Approximately Efficient Cost Sharing via Double Auctions*

Felix Fischer† Ian A. Kash‡ Peter Key§ Junxing Wang¶

Abstract

A well-known result in the cost-sharing literature asserts the non-existence of mechanisms that are strategyproof, (weakly) budget balanced, and economically efficient, even when budget balance and efficiency are sought only up to a multiplicative factor. An alternative measure of efficiency, approximate minimization of the social cost, can be achieved together with strategyproofness and budget balance, but in many settings of interest favors outcomes in which little or no service is provided. Focusing on a setting motivated by the provision and pricing of cloud services, we instead propose a mechanism that is strategyproof, budget balanced, and efficient up to an additive term that is independent of the number of potential consumers of the service. All three properties are thus achieved exactly in the large-market limit. The mechanism is inspired by the truthful double auction of McAfee and enables more uneven cost shares and consequently a less aggressive reduction of service than traditional approaches.

1 Introduction

Consider a service provider who can selectively serve a set of customers and then charge them for the service. Given a set N of potential customers the provider chooses to serve some subset $X \subseteq N$, incurs a cost $C(X)$, and charges each $i \in X$ a non-negative amount p_i. At the same time each potential customer $i \in N$ has a private value v_i for receiving service that it may misrepresent in order to maximize its payoff, which is equal to $v_i - p_i$ if it is served and 0 otherwise. An important property in this context is (weak) *budget balance*, which ensures that in any situation the revenue $\sum_{i \in X} p_i$ at least meets the cost $C(X)$. Budget balance is central to a large body of literature on cost sharing and notably puts emphasis on viability of the service, be it to provide a benefit to society or maintain longer-term business opportunities, rather than maximization of revenue in the short term.

A line of work pioneered by Moulin [22] has produced techniques for the design of cost-sharing mechanisms that are budget balanced as well as *strategyproof*, making it optimal for potential customers to truthfully reveal their values.[1] The mechanisms are, however, not efficient, i.e., they do not necessarily choose X to maximize the social welfare $sw(X) = (\sum_{i \in X} v_i) - C(X)$. In fact, strong impossibility results show strategyproofness, budget balance, and efficiency to be incompatible even in very simple settings [17, 27], and even when budget balance and efficiency are sought only in an approximate sense [14, 23].

To get around these impossibilities and still provide some form of efficiency guarantee, Roughgarden and Sundararajan [28] proposed to minimize the surrogate objective of social

*The first author is supported by the Einstein Foundation Berlin. Part of the work was done while the first and fourth authors were visiting Microsoft Research Cambridge.

†Technische Universität Berlin, Germany, fischerf@math.tu-berlin.de

‡Microsoft Research Cambridge, Cambridge, UK, iankash@microsoft.com

§Microsoft Research Cambridge, Cambridge, UK, peter.key@microsoft.com

¶Carnegie Mellon University, USA, thuwjx@gmail.com

[1] They are in fact group-strategyproof, such that no group of customers can benefit from jointly misrepresenting their values. We do not view this stronger property as essential in the setting we consider, and it is not achieved by our proposed solution.

cost, given by $sc(X) = (\sum_{i \notin X} v_i) + C(X)$. Mechanisms approximating the minimum social cost have since been obtained for many settings, most recently in terms of a black-box technique that provides an $O(\log n)$ approximation for any monotone cost function [16]. While minimization of the social cost also maximizes social welfare, approximation results for the former directly yield results for the latter only when the optimal solution is well separated in the sense that $sw(X) \geq kC(X)$, where k is the approximation ratio achieved with regard to social cost. When the optimal solution is not well separated, the approximate minimization of social cost potentially leads to poor practical performance. Indeed, in a situation where an investment of \$10 billion is enough to serve customers with a combined value of \$15 billion, not serving anyone provides a respectable 1.5-approximation with regard to social cost but leaves \$5 billion of welfare on the table.

The Problem We investigate the apparent tension between formal guarantees and good practical performance in cost sharing for a setting motivated by cloud computing. Companies providing cloud services, such as Amazon with AWS and Microsoft with Azure, face both a capacity planning and a pricing problem. Capacity planning includes not only the estimation of future demand as systems grow, but also the decision how much of that demand should be met. Since demand varies over the course of the day, week, and year, capacity built for peak demand may sit idle most of the time. Current pricing schemes work independently from capacity planning and tend to charge flat prices with little or no regard for demand dynamics. By contrast, determining both capacity and prices through a unified mechanism leads to a cost-sharing problem, which in turn can easily give rise to the aforementioned social cost issues. Suppose that a cloud service provider is considering buying a server at a cost of 1 that can be used to serve up to two customers, and that there are two customers with values $1 - \epsilon$ and $0.5 - \epsilon$, respectively. For ϵ close to zero, buying the server produces welfare of nearly 1.5 at a cost of 1 and thus is efficient. At the same time, not buying the server provides a $1.5/1 = 1.5$-approximation to the optimum social cost, and this continues to hold when the example is scaled by adding many copies of the two customers. What is more unfortunate is that known cost-sharing mechanisms, whether designed using the aforementioned black-box technique or by hand within the framework of Moulin, in fact opt not to buy the server.

Our Contribution We propose an alternative approach, inspired by the strategyproof double auction of McAfee [21], that arguably produces more practical outcomes. The approach maintains strategyproofness and budget balance, but compared to mechanisms in Moulin's framework relaxes efficiency in a less aggressive way. It guarantees to either produce an efficient outcome, or to provide a constant number of units of capacity less than an efficient outcome while rejecting the least valuable customers who would have been served by those units. This approaches perfect efficiency, and thus achieves all three properties, in the large-market limit. The key feature of an auction-based approach that enables our results is that it determines prices ex post based on the values of excluded bidders. By contrast, traditional cost-sharing mechanisms essentially determine prices ex ante, and based on the set of agents served rather than their values. On a more conceptual level, our results illustrate that there is significant scope for studying mechanisms outside the framework of Moulin, alternative efficiency guarantees, and cost sharing in large markets.

Related Work Strategyproof cost-sharing mechanisms have been studied for a wide variety of combinatorial optimization problems, an overview can be found in the survey of Jain and Mahdian [20]. We focus here on a setting where customers can be excluded and the cost of a solution is determined by its makespan. Mechanisms for sharing this type of cost have previously been considered by Brenner and Schäfer [7].

Moulin and Shenker [23] were the first to discuss the necessary tradeoff between budget balance and efficiency. Recognizing the impossibility to even approximate maximum social welfare subject to budget balance, Roughgarden and Sundararajan [28] instead framed the problem

in terms of the social cost and gave approximation results for a number of settings. More recently Georgiou and Swamy [16] obtained a black-box reduction that turns any algorithm approximately minimizing the cost of serving a set of customers into a mechanism that is strategyproof, budget balanced, and approximates the minimum social cost within an additional factor that is logarithmic in the number of customers. A less general reduction for settings with incomplete information was given by Fu et al. [15]. One may of course also relax strategyproofness to the end of achieving both budget balance and efficiency. This approach has been taken for example by Parkes et al. [26] in the context of a combinatorial exchange, but choosing the appropriate notion and degree of incentive compatibility in practice seems conceptually rather difficult.

Strategyproof double auctions for single-parameter settings were studied originally by McAfee [21], more recently by Deshmukh et al. [12] and Dütting et al. [13], and for an online setting by Bredin and Parkes [6]. Generalizations beyond a single unit of an undifferentiated good can roughly be assigned to two different categories. Work in the first category considers approaches like ours where prices are set for all participants simultaneously [19, 4, 5, 29]. This category of approaches has been applied in settings such as supply chains and dynamic spectrum auctions. Work in the second category sets prices first for one side of the market and then the other [10, 9]. This type of approach appears less relevant for our setting as it relies on a clear distinction between the two sides of the market. Carroll [8] considered double auctions that relax incentive compatibility to achieve efficiency. The significance of auction-based mechanisms for the tradeoff between efficiency and budget balance in cost sharing was noted more generally by Deb and Razzolini [11].

Capacity planning and pricing for cloud services has to our knowledge not previously been formulated as a cost-sharing problem. Perhaps closest to our work is work of Azar et al. [3] on strategyproof mechanisms for online scheduling with deadlines and their applications to cloud computing. We consider an offline model and in fact assume that demands are known and only valuations are private information, because even this simple model exhibits the phenomena we are interested in and allows for interesting new tradeoffs between budget balance and efficiency. A crucial aspect of our model that is absent from that of Azar et al. is the simultaneity of capacity planning and pricing. Other relevant topics that have been investigated in the context of cloud services include the viability of spot markets [1] and the effects of competition at various levels of the cloud ecosystem [2]. Cost sharing has been applied to other problems in networking, including multicast transmissions [14] and ISP pricing [18].

2 The Model

Consider a set $N = \{1, 2, \ldots, n\}$ of agents, where agent $i \in N$ requires $q_i \in Q$ units of service over a time period $[a_i, d_i)$, for a finite $Q \subseteq \mathbb{N}$ and $a_i, d_i \in \{0, 1, \cdots, m\}$ with $a_i < d_i$. Agent i values receiving service at v_i, and we assume that v_i is private information whereas the demand $\phi_i = (q_i, a_i, d_i) \in \Phi$ is publicly known.[2] A service provider decides to satisfy the demands of a subset $X \subseteq N$ of the agents by building a number of servers, at a constant cost c per server, each of which can then provide one unit of service during all m time slots $[0, 1), [1, 2), \cdots, [m-1, m)$. The number of servers required to serve a set X of agents[3] is thus equal to

$$\alpha(X) = \max_{j \in \{0,\ldots,m-1\}} \sum_{i \in X : j \in [a_i, d_i)} q_i$$

[2]We will see that interesting new tradeoffs in cost sharing can be obtained even for this relatively simple one-dimensional model. Whether our results can be generalized to multi-dimensional settings where demands are combinatorial, can be misrepresented, or both nevertheless is an interesting question.

[3]Note that because each agent requests service over an interval, we can in fact guarantee that each agent is served by the same server across all its time periods. This can for example be achieved by considering time periods one by one and making sure that any agent who was assigned a server in the previous time period is assigned the same server in the current one.

and the overall cost of serving them to

$$C(X) = c \cdot \alpha(X).$$

For a given set X we let x_i denote the indicator variable that is 1 if $i \in X$ and 0 otherwise.

The task of the service provider can be understood as that of choosing a (direct-revelation) mechanism M that takes a vector $v \in \mathbb{R}^n$ of reported valuations, as well as the known parameters N, ϕ, and c, and determines a subset $X \subseteq N$ of agents to serve and a vector $p \in \mathbb{R}^n$ of prices specifying how much each agent has to pay. We will usually omit the known parameters when they are clear from the context and write $M(v) = (X, p)$ for the subset and prices chosen by mechanism M for reported valuations v.

As it is common in the cost-sharing literature, we restrict our attention to mechanisms that are strategyproof and individually rational and study the tradeoff between two additional properties, budget balance and efficiency. Mechanism M is *strategyproof* if an agent cannot gain from misreporting its valuation, i.e., if for all $v \in \mathbb{R}^n$, $i \in N$, $v'_i \in \mathbb{R}$, $(X, p) = M(v)$, and $(X', p') = M(v'_i, v_{-i})$,

$$x_i v_i - p_i \geq x'_i v_i - p'_i.$$

It is *individually rational* if agents can only gain from participating, i.e., if for all $i \in N$, $v \in \mathbb{R}^n$, and $(X, p) = M(v)$,

$$x_i v_i - p_i \geq 0.$$ [4]

It is (weakly) *budget balanced* if the service provider always recovers its cost, i.e., if for all $v \in \mathbb{R}^n$ and $(X, p) = M(v)$,

$$\sum_{i \in N} p_i \geq C(X).$$

It finally is efficient if its choice of agents to serve maximizes social welfare, where the *social welfare* achieved by serving a subset X of agents when valuations are v is given by

$$sw(X, v) = \sum_{i \in X} v_i - C(X).$$

It is well known that the three properties cannot be achieved simultaneously, even if budget balance and efficiency are sought only up to multiplicative factors, for any nontrivial such factors [28]. We thus relax efficiency in an additive sense and call mechanism M *δ-efficient* if for all $v \in \mathbb{R}^n$ and $(X, p) = M(v)$,

$$sw(X, v) \geq \max_{X' \subseteq N} sw(X', v) - \delta.$$

Using this terminology, a mechanism is efficient if it is δ-efficient for $\delta = 0$. More generally, a mechanism that is δ-efficient for any δ independent of n approaches perfect efficiency as maximum social welfare grows. It is instructive to compare this bound to approximate minimization of the social cost measure proposed by Roughgarden and Sundararajan, which is equivalent to the existence of ρ such that for all $v \in \mathbb{R}^n$ and $(X, p) = M(v)$,

$$sw(X, v) \geq \max_{X' \subseteq N} \big(sw(X', v) - \rho C(X') \big).$$

We will achieve δ-efficiency for δ independent of n but dependent on other parameters of the problem at hand, and thus a new type of tradeoff that is incomparable to that provided by the social cost measure.

[4] For the mechanisms we consider, individual rationality follows directly from strategyproofness and the fact that $x_i = 0$ whenever $v_i = 0$ and $p_i = 0$ whenever $x_i = 0$.

3 An Example

We illustrate the model for the special case where each agent requires one unit of service for a single time slot, i.e., where $q_i = 1$ and $d_i = a_i + 1$ for all i. The efficient solution in this case can be determined by finding the maximum value k for which $\sum_j \{v_i : a_i = j\}_{(-k)} \geq c$, where $\{v_i : a_i = j\}_{(-k)}$ is the kth-highest value among agents interested in time slot j, and serving the k agents with the highest values for each time slot.

To see why existing approaches are far from efficient in this setting, consider the following simple mechanism, which is in fact the result of the reduction of Georgiou and Swamy [16]:

1. Initialize X to be the set of agents served in the efficient outcome.
2. If $v_i < c\alpha(X)/|X|$ for $i \in X$, remove i from X.
3. Repeat Step 2 until no agents are removed.
4. Serve agents in X; for $i \in X$, set $p_i = c\alpha(X)/|X|$.

Note that all agents served by the mechanism pay the same price, so the outcome can only be efficient if each agent served has a value of at least c/m. To see that the mechanism can go badly awry, set $m = 2$, $v_i = c - \epsilon$ if $a_i = 0$ and $v_i = c/2 - \epsilon$ if $a_i = 1$, where $\epsilon > 0$ is arbitrarily small. Then the mechanism does not serve a single agent whereas maximum social welfare grows linearly with the number of agents. If we generalize this example to m time slots with agents demanding time slot $[j, j+1)$ having a value of $c/(j+1) - \epsilon$, welfare per unit of supply is $\Theta(c \log m)$ but the mechanism still achieves welfare zero. While this is clearly not δ-efficient for any constant δ, it does provide a multiplicative approximation guarantee of $\Theta(\log m)$ with regard to social cost $sc(X) = (\sum_{i \notin X} v_i) + c\alpha(X)$ for the example and of $O(m)$ in general. To see the latter, note that the mechanism serves a subset of the agents served in the efficient outcome, and that any agent i with $v_i \geq c$ is guaranteed to be served. Denoting by A the set of agents served in the efficient outcome, the approximation ratio with regard to social cost is at most

$$\frac{c\alpha(A) + \sum_{i \in A \setminus X} v_i}{c\alpha(A)} \leq \frac{c\alpha(A) + \sum_{i \in A \setminus X} c}{c\alpha(A)} \leq \frac{c\alpha(A) + cm\alpha(A)}{c\alpha(A)} = m + 1.$$

This bound, which is even better than the generic bound of $O(\log n)$ provided by the black-box reduction when there are sufficiently many agents, suggests that approximate minimization of social cost is a rather poor proxy for good practical performance. There further is little hope to improve performance by means of a hand-designed Moulin mechanism: as these mechanisms fix the cost shares ex-ante, there always exists some instance where they will charge agents in one time slot too much and agents in another time slot too little, ending up with a welfare of zero.

3.1 McAfee's Double Auction

Rather than making all the allocation decisions, the service provider could instead simply determine a number of units of capacity to build and then sell these units to a subset of the agents, allowing the agents in turn to resell that capacity for the periods in which they do not need it. The operation of this secondary market can then be thought of as an auction, and capacity determination, selling, and operation of the secondary market can of course be thought to all take place in a single, unified mechanism.

To make this approach concrete, consider the case where $m = 2$ and assume that the service provider sells units of capacity at a fixed price of c to agents who desire capacity in the first time slot. These agents then become sellers in the secondary market, because they want to sell

the unused capacity for the second time slot, whereas agents who desire service in the second time slot become buyers. An agent i with $a_i = 0$ is willing to enter the secondary market as a seller as long as it is paid at least $c - v_i$, as any smaller amount would lead to a negative overall utility and would make it preferable not to get the unit of capacity in the first place. Similarly, an agent i with $a_i = 1$ would be willing to buy for a price of at most v_i. We are thus looking at a classic double auction setting, where sellers on one side need to be matched with buyers on the other.

A simplified version of the truthful double auction mechanism of McAfee [21] operates this market by sorting sellers from lowest asking price to highest, and buyers from highest willingness to pay to the lowest. It then pairs the lists and finds the *last efficient trade*, the lowest pair in the combined list for which the buyer is willing to pay at least what the seller is asking. It then executes all efficient trades except this last one, charging successful buyers the bid of the excluded buyer and paying successful sellers the asking price of the excluded seller.[5] The mechanism is strategyproof because agents cannot influence the price at which they trade, budget-balanced because prices are set based on a trade that would have covered the cost, and nearly efficient because it executes one less than the optimal number of trades.

3.2 A Generalized "Double" Auction

Assuming that each agent requires one unit of service for one time slot, McAfee's mechanism can easily be generalized to m time slots to obtain the following:

1. Let X' be the set of agents served in the efficient outcome.
2. Let S be the set of agents with the $\alpha(X') - 1$ highest values for each time slot.
3. Serve agents in $X = S \cup \{i \in X' \setminus S : v_i \geq c\}$; for $i \in X$ let p_i be the minimum of c and the value of the agent in $X' \setminus S$ with demand for the same time slots i, if such an agent exists, and 0 otherwise.

This mechanism is obviously strategyproof and budget-balanced. It is also δ-efficient for $\delta = cm$: the set of agents served is $X \subset X'$ with $|X' \setminus X| \leq m$, and for each $i \in X' \setminus X$, $v_i < c$.

In contrast to Moulin mechanisms, the mechanism sets prices for the different time slots based on agent values rather than fixing them ex ante. This is crucial for achieving approximate efficiency. In the next section we generalize the mechanism to solve two challenges posed by our more general model. First, when demands are for intervals rather than single time slots, there may no longer be a unique way to define the other side of the market relative to a particular agent. Second, agents for which $q_i > 1$ will be interested in buying multiple units from the respective other side, which makes the idea of reducing trade more complex.

4 The Mechanism

In our general model, strategyproofness, budget balance, and approximate efficiency is achieved by a mechanism we refer to as the Double-VCG mechanism and denote by M^{DVCG}. We assume for ease of exposition that ties are broken in an arbitrary but consistent way, and that demand is perfect in the sense that for each time period $j \in \{0, \ldots, m-1\}$ there is an infinite number of fictitious agents with demand $(1, j, j+1)$ and value 0. The former guarantees that values and the maximum social welfare in any situation are essentially unique, the latter that there exists an efficient solution in which every server that is built is used across all time periods.

Mechanism M^{DVCG} proceeds in three main rounds. In the first round, we invoke a mechanism M^{VCG} with VCG-like pricing that is maximal in range [25] subject to a perfect supply

[5] A more sophisticated version of the mechanism frequently is able to perform the last efficient trade as well, and to actually achieve strong budget balance when it does so.

Figure 1 VCG mechanism for perfect supply: $M^{VCG}(v, S) = (X, p)$

1: Maximize social welfare subject to perfect supply: let

$$X = \arg\max_{X' \in \mathcal{S}} sw(X', v),$$

where

$$\mathcal{S} = \left\{ X \subseteq S : \alpha(X) \cdot m = \sum_{i \in X} q_i(d_i - a_i) \right\}.$$

2: Charge agent i its externality: for $i \in S$, let

$$p_i = \left(\max_{X' \in \mathcal{S}: i \notin X'} sw(X', v)\right) - \left(sw(X, v) - x_i v_i\right).$$

Figure 2 Double-VCG mechanism: $M^{DVCG}(v) = (X, p)$

1: Augment the set of agents such that for each time period $j \in \{0, \ldots, m-1\}$, there is an infinite number of agents with demand $(1, j, j+1)$ and value 0. Denote the augmented set of agents by $\bar{N}$, the augmented demand and valuation vectors by $\bar{\phi}$ and $\bar{v}$.

2: VCG with perfect supply and demand: let $(S, p) = M^{VCG}(\bar{v}, \bar{N})$.

3: Trade reduction: for $i \in S$, let

$$p'_i = \left[\bar{v}_i\right]_{(LCM(Q)/q_i)}$$

$$S' = \{i \in S : \bar{v}_i > p'_i\},$$

where $[\bar{v}_i] = \{\bar{v}_\ell : \ell \in S, \bar{\phi}_\ell = \bar{\phi}_i\}$ and $Y_{(j)}$ denotes the jth order statistic of set Y.

4: VCG with perfect supply after trade reduction: let $(S'', p'') = M^{VCG}(\bar{v}, S')$.

5: Serve and price: let

$$X = (S'' \cap N) \cup \{i \in N \setminus S'' : v_i \geq cq_i\},$$

$$p_i = \begin{cases} \min\left\{cq_i, \max\{p_i, p'_i, p''_i\}\right\} & \text{if } i \in S'' \cap N, \\ cq_i & \text{if } i \in X \setminus (S'' \cap N), \\ 0 & \text{otherwise.} \end{cases}$$

restriction requiring all servers to be used across all time periods. This produces an initial set of agents to serve, S, with corresponding prices p. Note that S may include fictitious agents. A formal description of M^{VCG} is given in Figure 1. The combination of a restricted range and VCG-like prices makes M^{VCG} strategyproof; the perfect demand assumption implies that its first invocation produces an efficient outcome. The second round of M^{DVCG} consists of a trade reduction in the manner of the truthful double auction of McAfee [21]. This reduces the set S to a subset S'. One way to understand this trade reduction is in terms of endogenous reserve prices p', where the reserve price of agent i is equal to the jth-smallest value, for an appropriately chosen number j, among the agents with the same demand as i remaining after the first round. We will see that the appropriate number j to use is $LCM(Q)/q_i$ where

$$LCM(Q) = \min\{y : \text{for all } \ell \in Q,\ y \bmod \ell = 0\},$$

the least common multiple of the possible units of service requested by the agents. The third round runs M^{VCG} on the reduced S' to give candidate allocated agent set S'' and prices p''; the allocated agents are the non-fictitious agents in S'' together with those excluded agents for whom $v_i \geq cq_i$; prices are set for all allocated agents. A formal description of M^{DVCG} is given in Figure 2.

The goal in the latter two rounds is to achieve budget balance. In the simpler model of the

previous section, a reduction of one unit of trade per time slot was enough to ensure that the agents thus excluded could cover the cost of one unit of supply. This is no longer the case in the general model, where demands may differ among agents. Here, a reduction by $LCM(Q)/q_i$ units guarantees that the excluded agents collectively demand $LCM(Q)$ units of service and are willing to cover the corresponding cost.

Let us return to the example of Section 3, where $m = 2$, $v_i = c - \epsilon$ if $a_i = 0$ and $v_i = c/2 - \epsilon$ if $a_i = 1$, and suppose there is an even number of agents, $n/2$ of each type. The first invocation of the VCG mechanism then sets $S = N$, $p_1 = c/2 + \epsilon$, and $p_2 = \epsilon$. Trade reduction removes one agent of each type, such that $n/2 - 1$ agents of each type remain in S', and sets $p'_1 = v_1$ and $p'_2 = v_2$. The second invocation of the VCG mechanism finally sets $S'' = S'$, $p''_1 = p_1$, and $p''_2 = p_2$, so the Double-VCG mechanism ends up building $n/2 - 1$ units of service, serving $n/2 - 1$ agents of each type, and respectively charging them v_1 and v_2. The loss of efficiency is $O(1/n)$, illustrative of large-market efficiency.

5 Analysis

We are now ready to state the three key properties of the Double-VCG Mechanism: strategyproofness, budget balance, and approximate efficiency. We begin with strategyproofness, formal proofs of the results are relegated to Appendix A.

Theorem 1. *The Double-VCG mechanism is strategyproof.*

Budget balance holds because the first round of the mechanism determines an efficient outcome subject to perfect supply, and the perfect supply restriction is in fact crucial to achieving budget balance.

Theorem 2. *The Double-VCG mechanism is budget-balanced.*

Approximate efficiency clearly holds in McAfee's double auction because the outcome is reduced only by a single efficient trade. Our mechanism may end up reducing trade by a significantly larger number, both because we have many different demands and because we impose perfect supply. We are nevertheless able to bound the number by a value that is independent of n.

Theorem 3. *The Double-VCG mechanism is δ-efficient for $\delta = 2c\, LCM(Q)\, m^4\, |Q|$. In particular, δ is independent of n.*

With a more complex argument we can in fact reduce the bound by another factor of m. We present this argument in Appendix B, along with an example showing that the improved bound is tight.

6 Discussion

We have studied a cost-sharing problem motivated by capacity planning and pricing in cloud computing, where existing approaches provide a constant multiplicative approximation to social cost but no additive approximation to social welfare and solutions approximating social cost are not practical. Our solution to this problem is based on a double auction and in addition to satisfying strategyproofness and budget balance it approximates social welfare up to an additive term that is independent of the number of customers. It thus circumvents a well-known impossibility result, and achieves strategyproofness, budget balance, and efficiency, in the limit of a large market.

In deriving this result we have intentionally focused on a setting that is as simple as possible while still exposing some of the limitations of existing cost-sharing techniques and leading us to

a new type of approach. Particular assumptions we have made include that costs are linear and that agents have known demands over intervals. We have also used a particular mechanism to determine an initial outcome and have largely ignored computational considerations. We view the work reported here as a first step, and many interesting directions for future work remain.

Non-linear costs When costs are non-linear it is no longer obvious what it means for an agent to be willing to pay for its own servers, because the cost of doing so depends on the set of other agents being served. A change to the mechanism may thus be required. Possible choices for example include the marginal cost for a fixed ordering over all agents who are not currently being served, or the average marginal cost over all such orderings. As long as the chosen value is monotone, the mechanism will remain strategyproof. For budget balance and approximate efficiency the situation seems more complicated. The proof of budget balance works by contradiction and identifies a set of agents who collectively demand service in every time slot but the sum of whose per-unit payments do not cover the marginal cost of a machine. When costs are not convex, finding such a set of agents may no longer be possible. For convex cost functions there is hope that the same type of argument will go through, but additional work will be required to consider the average marginal cost of a number of servers when trade is reduced by more than one unit. Approximate efficiency is shown by bounding the aggregate value of agents served in an efficient outcome but not by the mechanism. Here as well, the intuition is that convex costs will only help in serving more agents, whereas the exclusion of agents in the trade reduction step may cause additional agents to be excluded when costs are non-convex.

Non-interval demands When agents demand sets of intervals rather than intervals, the number of different demands may become exponential rather than quadratic in m, which would directly impact the efficiency bound in a negative way. In practice this negative impact could be alleviated by limiting the number of intervals for each agent and thus the number of demands. Such a limitation of the number of demands could in fact also be applied in our current setting. Another problem that arises from non-interval demands is that in the proof of budget balance it becomes more difficult to find a subset of agents whose average payment per unit of service is too small. To restore budget balance it may thus be necessary to reduce trade more aggressively, again with a negative impact on the efficiency bound. Finally, in the presence of non-interval demands, it may no longer be possible to satisfy the additional constraint of serving each agent on the same servers across all its time periods without increasing the number of servers. This would lead to additional complications in the proofs of budget balance and approximate efficiency for settings where this constraint is required.

Unknown demands When demands are private information of the respective agents additional work will be required to achieve strategyproofness, for example to ensure that an agent cannot lower its price by requesting service over a superset of the time period for which it actually requires service. Such a guarantee would involve linking the pricing of related sets of time periods, and thus the trade reduction of such sets. What makes this a particularly interesting direction for future work is that independently of whether demands are known or not, linking trade reduction across different sets of time periods could potentially lead to a smaller reduction of trade and improved efficiency.

Uncertain or flexible demands In view of real-world applications in cloud computing, the assumption that demands are known in advance even to the agents and that capacity planning and pricing can be done at the same time seems rather strong. While there are obvious ways to decouple capacity planning and pricing, like treating costs as sunk and charging market-clearing prices, or viewing the available capacity only as a constraint and selecting an allocation and prices as if capacity was built when needed, this would almost certainly come at the expense of budget balance. It would therefore be more interesting to consider a stochastic or online model

Figure 3 VCG mechanism for perfect multiple supply: $M^{VCG}(v, S) = (X, p)$

1: Maximize social welfare subject to perfect multiple supply: let

$$X = \arg\max_{X' \in \mathcal{S}'} sw(X', v),$$

where

$$\mathcal{S}' = \{X \in \mathcal{S} : \forall \varphi \in \Phi, |\{i \in X : \phi_i = \varphi\}| \times q_i \bmod LCM(Q) = 0\}.$$

and

$$\mathcal{S} = \left\{X \subseteq S : \alpha(X) \cdot m = \sum\nolimits_{i \in X} q_i(d_i - a_i)\right\}$$

2: Charge agent i its externality: for $i \in S$, let

$$p_i = \left(\max_{X' \in \mathcal{S}' : i \notin X'} sw(X', v)\right) - \left(sw(X, v) - x_i v_i\right).$$

where capacity is planned based on a limited amount of information and updated as more information becomes available. A generalization of the online scheduling model of Azar et al. [3] to a variable number of machines provides an obvious starting point. Another interesting property of the model of Azar et al. is a certain degree of flexibility on the part of the agents, who may request q_i units of demand for n_i time periods between $[a_i, d_i)$ where $n_i \leq d_i - a_i$. Indeed, one of the benefits of time-differentiated pricing is that it may help flexible customers shift their demand appropriately.

Alternatives to the VCG mechanism While we have called our mechanism the Double-VCG mechanism, and the use of VCG-like pricing is instrumental to achieving strategyproofness, budget balance, and approximate efficiency in the setting at hand, it is an interesting observation that there is a certain degree of flexibility in the choice of mechanism used together with trade reduction. Replacing the VCG mechanism by any strategyproof mechanism maintains strategyproofness. Replacing the first invocation of the VCG by any mechanism that guarantees that no subset of S yields greater social welfare, or replacing the second invocation by any mechanism that guarantees the perfect supply restriction, maintains budget balance. Such substitutions may of course have a significant effect on the efficiency of the overall mechanism, but they may be useful in situations where a different objective than maximization of welfare is desired or where use of the VCG mechanism is impractical due to computational limitations.

Computational considerations Even for our relatively simple model, computational intractability of the VCG mechanism may be an issue. The outcome of the VCG mechanism can be computed efficiently via minimum-cost flows in the special case where $|Q| = 1$, and brute-force search for the outcome is fixed parameter tractable in $|Q|$ when all demands are for a single time period, i.e., when $d_i = a_i + 1$ for all i. By contrast, we show in Appendix C that the VCG mechanism is intractable in the most general version of our model via a reduction from the NP-complete EXACT-COVER problem.

While this makes M^{VCG} intractable in general, we may replace the VCG mechanism by a more restrictive maximal-in-range mechanism and still achieve strategyproofness and budget balance. Furthermore, our proof of approximate efficiency effectively provides a reduction to the case where $|Q| = 1$ by combining agents into super-agents whose total demand is $LCM(Q)$. The case where $|Q| = 1$ is computationally tractable, and it turns out that imposing this additional restriction on the maximal-in-range mechanism leads to an overall mechanism that is computationally efficient and δ-efficiency for the same bound $\delta = 2cm^4|Q|LCM(Q)$ as before. The modified maximal-in-range mechanism is formally specified in Figure 3.

To see that the claim concerning efficiency is correct, consider again the feasible solution X' consisting of super-agents we constructed in the proof of Theorem 3, and observe that it

is also feasible for a first invocation of the VCG mechanism with perfect multiple supply of Figure 3. By following the same steps as the proof of Theorem 3, we can thus obtain an upper bound on the loss in social welfare after this first invocation but before trade reduction of at most $cm^4|Q|LCM(Q)$. Going through the same steps again then yields an upper bound of $cm^4|Q|LCM(Q)$ on the additional loss in social welfare of the second invocation, because for each demand at most $LCM(Q)/q_i$ agents are eliminated by trade reduction. The crucial observation is that no additional agents need to be removed in order to form super-agents, because this was already done during the first invocation of the VCG mechanism with perfect multiple supply.

References

[1] V. Abhishek, I. A. Kash, and P. Key. Fixed and market pricing for cloud services. In *Proceedings of the 7th Workshop on the Economics of Networks, Systems, and Computation*, pages 157–162, 2012.

[2] J. Anselmi, D. Ardagna, J. C. S. Lui, A. Wierman, Y. Xu, and Z. Yang. The economics of the cloud: Price competition and congestion. In *Proceedings of the 8th Workshop on the Economics of Networks, Systems, and Computation*, 2013.

[3] Y. Azar, I. Kalp-Shaltiel, B. Lucier, I. Menache, J. Naor, and J. Yaniv. Truthful online scheduling with commitments. In *Proceedings of the 16th ACM Conference on Economics and Computation*, pages 715–732. ACM, 2015.

[4] M. Babaioff and W. E. Walsh. Incentive-compatible, budget-balanced, yet highly efficient auctions for supply chain formation. *Decision Support Systems*, 39(1):123–149, 2005.

[5] M. Babaioff, N. Nisan, and E. Pavlov. Mechanisms for a spatially distributed market. *Games and Economic Behavior*, 66(2):660–684, 2009.

[6] J. Bredin and D. C. Parkes. Models for truthful online double auctions. In *Proceedings of the 21st Annual Conference on Uncertainty in Artificial Intelligence*, pages 50–59. AUAI Press, 2005.

[7] J. Brenner and G. Schäfer. Group-strategyproof cost sharing mechanisms for makespan and other scheduling problems. *Theoretical Computer Science*, 401(1–3):96–106, 2008.

[8] G. Carroll. The efficiency-incentive tradeoff in double auction environments. Working Paper, 2011.

[9] L. Y. Chu. Truthful bundle/multiunit double auctions. *Management Science*, 55(7):1184–1198, 2009.

[10] L. Y. Chu and Z.-J. M. Shen. Truthful double auction mechanisms. *Operations Research*, 56(1):102–120, 2008.

[11] R. Deb and L. Razzolini. Auction-like mechanisms for pricing excludable public goods. *Journal of Economic Theory*, 99:340–368, 1999.

[12] K. Deshmukh, A. V. Goldberg, J. D. Hartline, and A. R. Karlin. Truthful and competitive double auctions. In *Proceedings of the 10th European Symposium on Algorithms*, pages 361–373. Springer, 2002.

[13] P. Dütting, T. Roughgarden, and I. Talgam-Cohen. Modularity and greed in double auctions. In *Proceedings of the 15th ACM Conference on Economics and Computation*, pages 241–258. ACM, 2014.

[14] J. Feigenbaum, C. Papadimitriou, and S. Shenker. Sharing the cost of muliticast transmissions. *Journal of Computer and System Sciences*, 63(1):21–41, 2001.

[15] H. Fu, B. Lucier, B. Sivan, and V. Syrgkanis. Cost-recovering Bayesian algorithmic mechanism design. In *Proceedings of the 14th ACM Conference on Economics and Computation*, pages 453–470. ACM, 2013.

[16] K. Georgiou and C. Swamy. Black-box reductions for cost-sharing mechanism design. *Games and Economic Behavior*, 2016. In press.

[17] J. Green, E. Kohlberg, and J.-J. Laffont. Partial equilibrium approach to the free-rider problem. *Journal of Public Economics*, 6(4):375–394, 1976.

[18] L. Gyarmati, M. Sirivianos, and N. Laoutaris. Sharing the cost of backbone networks: Simplicity vs. precision. In *Proceedings of the 7th Workshop on the Economics of Networks, Systems, and Computation*, pages 171–176, 2012.

[19] P. Huang, A. Scheller-Wolf, and K. Sycara. Design of a multi-unit double auction e-market. *Computational Intelligence*, 18(4):596–617, 2002.

[20] K. Jain and M. Mahdian. Cost sharing. In N. Nisan, T. Roughgarden, E. Tardos, and V. V. Vazirani, editors, *Algorithmic Game Theory*, chapter 15, pages 385–409. Cambridge University Press, 2007.

[21] R. P. McAfee. A dominant strategy double auction. *Journal of Economic Theory*, 56(2): 434–450, 1992.

[22] H. Moulin. Incremental cost sharing: Characterization by coalition strategy-proofness. *Social Choice and Welfare*, 16(2):279–320, 1999.

[23] H. Moulin and S. Shenker. Strategyproof sharing of submodular costs: Budget balance versus efficiency. *Economic Theory*, 18(3):511–533, 2001.

[24] R. B. Myerson. Optimal auction design. *Mathematics of Operations Research*, 6(1):58–73, 1981.

[25] N. Nisan and A. Ronen. Computationally feasible VCG mechanisms. *Journal of Artificial Intelligence Research*, 29:19–47, 2007.

[26] D. C. Parkes, J. Kalagnanam, and M. Eso. Achieving budget-balance with Vickrey-based payment schemes in exchanges. In *Proceedings of the 17th International Joint Conference on Artificial Intelligence*, pages 1161–1168, 2001.

[27] K. Roberts. The characterization of implementable choice rules. In J.-J. Laffont, editor, *Aggregation and revelation of preferences*. North-Holland, 1979.

[28] T. Roughgarden and M. Sundararajan. Quantifying inefficiency in cost-sharing mechanisms. *Journal of the ACM*, 56(4):23, 2009.

[29] X. Zhou and H. Zheng. Trust: A general framework for truthful double spectrum auctions. In *Proceedings of the 28th Conference on Computer Communications*, pages 999–1007. IEEE, 2009.

A Proofs of Results in Section 5

Proof of Theorem 1. By a well-known result of Myerson [24], it suffices to verify that the mechanism uses a monotone allocation rule and charges the correct payments. Since agents are either allocated or not, these payments correspond to the critical value, the minimum bid needed for the agent to be allocated. To be allocated, an agent must either be in S'' or have $v_i \geq cq_i$. The later requirement is clearly monotone. The former requires it to first be in S, then in S', and then in S''. Membership in S is chosen by a maximal-in-range mechanism, which is

monotone, and the critical value is equal to the VCG price p_i. Membership in S' requires not having one of the lowest $LCM(Q)/q_i$ values among agents with demand ϕ_i in S, which is monotone with critical value p'_i equal to the highest value among excluded agents. Membership in S'' is again determined by a maximal-in-range mechanism, which is monotone, and the critical value is equal to the VCG price p''_i. We conclude that the whole mechanism is monotone. The critical value is $\max\{p_i, p'_i, p''_i\}$ for membership in S'' and cq_i otherwise, and thus equal to $\min\{cq_i, \max\{p_i, p'_i, p''_i\}\}$ overall. □

Proof of Theorem 2. We assume that budget-balance is violated and show that this assumption contradicts the fact that the first invocation of the VCG mechanism found an efficient solution.

Suppose that the Double-VCG mechanism does not satisfy budget balance, i.e., that $\sum_i p_i < c\alpha(X)$. This means that the average payment per server is less than c, and hence in any way of dividing the total payment among the servers there is at least one server with a payment of less than c. Consider an assignment of the agents in S'' to servers such that each agent is served by the same server(s) in all time periods and each server is fully used; the former is possible as a consequence of demands being over intervals and the latter a consequence of perfect supply. We can thus find a set of agents who together could completely fill a server but whose pro-rated payments for that server sum to less than c, i.e., a set $X' \subseteq S''$ such that

$$\begin{aligned}
&\sum_{i \in X'} \frac{p_i}{q_i} < c \\
&(a_i, d_i) \cap (a_j, d_j) = \emptyset \text{ for all } i \neq j \in X', \text{ and} \\
&\sum\nolimits_{i \in X'} (d_i - a_i) = m.
\end{aligned}$$

For each agent $i \in X'$, there are $LCM(Q)/q_i$ agents in $S \setminus S'$ with the same demand ϕ_i. Let P denote the set of such agents, and observe that the agents in P collectively demand $LCM(Q)$ machines in each time period. Thus $S \setminus P$ satisfies the perfect supply restriction. By construction, any $i \in P$ has a parent agent $\sigma(i) \in X'$ with $\phi_i = \phi_{\sigma(i)}$. Since $i \in S \setminus S'$ and $\sigma(i) \in S''$, $v_i \leq p_{\sigma_i}$. It follows that

$$\begin{aligned}
\sum_{i \in P} v_i &\leqslant \sum_{i \in P} p_{\sigma(i)} \\
&= \sum_{i \in X'} p_i \cdot \frac{LCM(Q)}{q_i} \\
&= LCM(Q) \sum_{i \in X'} \frac{p_i}{q_i} \\
&< c\, LCM(Q),
\end{aligned}$$

and thus $sw(S \setminus P, v) > sw(S, v)$. This contradicts efficiency of S. □

Proof of Theorem 3. We prove the theorem by effectively reducing it to the case where $|Q| = 1$, finding a feasible solution for the second invocation of the VCG mechanism, and showing that this solution satisfies the bound. It then clearly has to hold also for the solution that is actually found.

By adding to S at most $LCM(Q) - 1$ agents with demand $(1, 0, m)$, we can assume that $\alpha(S)$ is a multiple of $LCM(Q)$. Now assume that there are k agents in S' with a particular demand ϕ_i, and combine all but $k \mod LCM(Q)/q_i \leq LCM(Q)/q_i$ of them into super-agents with demand $(LCM(Q), a_i, d_i)$. Do this for all demands, denote by Z the set of agents in S who have not been combined into super-agents, and note that Z contains at most $2LCM(Q)/q_i$ agents of each demand, those left over and those in $S \setminus S'$.

Let X' be a set of super-agents such that $\alpha(X')$ is maximal among all such sets that satisfy the perfect supply restriction. Then $Y = S \setminus X'$ satisfies the perfect supply restriction as

well. In particular, this means that there is a way to assign the superagents from Y to virtual servers of capacity $LCM(Q)$ so that a superagent is always assigned to the same server for its whole interval. By maximality of X', $\alpha(Y) \leq \sum_{0\leq j<m} \sum_{i\in Z \text{ s.t. } a_i\leq j<d_i} q_i$. There are at most $m^2\,|Q|$ possible demands, so $\alpha(Y) \leq m \sum_{\varphi\in\Phi} q_\varphi \cdot LCM(Q)/q_\varphi \leq 2LCM(Q)\,m^3\,|Q|$. Let $Y' = \{i \in Y \mid v_i \geq cq_i\}$, and note that it is cost-effective to serve agents in Y' even if doing so requires buying them their own machine. Meanwhile, X' is a feasible solution for the second invocation of the VCG mechanism. Thus $sw(X) \geq sw(X' \cup Y') \geq sw(S) - sw(Y \setminus Y')$. Since $v_i < cq_i$ for all $i \in Y \setminus Y'$, $sw(Y \setminus Y') \leq c\,m\,\alpha(Y) = 2\,c\,LCM(Q)\,m^4\,|Q|$. Since S is an efficient solution, the Double-VCG mechanism is δ-efficient for the given value of δ. □

B A Tighter Efficiency Analysis

In this section, we will redo the analysis about efficiency in Section 4, seeking for getting a tighter approximation result. Let $\mathcal{S}$ denote the collection of agent sets satisfying perfect supply restriction.

Lemma 1. *If $X \cup Y$ satisfies perfect supply and $q_i = \lambda$ for all agents $i \in X$, we have*

$$\max_{X'\subseteq X: X'\in\mathcal{S}} \alpha(X') \geqslant \alpha(X \cup Y) - \sum_{i\in Y} q_i$$

Proof. By splitting each agent $i \in Y$ into q_i agents, each with demand $(1, a_i, d_i)$, denoted by set Z_1, and padding $X \cup Z_1$ out with $\lambda' < \lambda$ new agents, each with demand $(1, 0, m)$, denoted by set Z_2, we can assume that $\alpha(X \cup Z) = \alpha(X \cup Y) + \lambda'$, where $Z = Z_1 \cup Z_2$, is a multiple of λ while $X \cup Z$ also satisfies prefect supply and $|Z| = \lambda' + \sum_{i\in Y} q_i$. Let $h_S(j) = \sum_{i\in S: j\in[a_i,d_i)} q_i$ denote the number of servers required by agents in S whose demands cover the time slot $[j, j+1)$. By the perfect supply condition for agents $X \cup Z$, we have $h_{X\cup Z}(j) = \alpha(X \cup Z) = \alpha(X \cup Y) + \lambda' \equiv 0(\text{mod}\lambda)$ for any time slot $j \in [m]$. Furthermore, $h_Z(j) = h_{X\cup Z}(j) - h_X(j) \equiv 0(\bmod \lambda)$ since $q_i = \lambda$ for each $i \in X$ by the given condition. Next, we update agent set Z in the following process:

while there exist agents $i, j \in Z$ such that $a_i < a_j \leqslant d_i < d_j$ **do**
  Remove agents i, j from Z;
  Add an agent with demand $(1, a_i, d_j)$ into Z;
  Add another agent with demand $(1, a_j, d_i)$ into Z if $a_j \neq d_i$.
end while

Notice that this process always terminates because in each iteration either the product of demand lengths $\prod_{i\in Z} d_i - a_i$ decreases, resulted by $(d_j - a_i)(a_j - d_i) < (d_i - a_i)(d_j - a_j)$ in the case two agents i and j are replaced by two new agents, or the number of agents in Z is reduced by 1 in the other case. At the end of the process, $|Z| \leq \lambda' + \sum_{i\in Y} q_i$ as the size of Z non-strictly decreases through each iteration, and $h_Z(j) \bmod \lambda = 0$ as $h_Z(j)$ keeps unchanged for each time slot $j \in [m]$ in the whole process. Let $g_Z(\varphi)$ denote the number of agents with demand $\varphi \in \Phi$ in Z. Next, we merging agents in Z to get a set Y' (initialized as an empty set) as follows:

while Z is not empty **do**
  Find an agent $i \in Z$ such that $[a_i, d_i) \subsetneq [a_j, d_j)$ for any $j \in Z$;
  Add $g_Z(\phi_i)/\lambda$ agents with demand (λ, a_i, d_i) into Y';
  Remove all agents in Z with the same demand as agent i.
end while

Notice that we can always choose the agent with longest demand interval as a feasible agent in the first step of each iteration. Consider the case we choose agent i. Let Z' contain every agent $i' \in Z$ if $[a_{i'}, d_{i'})$ is a not-equal-subset of $[a_i, d_i)$. Since no two demand intervals connect

or overlap with each other in Z', there must exist a time slot $[j^*, j^*+1) \in [a_i, d_i)$ that is not covered by any demand interval in Z', i.e., $h_{Z'}(j^*) = 0$. Knowing that the demand interval of each agent in Z is either the subset or independent of the interval $[a_i, d_i)$, we can draw a conclusion that $g_Z(\phi_i) = h_Z(j^*) - h_{Z'}(j^*) = h_Z(j^*)$. Since $h_Z(j) \bmod \lambda = 0$ for each $j \in [m]$ initially, we can prove $g_Z(\phi_i) \bmod \lambda = 0$ in each iteration starting with choosing i (which makes the second step valid in term of adding a integer number of agents) by inductively showing the number of agents removed in the third step, that is $g_Z(\phi_i)$, is always the multiple of λ by the conclusion $g_Z(\phi_i) = h_Z(j^*) \equiv 0 (\bmod \lambda)$ for one time slot $j^* \in [a_i, d_i)$.

Noticing that $h_{Z \cup Y'}(j)$ keeps unchanged during each iteration and $h_{Z \cup X}(j) = \alpha(X \cup Y) + \lambda'$ initially, we have $h_{X \cup Y'}(j) = \alpha(X \cup Y) + \lambda'$ when Z turns into empty set at the end of the merging process, which means $X \cup Y'$ satisfies perfect supply restriction while $\alpha(X \cup Y') = \alpha(X \cup Y) + \lambda'$. Since one agent is added into Y' whenever another λ associated agents are removed from Z during the merging process, we have $|Y'| \cdot \lambda \leqslant \lambda' + \sum_{i \in Y} q_i$.

Since all agents in $X \cup Y'$ are with demand of the same quantity λ, we can get the server allocation for agents in $X \cup Y'$ as if all demand quantities are 1s. By removing all agents served by the servers (each representing λ quantities) shared with at least one agent in Y', the remaining agents, a subset of X, should also satisfy perfect supply restriction. By the maximality, we get $\max_{X' \subseteq X : X' \in \mathcal{S}} \alpha(X') \geqslant \alpha(X \cup Y') - |Y'| \cdot \lambda \geqslant \alpha(X \cup Y) + \lambda' - (\sum_{i \in Y} q_i + \lambda') = \alpha(X \cup Y) - \sum_{i \in Y} q_i$. □

Theorem 4. *The Double-VCG mechanism is δ-efficient for $\delta = 2LCM(Q) \cdot |Q| \cdot cm^3$. In particular, δ is independent of n.*

Proof. In the last paragraph of the previous efficiency proof, we can get a stronger result that $\alpha(Y) = \alpha(S) - \alpha(X') \leqslant \alpha(S) - (\alpha(S) - \sum_{i \in Z} q_i) \leqslant \sum_{i \in Z} q_i$ by using Lemma 1. Then we can continue the proof all the way to the end until getting a new δ-efficiency result where $\delta = 2LCM(Q) \cdot |Q| \cdot cm^3$. □

Theorem 5. *The Double-VCG mechanism is δ-efficient, where δ is $\Omega(LCM(Q) \cdot |Q| \cdot cm^3)$.*

Proof. We are proving the tightness by showing the following instance. Consider the number of time slots $m = 3k$ where k is a integer. Then we construct three subsets of agents together with their demands and values,

$$
\begin{aligned}
&N^{(1)} = \{\langle i, j \rangle : 0 \leqslant i < k < j \leqslant 2k\} \\
&\forall \langle i, j \rangle \in N^{(1)}, \phi_{\langle i,j \rangle} = (1, i, j), v_{\langle i,j \rangle} = c - \varepsilon \\
&N^{(2)} = \left\{\langle i, j \rangle : 2k \leqslant i \leqslant 3k - 1, 1 \leqslant j \leqslant k^2\right\} \\
&\forall \langle i, j \rangle \in N^{(2)}, \phi_{\langle i,j \rangle} = (1, i, i+1), v_{\langle i,j \rangle} = c - \varepsilon
\end{aligned}
$$

Let $N^* = N^{(1)} \cup N^{(2)}$ and let Q is an arbitrary quantity option set. For each $q \in Q$, we revise the quantity for each agent from 1 to q as well as scale the value in the same way, and then we copy every agent $LCM(Q)/q$ times. Let N be the agent set including all these copies for each $q \in Q$. As we can check, the optimal solution welfare can be obtained by including all agents in N. We get the social welfare as follows,

$$
sw(N) \geqslant (k+1)k^2(c - \varepsilon) \, |Q| \, LCM(Q) = \Omega\left(cm^3 \, |Q| \, LCM(Q)\right)
$$

However, since we will remove all agents generated by $N^{(1)}$ through the trade reduction process so that no demand from the remaining agents can cover time slot $[k-1, k)$, the maximal subset of agents under perfect supply restriction would be empty set. Since no agent i is bidding more than cq_i, nobody would be accepted in the end. Hence, the social welfare loss is $sw(N)$, which is $\Omega(LCM(Q) \cdot |Q| \cdot cm^3)$. □

C A Reduction from Exact Cover to VCG with Perfect Supply

EXACT-COVER problem: Given a collection $\mathcal{S} = \{S_1, S_2, ..., S_m\}$ of subsets of a set $X = \{1, 2, ..., n\}$, an exact cover is a subcollection $\mathcal{S}^*$ of $\mathcal{S}$ such that each element in X is contained in exactly one subset in $\mathcal{S}^*$.

REDUCTION: We build up several agents in $N = N^{(S)} \cup N^{(X)} \cup N^{(E)}$ where

$$\begin{aligned} N^{(S)} &= \{1, 2, ..., m\} \\ N^{(X)} &= \{1, 2, ..., n\} \\ N^{(E)} &= \left\{\langle i, j\rangle : i \in N^{(S)}, j \in S_i\right\} \end{aligned}$$

Then we set the demands and values for agents as follows,

$$\begin{aligned} &\forall i \in N^{(S)}, \phi_i = \langle |S_i|, 0, i\rangle, v_i = 0 \\ &\forall i \in N^{(X)}, \phi_i = \langle 1, m+i, m+n+1\rangle, v_i = 1 \\ &\forall i \leqslant m, j \in S_i, k = \langle i, j\rangle \in N^{(E)}, \phi_k = \langle 1, i, m+j\rangle, v_i = 0 \end{aligned}$$

In the same time, let the cost c per server be 0. We are claiming that EXACT-COVER problems has a valid solution if and only if the allocation maximizing social welfare under prefect supply restriction in our VCG optimization provides n welfare, that is,

$$\max_{X' \subseteq N} sw(X', v; \phi) = n$$

In one direction, if there is a solution in EXACT-COVER problem, i.e., we can find a solution $\mathcal{S}^*$ such that

$$\begin{aligned} &\bigcup\nolimits_{S_i \in \mathcal{S}^*} S_i = X \\ &\forall i \neq j, S_i \cap S_j = \emptyset \end{aligned}$$

Then we construct the following subset of agents in N,

$$X' = \left\{i \in N^{(S)} : S_i \in \mathcal{S}^*\right\} \cup \{\langle i, j\rangle : S_i \in \mathcal{S}^*, j \in S_i\} \cup N^{(X)}$$

In order to show the perfect supply restriction being satisfied by solution set X', we are presenting the following equations,

$$\begin{aligned} &\forall 1 \leqslant j \leqslant m+n, \sum_{i \in X' : d_i = j} q_i = \sum_{i \in X' : a_i = j} q_i \\ &\sum_{i \in X' : a_i = 0} q_i = \sum_{i \in X' : d_i = m+n+1} q_i = n \end{aligned}$$

The first part is correct because for any set $S_i \in \mathcal{S}^*$, the associated agent who ends his demand at time instance d_i is attaching to another $|S_i|$ agents in X' who are all starting their demand at the same time instance, i.e., the quantity for both sides is the same, that is $|S_i|$ for both. For the second part, we know the unit of the solution should cover all elements in X at once, so the summation of quantities should also be n.

The equations above imply

$$\forall 0 \leqslant j \leqslant m+n, \sum_{i \in X' : j \in [a_i, d_i)} q_i = n$$

Hence,

$$\alpha(X') = \max_{0 \leqslant j \leqslant m+n} \sum_{i \in X' : j \in [a_i, d_i)} q_i = n$$

In the same time, we have

$$\sum_{i \in X'} q_i(d_i - a_i) = \sum_{0 \leqslant j \leqslant m+n} \sum_{i \in X' : j \in [a_i, d_i)} q_i = (m+n+1)n$$

Therefore, X' satisfies perfect supply restriction while providing the social welfare

$$\sum_{i \in X'} v_i = \sum_{i \in N^{(X)}} v_i = \left| N^{(X)} \right| = n$$

In the other direction, if we can find an agent set X' such that the social welfare of them is n and X' satisfies perfect supply restriction, we can show $\mathcal{S}^* = \left\{ S_i : i \in N^{(S)} \cap X' \right\}$ is also a valid solution in EXACT-COVER problem. The proof idea is basically the same as the other direction. Actually, the following equations, which has been shown above, can be deduced directly by the perfect supply restriction and the welfare being n,

$$\forall 1 \leqslant j \leqslant m+n, \sum_{i \in X' : d_i = j} q_i = \sum_{i \in X' : a_i = j} q_i$$

$$\sum_{i \in X' : a_i = 0} q_i = \sum_{i \in X' : d_i = m+n+1} q_i = n$$

The first equation says that all element that is covered in X is covered by at most once (for those $m+1 \leq j \leq m+n$) and all elements in each set $S_i \in \mathcal{S}^*$ is matching to corresponding elements in X (for those $1 \leq j \leq m$). In the same time, the second equation shows that all element in X is covered by at least one set in $\mathcal{S}^*$. Therefore, $\mathcal{S}^*$ is a solution of EXACT-COVER problem.