_1, \ldots, c_k$ be the true costs of elements in S, let $b_1', \ldots, b_k'$ be their bids that correspond to $\mathrm{TUmin}(\mathbf{c})$, and let $b_1'', \ldots, b_k''$ be their bids that correspond to $\mathrm{TUmax}(\mathbf{c})$.

Consider the conditions (2) and (3) for S. One can pick a subset $\mathcal{L}$ of at most k inequalities in (2) so that for each $i = 1, \ldots, k$ there is at least one inequality in $\mathcal{L}$ that is tight for b_i'. Suppose that the jth inequality in $\mathcal{L}$ is of the form $b_{i_1} + \cdots + b_{i_t} \leq c(T_j \setminus S)$. For b_i', all inequalities in $\mathcal{L}$ are, in fact, equalities. Hence, by adding up all of them we obtain $k \sum_{i=1,\ldots,k} b_i' \geq \sum_{j=1,\ldots,k} c(T_j \setminus S)$. On the other hand, all these inequalities appear in condition (2), so they must hold for b_i'', i.e., $\sum_{i=1,\ldots,k} b_i'' \leq \sum_{j=1,\ldots,k} c(T_j \setminus S)$. Combining these two inequalities, we obtain

$$n\,\mathrm{TUmin}(\mathbf{c}) \geq k\,\mathrm{TUmin}(\mathbf{c}) \geq \mathrm{TUmax}(\mathbf{c}).$$

□

REMARK 2. *The final line of the proof of Theorem 2 shows that, in fact, the upper bound on* $\mathrm{TUmax}(\mathbf{c})/\mathrm{TUmin}(\mathbf{c})$ *can be strengthened to the size of the winning set,* k. *Note that in Proposition 5, as well as in Corollaries 1 and 2,* $k = n - 1$, *so these results do not contradict each other.*

For path auctions, this upper bound can be improved to 2, matching the lower bounds of Section 4.1.

THEOREM 3. *For any instance of the shortest path problem,* $\mathrm{TUmax}(\mathbf{c}) \leq 2\,\mathrm{TUmin}(\mathbf{c})$.

PROOF. Given a network (G, s, t), assume without loss of generality that the lexicographically-least cheapest s–t path, P, in G is $\{e_1, \ldots, e_k\}$, where $e_1 = (s, v_1), e_2 = (v_1, v_2), \ldots, e_k = (v_{k-1}, t)$. Let $c_1, \ldots, c_k$ be the true costs of $e_1, \ldots, e_k$, and let $\mathbf{b}' = (b_1', \ldots, b_k')$ and $\mathbf{b}'' = (b_1'', \ldots, b_k'')$ be bid vectors that correspond to $\mathrm{TUmin}(\mathbf{c})$ and $\mathrm{TUmax}(\mathbf{c})$, respectively.

For any $i = 1, \ldots, k$, there is a constraint in (2) that is tight for b_i' with respect to the bid vector $\mathbf{b}'$, i.e., an s–t path P_i that avoids e_i and satisfies $b'(P \setminus P_i) = c(P_i \setminus P)$. We can assume without loss of generality that P_i coincides with P up to some vertex x_i, then deviates from P to avoid e_i, and finally returns to P at a vertex

y_i and coincides with P from then on (clearly, it might happen that $s = x_i$ or $t = y_i$). Indeed, if P_i deviates from P more than once, one of these deviations is not necessary to avoid e_i and can be replaced with the respective segment of P without increasing the cost of P_i. Among all paths of this form, let $\hat{P}_i$ be the one with the largest value of y_i, i.e., the "rightmost" one. This path corresponds to an inequality I_i of the form $b'_{x_i+1} + \cdots + b'_{y_i} \leq c(\hat{P}_i \setminus P)$.

As in the proof of Theorem 2, we construct a set of tight constraints $\mathcal{L}$ such that every variable b'_i appears in at least one of these constraints; however, now we have to be more careful about the choice of constraints in $\mathcal{L}$. We construct $\mathcal{L}$ inductively as follows. Start by setting $\mathcal{L} = \{I_1\}$. At the jth step, suppose that all variables up to (but not including) b'_{i_j} appear in at least one inequality in $\mathcal{L}$. Add I_{i_j} to $\mathcal{L}$.

Note that for any j we have $y_{i_{j+1}} > y_{i_j}$. This is because the inequalities added to $\mathcal{L}$ during the first j steps did not cover $b'_{i_{j+1}}$. See Figure 3. Since $y_{i_{j+2}} > y_{i_{j+1}}$, we must also have $x_{i_{j+2}} > y_{i_j}$: otherwise, $\hat{P}_{i_{j+1}}$ would not be the "rightmost" constraint for $b'_{i_{j+1}}$. Therefore, the variables in $I_{i_{j+2}}$ and I_{i_j} do not overlap, and hence no b'_i can appear in more than two inequalities in $\mathcal{L}$.

Now we follow the argument of the proof of Theorem 2 to finish. By adding up all of the (tight) inequalities in $\mathcal{L}$ for b'_i we obtain $2\sum_{i=1,\ldots,k} b'_i \geq \sum_{j=1,\ldots,k} c(\hat{P}_j \setminus P)$. On the other hand, all these inequalities appear in condition (2), so they must hold for b''_i, i.e., $\sum_{i=1,\ldots,k} b''_i \leq \sum_{j=1,\ldots,k} c(\hat{P}_j \setminus P)$, so $\mathrm{TUmax}(\mathbf{c}) \leq 2\mathrm{TUmin}(\mathbf{c})$. □

5. TRUTHFUL MECHANISMS FOR VERTEX COVER

Recall that for a vertex-cover auction on a graph $G = (V, E)$, an *allocation rule* is an algorithm that takes as input a bid b_v for each vertex and returns a vertex cover $\hat{S}$ of G. As explained in Section 2, we can combine a monotone allocation rule with threshold payments to obtain a truthful auction.

Two natural examples of monotone allocation rules are A_{opt}, i.e., the algorithm that finds an optimal vertex cover, and the greedy algorithm A_{GR}. However, A_{opt} cannot be guaranteed to run in polynomial time unless $P = NP$ and A_{GR} has approximation ratio of $\log n$.

Another approximation algorithm for vertex cover, which has approximation ratio 2, is the *local ratio* algorithm A_{LR} [2, 3]. This algorithm considers the edges of G one by one. Given an edge $e = (u, v)$, it computes $\epsilon = \min\{b_u, b_v\}$ and sets $b_u = b_u - \epsilon$, $b_v = b_v - \epsilon$. After all edges have been processed, A_{LR} returns the set of vertices $\{v \mid b_v = 0\}$. It is not hard to check that if the order in which the edges are considered is independent of the bids, then this algorithm is monotone as well. Hence, we can use it to construct a truthful auction that is guaranteed to select a vertex cover whose cost is within a factor of 2 from the optimal.

However, while the quality of the solution produced by A_{LR} is much better than that of A_{GR}, we still need to show that its total payment is not too high. In the next subsection, we bound the frugality ratio of A_{LR} (and, more generally, all algorithms that satisfy the condition of *local optimality*, defined later) by 2Δ, where Δ is the maximum degree of G. We then prove a matching lower bound showing that for some graphs the frugality ratio of any truthful auction is at least $\Delta/2$.

5.1 Upper bound

We say that an allocation rule is *locally optimal* if whenever $b_v > \sum_{w\sim v} b_w$, the vertex v is not chosen. Note that for any such rule the threshold bid of v satisfies $t_v \leq \sum_{w\sim v} b_w$.

CLAIM 1. *The algorithms A_{opt}, A_{GR}, and A_{LR} are locally optimal.*

THEOREM 4. *Any vertex cover auction $\mathcal{M}$ that has a locally optimal and monotone allocation rule and pays each agent his threshold bid has frugality ratio $\phi_{\mathrm{NTUmin}}(\mathcal{M}) \leq 2\Delta$.*

To prove Theorem 4, we first show that the total payment of any locally optimal mechanism does not exceed $\Delta c(V)$. We then demonstrate that $\mathrm{NTUmin}(\mathbf{c}) \geq c(V)/2$. By combining these two results, the theorem follows.

LEMMA 5. *Consider a graph $G = (V, E)$ with maximum degree Δ. Let $\mathcal{M}$ be a vertex-cover auction on G that satisfies the conditions of Theorem 4. Then for any cost vector $\mathbf{c}$, the total payment of $\mathcal{M}$ satisfies $p_{\mathcal{M}}(\mathbf{c}) \leq \Delta c(V)$.*

PROOF. First note that any such auction is truthful, so we can assume that each agent's bid is equal to his cost. Let $\hat{S}$ be the vertex cover selected by $\mathcal{M}$. Then by local optimality

$$p_{\mathcal{M}}(\mathbf{c}) = \sum_{v\in\hat{S}} t_v \leq \sum_{v\in\hat{S}} \sum_{w\sim v} c_w \leq \sum_{w\in V} \Delta c_w = \Delta c(V).$$

□

We now derive a lower bound on $\mathrm{TUmax}(\mathbf{c})$; while not essential for the proof of Theorem 4, it helps us build the intuition necessary for that proof.

LEMMA 6. *For a vertex cover instance $G = (V, E)$ in which S is a minimum vertex cover, $\mathrm{TUmax}(\mathbf{c}, S) \geq c(V \setminus S)$*

PROOF. For a vertex w with at least one neighbour in S, let $d(w)$ denote the number of neighbours that w has in S. Consider the bid vector $\mathbf{b}$ in which, for each $v \in S$, $b_v = \sum_{w\sim v, w\notin S} \frac{c_w}{d(w)}$. Then $\sum_{v\in S} b_v = \sum_{v\in S} \sum_{w\sim v, w\notin S} c_w/d(w) = \sum_{w\notin S} c_w = c(V \setminus S)$. To finish we want to show that $\mathbf{b}$ is feasible in the sense that it satisfies (2). Consider a vertex cover T, and extend the bid vector $\mathbf{b}$ by assigning $b_v = c_v$ for $v \notin S$. Then

$$b(T) = c(T\setminus S) + b(S\cap T) \geq c(T\setminus S) + \sum_{v\in S\cap T} \sum_{w\in\overline{S}\cap\overline{T}: w\sim v} \frac{c_w}{d(w)},$$

and since all edges between $\overline{S} \cap \overline{T}$ and S go to $S \cap T$, the right-hand-side is equal to

$$c(T \setminus S) + \sum_{w\in\overline{S}\cap\overline{T}} c_w = c(T \setminus S) + c(\overline{S}\cap\overline{T}) = c(V \setminus S) = b(S).$$

□

Next, we prove a lower bound on $\mathrm{NTUmax}(\mathbf{c}, S)$; we will then use it to obtain a lower bound on $\mathrm{NTUmin}(\mathbf{c})$.

LEMMA 7. *For a vertex cover instance $G = (V, E)$ in which S is a minimum vertex cover, $\mathrm{NTUmax}(\mathbf{c}, S) \geq c(V \setminus S)$*

PROOF. If $c(S) \geq c(V \setminus S)$, by condition (1) we are done. Therefore, for the rest of the proof we assume that $c(S) < c(V \setminus S)$. We show how to construct a bid vector $(b_e)_{e\in S}$ that satisfies conditions (1) and (2) such that $b(S) \geq c(V \setminus S)$; clearly, this implies $\mathrm{NTUmax}(\mathbf{c}, S) \geq c(V \setminus S)$.

Recall that a network flow problem is described by a directed graph $\Gamma = (V_\Gamma, E_\Gamma)$, a source node $s \in V_\Gamma$, a sink node $t \in V_\Gamma$, and a vector of capacity constraints a_e, $e \in E_\Gamma$. Consider a network (V_Γ, E_Γ) such that $V_\Gamma = V \cup \{s, t\}$, $E_\Gamma = E_1 \cup E_2 \cup E_3$, where $E_1 = \{(s, v) \mid v \in S\}$, $E_2 = \{(v, w) \mid v \in S, w \in$

$V \setminus S, (v,w) \in E\}$, $E_3 = \{(w,t) \mid w \in V \setminus S\}$. Since S is a vertex cover for G, no edge of E can have both of its endpoints in $V \setminus S$, and by construction, E_2 contains no edges with both endpoints in S. Therefore, the graph (V, E_2) is bipartite with parts $(S, V \setminus S)$.

Set the capacity constraints for $e \in E_\Gamma$ as follows: $a_{(s,v)} = c_v$, $a_{(w,t)} = c_w$, $a_{(v,w)} = +\infty$ for all $v \in S$, $w \in V \setminus S$. Recall that a *cut* is a partition of the vertices in V_Γ into two sets C_1 and C_2 so that $s \in C_1$, $t \in C_2$; we denote such a cut by $C = (C_1, C_2)$. Abusing notation, we write $e = (u,v) \in C$ if $u \in C_1, v \in C_2$ or $u \in C_2, v \in C_1$, and say that such an edge $e = (u,v)$ *crosses* the cut C. The *capacity* of a cut C is computed as $\mathrm{cap}(C) = \sum_{(v,w)\in C} a_{(v,w)}$. We have $\mathrm{cap}(s, V \cup \{t\}) = c(S)$, $\mathrm{cap}(\{s\} \cup V, t) = c(V \setminus S)$.

Let $C_{\min} = (\{s\} \cup S' \cup W', \{t\} \cup S'' \cup W'')$ be a minimum cut in Γ, where $S', S'' \subseteq S$, $W', W'' \subseteq V \setminus S$. See Figure 4. As $\mathrm{cap}(C_{\min}) \le \mathrm{cap}(s, V \cup \{t\}) = c(S) < +\infty$, and any edge in E_2 has infinite capacity, no edge $(u,v) \in E_2$ crosses $C_{\min}$.

Consider the network $\Gamma' = (V_{\Gamma'}, E_{\Gamma'})$, where $V_{\Gamma'} = \{s\} \cup S' \cup W' \cup \{t\}$, $E_{\Gamma'} = \{(u,v) \in E_\Gamma \mid u, v \in V_{\Gamma'}\}$. Clearly, $C' = (\{s\} \cup S' \cup W', \{t\})$ is a minimum cut in Γ' (otherwise, there would exist a smaller cut for Γ). As $\mathrm{cap}(C') = c(W')$, we have $c(S') \ge c(W')$.

Now, consider the network $\Gamma'' = (V_{\Gamma''}, E_{\Gamma''})$, where $V_{\Gamma''} = \{s\} \cup S'' \cup W'' \cup \{t\}$, $E_{\Gamma''} = \{(u,v) \in E_\Gamma \mid u, v \in V_{\Gamma''}\}$. Similarly, $C'' = (\{s\}, S'' \cup W'' \cup \{t\})$ is a minimum cut in Γ'', $\mathrm{cap}(C'') = c(S'')$. As the size of a maximum flow from s to t is equal to the capacity of a minimum cut separating s and t, there exists a flow $\mathcal{F} = (f_e)_{e \in E_{\Gamma''}}$ of size $c(S'')$. This flow has to saturate all edges between s and S'', i.e., $f_{(s,v)} = c_v$ for all $v \in S''$. Now, increase the capacities of all edges between s and S'' to $+\infty$. In the modified network, the capacity of a minimum cut (and hence the size of a maximum flow) is $c(W'')$, and a maximum flow $\mathcal{F}' = (f'_e)_{e \in E_{\Gamma''}}$ can be constructed by greedily augmenting $\mathcal{F}$.

Set $b_v = c_v$ for all $v \in S'$, $b_v = f'_{(s,v)}$ for all $v \in S''$. As $\mathcal{F}'$ is constructed by augmenting $\mathcal{F}$, we have $b_v \ge c_v$ for all $v \in S$, i.e., condition (1) is satisfied.

Now, let us check that no vertex cover $T \subseteq V$ can violate condition (2). Set $T_1 = T \cap S'$, $T_2 = T \cap S''$, $T_3 = T \cap W'$, $T_4 = T \cap W''$; our goal is to show that $b(S' \setminus T_1) + b(S'' \setminus T_2) \le c(T_3) + c(T_4)$. Consider all edges $(u,v) \in E$ such that $u \in S' \setminus T_1$. If $(u,v) \in E_2$ then $v \in T_3$ (no edge in E_2 can cross the cut), and if $u, v \in S$ then $v \in T_1 \cup T_2$. Hence, $T_1 \cup T_3 \cup S''$ is a vertex cover for G, and therefore $c(T_1) + c(T_3) + c(S'') \ge c(S) = c(T_1) + c(S' \setminus T_1) + c(S'')$. Consequently, $c(T_3) \ge c(S' \setminus T_1) = b(S' \setminus T_1)$. Now, consider the vertices in $S'' \setminus T_2$. Any edge in E_2 that starts in one of these vertices has to end in T_4 (this edge has to be covered by T, and it cannot go across the cut). Therefore, the total flow out of $S'' \setminus T_2$ is at most the total flow out of T_4, i.e., $b(S'' \setminus T_2) \le c(T_4)$. Hence, $b(S' \setminus T_1) + b(S'' \setminus T_2) \le c(T_3) + c(T_4)$. □

Finally, we derive a lower bound on the payment bound that is of interest to us, namely, $\mathrm{NTUmin}(\mathbf{c})$.

LEMMA 8. *For a vertex cover instance $G = (V, E)$ in which S is a minimum vertex cover,* $\mathrm{NTUmin}(\mathbf{c}, S) \ge c(V \setminus S)$

PROOF. Suppose for contradiction that $\mathbf{c}$ is a cost vector with minimum-cost vertex cover S and $\mathrm{NTUmin}(\mathbf{c}, S) < c(V \setminus S)$. Let $\mathbf{b}$ be the corresponding bid vector and let $\mathbf{c}'$ be a new cost vector with $c'_v = b_v$ for $v \in S$ and $c'_v = c_v$ for $v \notin S$. Condition (2) guarantees that S is an optimal solution to the cost vector $\mathbf{c}'$. Now compute a bid vector $\mathbf{b}'$ corresponding to $\mathrm{NTUmax}(\mathbf{c}', S)$. We claim that $b'_v = c'_v$ for any $v \in S$. Indeed, suppose that $b'_v > c'_v$ for some $v \in S$ ($b'_v = c'_v$ for $v \notin S$ by construction). As $\mathbf{b}$ satisfies conditions (1)–(3), among the inequalities in (2) there is one that is tight for v and the bid vector $\mathbf{b}$. That is, $b(S \setminus T) = c(T \setminus S)$. By the construction of $\mathbf{c}'$, $c'(S \setminus T) = c'(T \setminus S)$. Now since $b'_w \ge c'_w$ for all $w \in S$, $b'_v > c'_v$ implies $b'(S \setminus T) > c'(S \setminus T) = c'(T \setminus S)$. But this violates (2). So we now know $\mathbf{b}' = \mathbf{c}'$. Hence, we have $\mathrm{NTUmax}(\mathbf{c}', S) = \sum_{v \in S} b_v = \mathrm{NTUmin}(\mathbf{c}, S) < c(V \setminus S)$, giving a contradiction to the fact that $\mathrm{NTUmax}(\mathbf{c}', S) \ge c'(V \setminus S)$ which we proved in Lemma 7. □

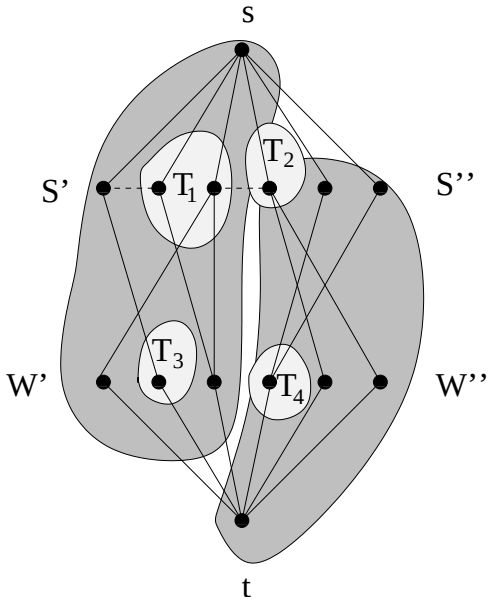

Figure 4: Proof of Lemma 7. Dashed lines correspond to edges in $E \setminus E_2$

As $\mathrm{NTUmin}(\mathbf{c}, S)$ satisfies condition (1), it follows that we have $\mathrm{NTUmin}(\mathbf{c}, S) \ge c(S)$. Together will Lemma 8, this implies $\mathrm{NTUmin}(\mathbf{c}, S) \ge \max\{c(V \setminus S), c(S)\} \ge c(V)/2$. Combined with Lemma 5, this completes the proof of Theorem 4.

REMARK 3. *As $\mathrm{NTUmin}(\mathbf{c}) \le \mathrm{NTUmax}(\mathbf{c}) \le \mathrm{TUmax}(\mathbf{c})$, our bound of 2Δ extends to the smaller frugality ratios that we consider, i.e., $\phi_{\mathrm{NTUmax}}(\mathcal{M})$ and $\phi_{\mathrm{TUmax}}(\mathcal{M})$. It is not clear whether it extends to the larger frugality ratio $\phi_{\mathrm{TUmin}}(\mathcal{M})$. However, the frugality ratio $\phi_{\mathrm{TUmin}}(\mathcal{M})$ is not realistic because the payment bound $\mathrm{TUmin}(\mathbf{c})$ is inappropriately low – we show in Section 6 that $\mathrm{TUmin}(\mathbf{c})$ can be significantly smaller than the total cost of a cheapest vertex cover.*

Extensions

We can also apply our results to monotone vertex-cover algorithms that do not necessarily output locally-optimal solutions. To do so, we simply take the vertex cover produced by any such algorithm and transform it into a locally-optimal one, considering the vertices in lexicographic order and replacing a vertex v with its neighbours whenever $b_v > \sum_{u \sim v} b_u$. Note that if a vertex u has been added to the vertex cover during this process, it means that it has a neighbour whose bid is higher than b_u, so after one pass all vertices in the vertex cover satisfy $b_v \le \sum_{u \sim v} b_u$. This procedure is monotone in bids, and it can only decrease the cost of the vertex cover. Therefore, using it on top of a monotone allocation rule with approx-

imation ratio α, we obtain a monotone locally-optimal allocation rule with approximation ratio α. Combining it with threshold payments, we get an auction with $\phi_{\mathrm{NTUmin}} \leq 2\Delta$. Since any truthful auction has a monotone allocation rule, this procedure transforms any truthful mechanism for the vertex-cover problem into a frugal one while preserving the approximation ratio.

5.2 Lower bound

In this subsection, we prove that the upper bound of Theorem 4 is essentially optimal. Our proof uses the techniques of [9], where the authors prove a similar result for shortest-path auctions.

THEOREM 5. *For any $\Delta > 0$ and any n, there exist a graph G of maximum degree Δ and size $N > n$ such that for any truthful mechanism $\mathcal{M}$ on G we have $\phi_{\mathrm{NTUmin}}(\mathcal{M}) \geq \Delta/2$.*

PROOF. Given n and Δ, set $k = \lceil n/2\Delta \rceil$. Let G be the graph that consists of k blocks $B_1, \ldots, B_k$ of size 2Δ each, where each B_i is a complete bipartite graph with parts L_i and R_i, $|L_i| = |R_i| = \Delta$.

We will consider two families of cost vectors for G. Under a cost vector $\mathbf{x} \in X$, each block B_i has one vertex of cost 1; all other vertices cost 0. Under a cost vector $\mathbf{y} \in Y$, there is one block that has two vertices of cost 1, one in each part, all other blocks have one vertex of cost 1, and all other vertices cost 0. Clearly, $|X| = (2\Delta)^k$, $|Y| = k(2\Delta)^{k-1}\Delta^2$. We will now construct a bipartite graph W with the vertex set $X \cup Y$ as follows.

Consider a cost vector $\mathbf{y} \in Y$ that has two vertices of cost 1 in B_i; let these vertices be $v_l \in L_i$ and $v_r \in R_i$. By changing the cost of either of these vertices to 0, we obtain a cost vector in X. Let $\mathbf{x}_l$ and $\mathbf{x}_r$ be the cost vectors obtained by changing the cost of v_l and v_r, respectively. The vertex cover chosen by $\mathcal{M}(\mathbf{y})$ must either contain all vertices in L_i or it must contain all vertices in R_i. In the former case, we put in W an edge from $\mathbf{y}$ to $\mathbf{x}_l$ and in the latter case we put in W an edge from $\mathbf{y}$ to $\mathbf{x}_r$ (if the vertex cover includes all of B_i, W contains both of these edges).

The graph W has at least $k(2\Delta)^{k-1}\Delta^2$ edges, so there must exist an $\mathbf{x} \in X$ of degree at least $k\Delta/2$. Let $y_1, \ldots, y_{k\Delta/2}$ be the other endpoints of the edges incident to $\mathbf{x}$, and for each $i = 1, \ldots, k\Delta/2$, let v_i be the vertex whose cost is different under $\mathbf{x}$ and $\mathbf{y}_i$; note that all v_i are distinct.

It is not hard to see that $\mathrm{NTUmin}(\mathbf{x}) \leq k$: the cheapest vertex cover contains the all-0 part of each block, and we can satisfy conditions (1)–(3) by letting one of the vertices in the all-0 part of each block to bid 1, while all other the vertices in the cheapest set bid 0.

On the other hand, by monotonicity of $\mathcal{M}$ we have $v_i \in \mathcal{M}(\mathbf{x})$ for $i = 1, \ldots, k\Delta/2$ (v_i is in the winning set under $\mathbf{y}_i$, and $\mathbf{x}$ is obtained from $\mathbf{y}_i$ by decreasing the cost of v_i), and moreover, the threshold bid of each v_i is at least 1, so the total payment of $\mathcal{M}$ on $\mathbf{x}$ is at least $k\Delta/2$. Hence, $\phi_{\mathrm{NTUmin}}(\mathcal{M}) \geq \mathcal{M}(\mathbf{x})/\mathrm{NTUmin}(\mathbf{x}) \geq \Delta/2$. □

REMARK 4. *The lower bound of Theorem 5 can be generalised to randomised mechanisms, where a randomised mechanism is considered to be truthful if it can be represented as a probability distribution over truthful mechanisms. In this case, instead of choosing the vertex $x \in X$ with the highest degree, we put both $(\mathbf{y}, \mathbf{x}_l)$ and $(\mathbf{y}, \mathbf{x}_r)$ into W, label each edge with the probability that the respective part of the block is chosen, and pick $\mathbf{x} \in X$ with the highest weighted degree. The argument can be further extended to a more permissive definition of truthfulness for randomised mechanisms, but this discussion is beyond the scope of this paper.*

6. PROPERTIES OF PAYMENT BOUNDS

In this section we consider several desirable properties of payment bounds and evaluate the four payment bounds proposed in this paper with respect to them. The particular properties that we are interested in are independence of the choice of S (Section 6.3), monotonicity (Section 6.4.1), computational hardness (Section 6.4.2), and the relationship with other reasonable bounds, such as the total cost of the cheapest set (Section 6.1), or the total VCG payment (Section 6.2).

6.1 Comparison with total cost

Our first requirement is that a payment bound should not be less than the total cost of the selected set. Payment bounds are used to evaluate the performance of set-system auctions. The latter have to satisfy individual rationality, i.e., the payment to each agent must be at least as large as his incurred costs; it is only reasonable to require the payment bound to satisfy the same requirement.

Clearly, $\mathrm{NTUmax}(\mathbf{c})$ and $\mathrm{NTUmin}(\mathbf{c})$ satisfy this requirement due to condition (1), and so does $\mathrm{TUmax}(\mathbf{c})$, since $\mathrm{TUmax}(\mathbf{c}) \geq \mathrm{NTUmax}(\mathbf{c})$. However, $\mathrm{TUmin}(\mathbf{c})$ fails this test. The example of Proposition 4 shows that for path auctions, $\mathrm{TUmin}(\mathbf{c})$ can be smaller than the total cost by a factor of 2. Moreover, there are set systems and cost vectors for which $\mathrm{TUmin}(\mathbf{c})$ is smaller than the cost of the cheapest set S by a factor of $\Omega(n)$. Consider, for example, the vertex-cover auction for the graph of Proposition 5 with the costs $c_{X_1} = \cdots = c_{X_{n-2}} = c_{X_{n-1}} = 1$, $c_{X_0} = 0$. The cost of a cheapest vertex cover is $n-2$, and the lexicographically first vertex cover of cost $n-2$ is $\{X_0, X_1, \ldots, X_{n-2}\}$. The constraints in (2) are $b_{X_i} + b_{X_0} \leq c_{X_{n-1}} = 1$. Clearly, we can satisfy conditions (2) and (3) by setting $b_{X_1} = \cdots = b_{X_{n-2}} = 0$, $b_{X_0} = 1$, which means that $\mathrm{TUmin}(\mathbf{c}) \leq 1$. This observation suggests that the payment bound $\mathrm{TUmin}(\mathbf{c})$ is too strong to be realistic, since it can be substantially lower than the cost of the cheapest feasible set.

Nevertheless, some of the positive results that were proved in [16] for $\mathrm{NTUmin}(\mathbf{c})$ go through for $\mathrm{TUmin}(\mathbf{c})$ as well. In particular, one can show that if the feasible sets are the bases of a monopoly-free matroid, then $\phi_{\mathrm{TUmin}}(\mathrm{VCG}) = 1$. To show that $\phi_{\mathrm{TUmin}}(\mathrm{VCG})$ is at most 1, one must prove that the VCG payment is at most $\mathrm{TUmin}(\mathbf{c})$. This is shown for $\mathrm{NTUmin}(\mathbf{c})$ in the first paragraph of the proof of Theorem 5 in [16]. Their argument does not use condition (1) at all, so it also applies to $\mathrm{TUmin}(\mathbf{c})$. On the other hand, $\phi_{\mathrm{TUmin}}(\mathrm{VCG}) \geq 1$ since $\phi_{\mathrm{TUmin}}(\mathrm{VCG}) \geq \phi_{\mathrm{NTUmin}}(\mathrm{VCG})$ and $\phi_{\mathrm{NTUmin}}(\mathrm{VCG}) \geq 1$ by Proposition 7 of [16] (and also by Proposition 6 below).

6.2 Comparison with VCG payments

Another measure of suitability for payment bounds is that they should not result in frugality ratios that are less then 1 for well-known truthful mechanisms. If this is indeed the case, the payment bound may be too weak, as it becomes too easy to design mechanisms that perform well with respect to it. It particular, a reasonable requirement is that a payment bound should not exceed the total payment of the classical VCG mechanism.

The following proposition shows that $\mathrm{NTUmax}(\mathbf{c})$, and therefore also $\mathrm{NTUmin}(\mathbf{c})$ and $\mathrm{TUmin}(\mathbf{c})$, do not exceed the VCG payment $p_{\mathrm{VCG}}(\mathbf{c})$. The proof essentially follows the argument of Proposition 7 of [16] and can be found in the full version of this paper [8].

PROPOSITION 6. $\phi_{\mathrm{NTUmax}}(\mathrm{VCG}) \geq 1$.

Proposition 6 shows that none of the payment bounds $\mathrm{TUmin}(\mathbf{c})$, $\mathrm{NTUmin}(\mathbf{c})$ and $\mathrm{NTUmax}(\mathbf{c})$ exceeds the payment of VCG. However, the payment bound $\mathrm{TUmax}(\mathbf{c})$ can be larger that the total

VCG payment. In particular, for the instance in Proposition 5, the VCG payment is smaller than $\mathrm{TUmax}(\mathbf{c})$ by a factor of $n-2$. We have already seen that $\mathrm{TUmax}(\mathbf{c}) \geq n-2$. On the other hand, under VCG, the threshold bid of any X_i, $i = 1, \ldots, n-2$, is 0: if any such vertex bids above 0, it is deleted from the winning set together with X_0 and replaced with X_{n-1}. Similarly, the threshold bid of X_0 is 1, because if X_0 bids above 1, it can be replaced with X_{n-1}. So the VCG payment is 1.

This result is not surprising: the definition of $\mathrm{TUmax}(\mathbf{c})$ implicitly assumes there is co-operation between the agents, while the computation of VCG payments does not take into account any interaction between them. Indeed, co-operation enables the agents to extract higher payments under VCG. That is, VCG is not group-strategyproof. This suggests that as a payment bound, $\mathrm{TUmax}(\mathbf{c})$ may be too liberal, at least in a context where there is little or no co-operation between agents. Perhaps $\mathrm{TUmax}(\mathbf{c})$ can be a good benchmark for measuring the performance of mechanisms designed for agents that can form coalitions or make side payments to each other, in particular, group-strategyproof mechanisms.

Another setting in which bounding ϕ_{TUmax} is still of some interest is when, for the underlying problem, the optimal allocation and VCG payments are NP-hard to compute. In this case, finding a *polynomial-time computable* mechanism with good frugality ratio with respect to $\mathrm{TUmax}(\mathbf{c})$ is a non-trivial task, while bounding the frugality ratio with respect to more challenging payment bounds could be too difficult. To illustrate this point, compare the proofs of Lemma 6 and Lemma 7: both require some effort, but the latter is much more difficult than the former.

6.3 The choice of S

All payment bounds defined in this paper correspond to the total bid of all elements in the cheapest feasible set, where ties are broken lexicographically. While this definition ensures that our payment bounds are well-defined, the particular choice of the draw-resolution rule appears arbitrary, and one might wonder if our payment bounds are sufficiently robust to be independent of this choice. It turns out that is indeed the case for $\mathrm{NTUmin}(\mathbf{c})$ and $\mathrm{NTUmax}(\mathbf{c})$, i.e., these bounds do not depend on the draw-resolution rule. To see this, suppose that two feasible sets S_1 and S_2 have the same cost. In the computation of $\mathrm{NTUmin}(\mathbf{c}, S_1)$, all vertices in $S_1 \setminus S_2$ would have to bid their true cost, since otherwise S_2 would become cheaper than S_1. Hence, any bid vector for S_1 can only have $b_e \neq c_e$ for $e \in S_1 \cap S_2$, and hence constitutes a valid bid vector for S_2 and vice versa. A similar argument applies to $\mathrm{NTUmax}(\mathbf{c})$.

However, for $\mathrm{TUmin}(\mathbf{c})$ and $\mathrm{TUmax}(\mathbf{c})$ this is not the case. For example, consider the set system

$$\mathcal{E} = \{e_1, e_2, e_3, e_4, e_5\},$$
$$\mathcal{F} = \{S_1 = \{e_1, e_2\}, S_2 = \{e_2, e_3, e_4\}, S_3 = \{e_4, e_5\}\}$$

with the costs $c_1 = 2$, $c_2 = c_3 = c_4 = 1$, $c_5 = 3$. The cheapest sets are S_1 and S_2. Now $\mathrm{TUmax}(\mathbf{c}, S_1) \leq 4$, as the total bid of the elements in S_1 cannot exceed the total cost of S_3. On the other hand, $\mathrm{TUmax}(\mathbf{c}, S_2) \geq 5$, as we can set $b_2 = 3, b_3 = 0, b_4 = 2$. Similarly, $\mathrm{TUmin}(\mathbf{c}, S_1) = 4$, because the inequalities in (2) are $b_1 \leq 2$ and $b_1 + b_2 \leq 4$. But $\mathrm{TUmin}(\mathbf{c}, S_2) \leq 3$ as we can set $b_2 = 1$, $b_3 = 2$, $b_4 = 0$.

6.4 Negative results for $\mathrm{NTUmin}(\mathbf{c})$ and $\mathrm{TUmin}(\mathbf{c})$

The results in [16] and our vertex cover results are proved for the frugality ratio ϕ_{NTUmin}. Indeed, it can be argued that ϕ_{NTUmin} is the "best" definition of frugality ratio, because among all reasonable payment bounds (i.e., ones that are at least as large as the cost of the cheapest feasible set), it is most demanding of the algorithm. However, $\mathrm{NTUmin}(\mathbf{c})$ is not always the easiest or the most natural payment bound to work with. In this subsection, we discuss several disadvantages of $\mathrm{NTUmin}(\mathbf{c})$ (and also $\mathrm{TUmin}(\mathbf{c})$) as compared to $\mathrm{NTUmax}(\mathbf{c})$ and $\mathrm{TUmax}(\mathbf{c})$.

6.4.1 Nonmonotonicity

The first problem with $\mathrm{NTUmin}(\mathbf{c})$ is that it is not monotone with respect to $\mathcal{F}$, i.e., it may increase when one adds a feasible set to $\mathcal{F}$. (It is, however, monotone in the sense that a losing agent cannot become a winner by raising his cost.) Intuitively, a good payment bound should satisfy this monotonicity requirement, as adding a feasible set increases the competition, so it should drive the prices down. Note that this indeed the case for $\mathrm{NTUmax}(\mathbf{c})$ and $\mathrm{TUmax}(\mathbf{c})$ since a new feasible set adds a constraint in (2), thus limiting the solution space for the respective linear program.

PROPOSITION 7. *Adding a feasible set to $\mathcal{F}$ can increase the value of $\mathrm{NTUmin}(\mathbf{c})$ by a factor of $\Omega(n)$.*

PROOF. Let $\mathcal{E} = \{x, xx, y_1, \ldots, y_n, z_1, \ldots, z_n\}$. Set $Y = \{y_1, \ldots, y_n\}$, $S = Y \cup \{x\}$, $T_i = Y \setminus \{y_i\} \cup \{z_i\}$, $i = 1, \ldots, n$, and suppose that $\mathcal{F} = \{S, T_1, \ldots, T_n\}$. The costs are $c_x = 0$, $c_{xx} = 0$, $c_{y_i} = 0$, $c_{z_i} = 1$ for $i = 1, \ldots, n$. Note that S is the cheapest feasible set. Let $\mathcal{F}' = \mathcal{F} \cup \{T_0\}$, where $T_0 = Y \cup \{xx\}$. For $\mathcal{F}$, the bid vector $b_{y_1} = \cdots = b_{y_n} = 0$, $b_x = 1$ satisfies (1), (2), and (3), so $\mathrm{NTUmin}(\mathbf{c}) \leq 1$. For $\mathcal{F}'$, S is still the lexicographically-least cheapest set. Any optimal solution has $b_x = 0$ (by constraint in (2) with T_0). Condition (3) for y_i implies $b_x + b_{y_i} = c_{z_i} = 1$, so $b_{y_i} = 1$ and $\mathrm{NTUmin}(\mathbf{c}) = n$. □

For path auctions, it has been shown [18] that $\mathrm{NTUmin}(\mathbf{c})$ is non-monotone in a slightly different sense, i.e., with respect to adding a new edge (agent) rather than a new feasible set (a team of existing agents).

REMARK 5. *We can also show that $\mathrm{NTUmin}(\mathbf{c})$ is non-monotone for vertex cover. In this case, adding a new feasible set corresponds to* deleting *edges from the graph. It turns out that deleting a single edge can increase $\mathrm{NTUmin}(\mathbf{c})$ by a factor of $n-2$; the construction is similar to that of Proposition 5.*

6.4.2 NP-Hardness

Another problem with $\mathrm{NTUmin}(\mathbf{c}, S)$ is that it is NP-hard to compute even if the number of feasible sets is polynomial in n. Again, this puts it at a disadvantage compared to $\mathrm{NTUmax}(\mathbf{c}, S)$ and $\mathrm{TUmax}(\mathbf{c}, S)$ (see Remark 1).

THEOREM 6. *Computing $\mathrm{NTUmin}(\mathbf{c})$ is NP-hard, even when the lexicographically-least feasible set S is given in the input.*

PROOF. We reduce EXACT COVER BY 3-SETS(X3C) to our problem. An instance of X3C is given by a universe $G = \{g_1, \ldots, g_n\}$ and a collection of subsets $C_1, \ldots, C_m$, $C_i \subset G$, $|C_i| = 3$, where the goal is to decide whether one can cover G by $n/3$ of these sets. Observe that if this is indeed the case, each element of G is contained in exactly one set of the cover.

LEMMA 9. *Consider a minimisation problem P of the following form: Minimise $\sum_{i=1,\ldots,n} b_i$ under conditions (1) $b_i \geq 0$ for all $i = 1, \ldots, n$; (2) for any $j = 1, \ldots, k$ we have $\sum_{b_i \in S_j} b_i \leq a_j$, where $S_j \subseteq \{b_1, \ldots, b_n\}$; (3) for each b_j, one of the constraints in (2) involving it is tight. For any such P, one can construct a set system S and a vector of costs c such that $\mathrm{NTUmin}(\mathbf{c})$ is the optimal solution to P.*

PROOF. The construction is straightforward: there is an element of cost 0 for each b_i, an element of cost a_j for each a_j, the feasible solutions are $\{b_1, \ldots, b_n\}$, or any set obtained from $\{b_1, \ldots, b_n\}$ by replacing the elements in S_j by a_j. □

By this lemma, all we have to do to prove Theorem 6 is to show how to solve X3C by using the solution to a minimisation problem of the form given in Lemma 9. We do this as follows. For each C_i, we introduce 4 variables x_i, $\bar{x}_i$, a_i, and b_i. Also, for each element g_j of G there is a variable d_j. We use the following set of constraints:

- In (1), we have constraints $x_i \geq 0$, $\bar{x}_i \geq 0$, $a_i \geq 0$, $b_i \geq 0$, $d_j \geq 0$ for all $i = 1, \ldots, m$, $j = 1, \ldots, n$.
- In (2), for all $i = 1, \ldots, m$, we have the following 5 constraints: $x_i + \bar{x}_i \leq 1$, $x_i + a_i \leq 1$, $\bar{x}_i + a_i \leq 1$, $x_i + b_i \leq 1$, $\bar{x}_i + b_i \leq 1$. Also, for all $j = 1, \ldots, n$ we have a constraint of the form $x_{i_1} + \cdots + x_{i_k} + d_j \leq 1$, where $C_{i_1}, \ldots, C_{i_k}$ are the sets that contain g_j.

The goal is to minimize $z = \sum_i (x_i + \bar{x}_i + a_i + b_i) + \sum_j d_j$.

Observe that for each j, there is only one constraint involving d_j, so by condition (3) it must be tight.

Consider the two constraints involving a_i. One of them must be tight, and therefore $x_i + \bar{x}_i + a_i + b_i \geq x_i + \bar{x}_i + a_i \geq 1$. Hence, for any feasible solution to (1)–(3) we have $z \geq m$. Now, suppose that there is an exact set cover. Set $d_j = 0$ for $j = 1, \ldots, n$. Also, if C_i is included in this cover, set $x_i = 1$, $\bar{x}_i = a_i = b_i = 0$, otherwise set $\bar{x}_i = 1$, $x_i = a_i = b_i = 0$. Clearly, all inequalities in (2) are satisfied (we use the fact that each element is covered exactly once), and for each variable, one of the constraints involving it is tight. This assignment results in $z = m$.

Conversely, suppose there is a feasible solution with $z = m$. As each addend of the form $x_i + \bar{x}_i + a_i + b_i$ contributes at least 1, we have $x_i + \bar{x}_i + a_i + b_i = 1$ for all i, $d_j = 0$ for all j. We will now show that for each i, either $x_i = 1$ and $\bar{x}_i = 0$, or $x_i = 0$ and $\bar{x}_i = 1$. For the sake of contradiction, suppose that $x_i = \delta < 1$, $\bar{x}_i = \delta' < 1$. As one of the constraints involving a_i must be tight, we have $a_i \geq \min\{1 - \delta, 1 - \delta'\}$. Similarly, $b_i \geq \min\{1 - \delta, 1 - \delta'\}$. Hence, $x_i + \bar{x}_i + a_i + b_i = 1 = \delta + \delta' + 2\min\{1 - \delta, 1 - \delta'\} > 1$. To finish the proof, note that for each $j = 1, \ldots, m$ we have $x_{i_1} + \cdots + x_{i_k} + d_j = 1$ and $d_j = 0$, so the subsets that correspond to $x_i = 1$ constitute a set cover.

REMARK 6. *In the proofs of Proposition 7 and Theorem 6 all constraints in (1) are of the form $b_e \geq 0$. Hence, the same results are true for* $\mathrm{TUmin}(\mathbf{c})$.

REMARK 7. *For shortest-path auctions, the size of $\mathcal{F}$ can be superpolynomial. However, there is a polynomial-time separation oracle for constraints in (2) (to construct one, use any algorithm for finding shortest paths), so one can compute* $\mathrm{NTUmax}(\mathbf{c})$ *and* $\mathrm{TUmax}(\mathbf{c})$ *in polynomial time. On the other hand, recently and independently it was shown [18] that computing* $\mathrm{NTUmin}(\mathbf{c})$ *for shortest-path auctions is NP-hard.*

7. REFERENCES

[1] A. Archer and E. Tardos, Frugal path mechanisms. In *Proceedings of the 13th Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 991–999, 2002

[2] R. Bar-Yehuda, K. Bendel, A. Freund, and D. Rawitz, Local ratio: A unified framework for approximation algorithms. In Memoriam: Shimon Even 1935-2004. *ACM Comput. Surv.*, 36(4):422–463, 2004

[3] R. Bar-Yehuda and S. Even, A local-ratio theorem for approximating the weighted vertex cover problem. *Annals of Discrete Mathematics*, 25:27–46, 1985

[4] E. Clarke, Multipart pricing of public goods. *Public Choice*, 8:17–33, 1971

[5] G. Calinescu, Bounding the payment of approximate truthful mechanisms. In *Proceedings of the 15th International Symposium on Algorithms and Computation*, pages 221–233, 2004

[6] A. Czumaj and A. Ronen, On the expected payment of mechanisms for task allocation. In *Proceedings of the 5th ACM Conference on Electronic Commerce (EC'04)*, 2004

[7] E. Elkind, True costs of cheap labor are hard to measure: edge deletion and VCG payments in graphs. In *Proceedings of the 6th ACM Conference on Electronic Commerce (EC'05)*, 2005

[8] E. Elkind, L. A. Goldberg, and P. W. Goldberg, Frugality ratios and improved truthful mechanisms for vertex cover. Available from `http://arxiv.org/abs/cs/0606044`, 2006

[9] E. Elkind, A. Sahai, and K. Steiglitz, Frugality in path auctions. In *Proceedings of the 15th Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 694–702, 2004

[10] J. Feigenbaum, C. H. Papadimitriou, R. Sami, and S. Shenker, A BGP-based mechanism for lowest-cost routing. In *Proceedings of the 21st Symposium on Principles of Distributed Computing*, pages 173–182, 2002

[11] A. Fiat, A. Goldberg, J. Hartline, and A. Karlin, Competitive generalized auctions. In *Proceedings of the 34th Annual ACM Symposium on Theory of Computation*, pages 72–81, 2002

[12] R. Garg, V. Kumar, A. Rudra and A. Verma, Coalitional games on graphs: core structures, substitutes and frugality. In *Proceedings of the 4th ACM Conference on Electronic Commerce (EC'03)*, 2005

[13] A. Goldberg, J. Hartline, and A. Wright, Competitive auctions and digital goods. In *Proceedings of the 12th Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 735–744, 2001

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

[15] N. Immorlica, D. Karger, E. Nikolova, and R. Sami, First-price path auctions. In *Proceedings of the 6th ACM Conference on Electronic Commerce (EC'05)*, 2005

[16] A. R. Karlin, D. Kempe, and T. Tamir, Beyond VCG: frugality of truthful mechanisms. In *Proceedings of the 46th Annual IEEE Symposium on Foundations of Computer Science*, pages 615–626, 2005

[17] D. Kempe, Personal communication, 2006

[18] N. Chen, A. R. Karlin, Cheap labor can be expensive, In *Proceedings of the 18th Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 735–744, 2007

[19] N. Nisan and A. Ronen, Algorithmic mechanism design. In *Proceedings of the 31st Annual ACM Symposium on Theory of Computation*, pages 129–140, 1999

[20] A. Ronen and R. Talisman, Towards generic low payment mechanisms for decentralized task allocation. In *Proceedings of the 7th International IEEE Conference on E-Commerce Technology*, 2005

[21] K. Talwar, The price of truth: frugality in truthful mechanisms. In *Proceedings of 20th International Symposium on Theoretical Aspects of Computer Science*, 2003

[22] W. Vickrey, Counterspeculation, auctions, and competitive sealed tenders. *Journal of Finance*, 16:8–37, 1961