Single-Parameter Domains and Implementation in Undominated Strategies

Moshe Babaioff [*], Ron Lavi [†], and Elan Pavlov [‡]

Abstract

This paper studies Algorithmic Mechanism Design where the bidders are "single-parameter" (have the same private value for all desired outcomes), with a focus on Combinatorial Auctions (CA). We study four variants of the CA model: where each player desires a single bundle ("single-minded") or he desires any one of several bundles ("multi-minded"), and where the desired bundles are public information ("known") or private information ("unknown"). We provide several mechanisms and examples in these settings. Our first main result is a general technique to convert any *algorithm* to a truthful ascending *mechanism* for "known" domains (not only for CA). For the "known" single-minded CA domain, it almost preserves the approximation ratio of the original algorithm. Our second main result provides the first computationally efficient mechanism for the case of "unknown" single-parameter multi-minded bidders, with close to optimal welfare approximation. This mechanism also works in general, non single-parameter, combinatorial auctions, with some additional approximation loss. This mechanism is a *computationally feasible implementation in undominated strategies*, a notion that we define and justify, and which we believe is of independent interest.

1 Introduction

The field of Algorithmic Mechanism Design [15] studies the design of computationally efficient algorithms under the assumption that the input is distributed among a set of rational selfish players. One successful approach to this problem is to design *truthful* mechanisms, in which a player always maximizes his utility by simply revealing his true input (incentive-compatible in dominant strategies). The VCG [5, 6, 18] scheme is the only general technique, known to date, to create truthful mechanisms, but it is computationally infeasible in many interesting domains (e.g. Combinatorial Auctions). Numerous papers e.g. [12, 2, 1, 4] written over the past several years demonstrate that the problem of designing truthful, computationally efficient mechanisms can be overcome for domains in which the secret input of each player is *one* single number. Such domains were termed "one parameter domains" [1] by Archer and Tardos [2].

Much of the recent research in the field focuses on *Combinatorial Auctions (CA)*: m heterogenous items are to be partitioned among n players and the seller, and each player values (either one or several) subsets of items ("bundles") [2]. This model generalizes many classic resource allocation

[*]mosheb@cs.huji.ac.il. School of Engineering and Computer Science, The Hebrew University of Jerusalem, Israel.
[†]ronlavi@ist.caltech.edu. SISL/IST, California Institute of Technology.
[‡]elan@cs.huji.ac.il. School of Engineering and Computer Science, The Hebrew University of Jerusalem, Israel.

[1]The one parameter domains of [2] are somewhat more general.

[2]The seller has zero valuation for the goods. Players value the empty set for zero, each player care only about the bundle he gets and his valuation is monotonic in the bundles.

problems, e.g. scheduling, network routing, and more. We distinguish between four categories of Single-Parameter (SP) CAs, defined by two orthogonal characteristics. The first characteristic concerns each player set of desired outcomes. In a *Single-Minded (SM)* CA, each player i has a value of $\bar{v}_i > 0$ for any bundle containing one specific bundle $\bar{s}_i$ he desires (and zero for any other bundle). In contrast, in a single-parameter *Multi-Minded (MM)* CA, each player has a set of desired bundles (not all containing the same desired bundle), all with the same value, and other bundles have zero value. (A natural example for SP multi-minded CA is the Edge-Disjoint Paths problem, we later describe.) The second characteristic concerns the information model. In a *"Known" CA* domain, the set of bundles that each player desires is publicly known and the value $\bar{v}_i$ is the only private information of player i, while in an *"Unknown" CA* domain, both the set of bundles as well as $\bar{v}_i$ are private information of player i.

The unknown single-minded (USM) domain was suggested and first studied by Lehmann, O'Callaghan, and Shoham [12], which presented a specific greedy mechanism that is incentive-compatible in dominant strategies and achieves a $\sqrt{m}$-approximation to the social welfare (the sum of the players valuations for the outcome). They also showed that the algorithmic bound (even without taking incentives into account, and when all players have $\bar{v}_i = 1$) is $m^{1/2-\epsilon}$ for any $\epsilon > 0$ (unless ZPP=NP). Mu'alem and Nisan [13] looked at the Known Single-Minded (KSM) domain, this enabled them to achieve an $\epsilon\sqrt{m}$-approximation for any fixed ϵ. In order to break the lower bound for KSM domain, Archer et al. [1] relaxed the solution concept and assumed that there are multiple copies of each good. We are not aware of any positive results (non-trivial computationally efficient mechanisms that guarantees good welfare approximation) for general CA, or for the easier case of unknown single parameter multi-minded CA.

1.1 Our Results

We start our study with Known Single-Parameter (KSP) domains. We present a general method to convert **any** given *algorithm* into a truthful ("value monotonic") ascending *mechanism* for *any* KSP domain (not only CA). However, for combinatorial auctions with single minded bidders, we are also able to show that this method approximately preserves the original approximation ratio. This is especially useful since many well-known techniques for this problem domain are not truthful to begin with (e.g. LP-based or branch and bound methods):

Theorem. *Any (deterministic) algorithm for a single-parameter domain can be converted into a truthful (deterministic) mechanism for KSP bidders.*

Additionally, any c-approximation algorithm for single-minded CA can be converted into a truthful $O(log(\bar{v}_{max}) \cdot c)$-approximation mechanism for KSM bidders. [3] [4]

At first look, it seems that our result is inferior to the $\sqrt{m}$-approximation for KSM of [12], which is optimal. Nevertheless, Sandholm [17] enumerates many special cases (where there is some additional structure on the bids) in which there exist algorithms defeating this bound (e.g. when each bid has at most ω items, when each bid shares items with at most Δ other bids, etc.). In each such a case, our method creates a truthful mechanism (even if the original algorithm is not "value monotonic") with only $O(\ln \bar{v}_{max})$ additional approximation loss! We also get the same loss for a "combined" algorithm, which takes the best allocation over some set of polynomial time

[3] Throughout this paper we assume that for any i, $\bar{v}_i \geq 1$ (that is, the minimal *a-priori* value is at least 1) and we denote by $\bar{v}_{max}$ the maximal observed value of any player.

[4] The mechanism is also an $O(min(m, n))$-approximation.

approximation algorithms (randomized or deterministic, general or for special cases). Moreover, if the algorithm is randomized, then the resulting mechanism is randomized, nevertheless, each player has a dominant strategy for *any* realization of the coins (which is stronger than having a strategy that maximizes the expected utility).

This method also improves upon a previous similar result[5] of Awerbuch, Azar, and Meyerson [3], in two aspects: (1) if the original algorithm is deterministic, then our mechanism is also deterministic, and, (2) our method does not require a-priori knowledge of $\bar{v}_{max}$. One additional interesting property of our method is that it produces a mechanism with the classic economic structure of an ascending auction: players compete by gradually increasing their offers, and the winners pay their bid. It has been argued many times [16] that such indirect mechanisms should be preferred over the more common "direct revelation" mechanisms, where players simply reveal their true value.

Next ,we turn to multi-minded CA domains. Although the above technique creates a truthful mechanism for the KSP multi-minded CA domain, its approximation ratio is poor. We provide a truthful mechanism based on the greedy mechanism of Lehmann et al. [12] that achieves a $\sqrt{m}+1$-approximation for KSP multi-minded CA. We then concentrate on the Edge-Disjoint Paths (EDP) domain, which is a natural SP multi-minded CA domain – each player desires a path from his source node to his target node in some underlying graph with m edges (goods), where any such path is worth $\bar{v}_i$ to him. Even though the number of such paths might be exponential, we can modify our mechanism for KSP multi-minded players mentioned above, to run in polynomial time, and keep the same approximation ratio [6]. Although the original mechanism of [12] was truthful for the unknown (single-minded) domain, it turns out that our modification is not truthful for unknown SP multi-minded CA. We provide a characterization of direct revelation mechanisms that are truthful for general Unknown Single-Parameter (USP) domains[7] (not only CA domains), and show that the above mechanism violates our characterization. Our characterization generalizes the characterization of [2] for the KSP domains, as well as the characterization of [12] for USM CA, and sheds some surprising light on the requirements needed for truthfulness for the general USP domains.

Like all previous papers, we were not able to provide a dominant strategies truthful mechanism for the multi-minded CA domains, and in particular to the USP multi-minded CA domain. In order to achieve positive results for this case we turn to the previously studied game-theoretic concept of *"implementation in undominated strategies"* [9]: Let $u_i(s_i, s_{-i})$ denote the resulting utility of player i when he plays the strategy s_i and the other players play s_{-i}. A strategy s'_i of player i is dominated by another strategy s_i if, for every s_{-i}, $u_i(s_i, s_{-i}) \geq u_i(s'_i, s_{-i})$, and this inequality is strict for at least one specific s_{-i}. A strategy is *undominated* if it is not dominated by any other strategy. By the concept of implementation in undominated strategies, there is a non-empty *set* of strategies that jointly dominate all the other strategies, rather than one single strategy that dominates all the rest. This set of strategies is the set of all undominated strategies. The game-theoretic analysis will not be able to exactly tell us which strategy out of this set will be chosen by the player, but ensures that the chosen strategy will belong to this set. More formally, we define the concept of a (computationally efficient) "c-approximation in undominated strategies" as follows:

Definition. *A mechanism M implements a* c-approximation in undominated strategies *if:*

[5]their result is for the online setting while ours is for the offline setting.

[6]This is an asymptotically optimal result, since for directed graphs, it is NP-hard to obtain an approximation ratio of $\Omega(m^{1/2-\epsilon})$ for any $\epsilon > 0$ [7]

[7]One of the contributions of this paper is a formal definition of the general (non CA) USP domains.

- *For* **any** *combination of undominated strategies of the players, M outputs a c-approximation outcome, in polynomial time.*
- *A player can never end up with a negative utility when playing any undominated strategy (i.e. the mechanism is ex-post individually rational).*
- *For any dominated strategy, a player can efficiently compute an undominated strategy that dominates it.*

We argue that this concept captures all the truly important ingredients of the computationally efficient dominant strategies solution concept: First, it does not require any form of coordination among the players (unlike Nash equilibrium), or that players have any assumptions on the rationality behavior of the others. The only assumption is that a player does not play a dominated strategy – indeed, given such a strategy he can easily compute another strategy, that guarantees the at least the same utility even in the worst-case! Thus, we think that this assumption is as reasonable as the assumption that a player will play his dominant strategy, if he has one. Second, we ensure that players have no risk in participating in the auction, as their resulting utility cannot be negative, and, finally, we strictly maintain all aspects of computational feasibility. This game-theoretic relaxation forces us, on the other side of the equation, to strengthen the algorithmic analysis, as we now have to show that the resulting outcome of the mechanism will be approximately optimal for *every* choice of undominated strategies that the players make, and not only for the specific choice of dominant strategies, as in truthful mechanisms. Such a trade-off has been recently suggested by Lavi and Nisan [11], but their game-theoretic notions were significantly weaker than the ones suggested here.

It should be noted that, although implementation in undominated strategies is a well-known game-theoretic concept, very few positive results have been achieved by using it [9]; We are not aware of any positive results in this framework for games with incomplete information, and, in particular, we are the first to use it in the context of computationally efficient approximation algorithms with worst-case guarantees. We believe that our definitions and techniques will turn out to be useful for many other settings as well, thus offering a new method to bypass the inherent difficulties of the truthfulness concept, in a way that suits the worst-case notions of the CS literature.

Using this concept, we design the "1-CA Japanese Mechanism" for combinatorial auctions with single parameter (multi-minded) players:

Main Theorem. *The 1-CA Japanese Mechanism implements an $O(log^2(\bar{v}_{max})\sqrt{m})$ approximation in undominated strategies, for the Unknown Single-Parameter multi-minded CA domain.* [8]

To the best of our knowledge, this is the first non-trivial computationally efficient mechanism for the USP multi-minded CA domain. The mechanism achieves an approximation guarantee for the social welfare that is close to the computational bound of $m^{1/2-\epsilon}$. Nisan and Segal [14] proved that $\min(n^{1-\epsilon}, m^{1/2-\epsilon})$ is an upper bound on the approximation achievable in polynomial communication[9] (even for SP multi-minded CA with $\bar{v}_i = 1$ for any player i). Our result almost matches this bound. The "1-CA Japanese Mechanism" is in fact more robust, and handles players that are "approximately single parameter": For each player, the ratio between his maximal value for any bundle and his minimal positive value for any bundle is at most δ (*any* combinatorial

[8] The mechanism is also an $O(min(m, n))$-approximation.

[9] independent of the computational complexity, it holds even if ZPP=NP.

auctions satisfies this with the appropriate value of δ). The proof shows that, for such players, given that they play *any* combination of undominated strategies, the resulting outcome will be an $O(\delta \cdot log^2(\bar{v}_{max}) \cdot \sqrt{m})$ approximation to the optimal welfare.

We also present an additional, different technique for implementation in undominated strategies. This technique makes formal the idea that by a careful extension of the strategy space, we can allow players to place "lies" about their bundles, in addition to the truth revelation. We exploit this technique to close an exponential gap in the approximation ratio left by Babaioff and Blumrosen [4] between the cases of "known" and "unknown" single-minded CA for axis-parallel rectangles in the plane.

The rest of this paper is organized as follows. In section 2 we define single parameter domains, give necessary and sufficient conditions for truthfulness, and discuss the specific EDP example. In section 3 we focus on the case of "known" domains, describing our technique to convert approximation algorithms to truthful mechanisms, and section 4 describes our main results about implementations in undominated strategies. All proofs and some additional results appear in the Appendix.

2 Single Parameter Domains and Truthful Mechanisms

In a *Single-Parameter Domain*, there is a finite set of players N ($|N| = n$) and a set of outcomes Ω. Each player $i \in N$ has a *value* $\bar{v}_i \in \Re_{++}$ (we assume that $\bar{v}_i \geq 1$), and a *satisfying set* $\bar{A}_i \subset \Omega$. The interpretation is that i obtains a value $\bar{v}_i$ from any outcome $\omega \in \bar{A}_i$ (in this case we say that i is *satisfied* by ω), and 0 otherwise. The set $\bar{A}_i$ belongs to a predefined family of valid outcome subsets $\mathcal{A}_i$. Let $\bar{t}_i = (\bar{v}_i, \bar{A}_i) \in \Re_{++} \times \mathcal{A}_i$ denote the *type* of player i, and $\mathcal{T} = \Re^n_{++} \times \mathcal{A}$ denote the domain of all player types, where $\mathcal{A} = \mathcal{A}_1 \times, \ldots, \times \mathcal{A}_n$.

For example, the Edge-Disjoint Paths (EDP) domain described in the Introduction is a single parameter domain: if player i desires a path from s_i to t_i (his type is a value $\bar{v}_i$ and some specific pair of nodes s_i and t_i), then A_i is the set of all allocations in which i receives a bundle that contains a path from s_i to t_i. $\mathcal{A}_i$ is the set of all such A_i, for any possible pair of source-target nodes. Although this example is a special case of combinatorial auctions, it clarifies the difference between single minded players (which are players that desire only one specific subset of items) and single parameter players: the players in the EDP problem are not single minded, as any path between their source and target nodes will satisfy them, but they are indeed single parameter.

Definition 1 ("Known" and "Unknown" Single Parameter domains)**.** *A Known Single Parameter (KSP) domain $\mathcal{T}$ is a single parameter domain in which all information is public, except the players' values $\{\bar{v}_i\}_{i \in N}$: each $\bar{v}_i$ is known only to player i.*

An Unknown Single Parameter domain (USP) $\mathcal{T}$ is a single parameter domain in which the entire type (value and satisfying set) is known only to the player himself (private information).

A *strategic mechanism* M is constructed from a *strategy space* $\mathcal{S} = \mathcal{S}_1 \times, \ldots, \times \mathcal{S}_n$, an *allocation rule* $G : \mathcal{S} \to \Omega$, and a *payment rule* $P : \mathcal{S} \to \Re^n_{++}$. Each player i acts strategically in order to maximize his *utility*: $u_i(s, \bar{t}_i) = v_i(G(s), \bar{t}_i) - P_i(s)$. In *direct revelation mechanisms*, players are required to reveal their type, i.e. $\mathcal{S}_i = \mathcal{T}_i$.

Definition 2 (Truthfulness)**.** *A direct revelation mechanism M is* truthful *if, for any i, any true type $\bar{t}_i \in \mathcal{T}_i$, and any reported types $t \in \mathcal{T}$: $u_i((\bar{t}_i, t_{-i}), \bar{t}_i) \geq u_i(t, \bar{t}_i)$, and $u_i((\bar{t}_i, t_{-i}), \bar{t}_i) \geq 0$.*[10]

[10]Throughout the paper we use the notation $x_{-i} = (x_1, ..., x_{i-1}, x_{i+1}, ..., x_n)$.

An allocation rule is truthfully implementable *(in dominant strategies) if there are payments P such that the direct revelation mechanism $M = (G, P)$ is truthful.*

The following monotonicity condition, which was repeatedly identified in recent years ([12, 2, 13], etc...), completely characterizes truthful mechanisms for KSP domains.

Definition 3 (Value monotonicity)**.** *An algorithm G is* value monotonic *if for all $i \in N$, $A \in \mathcal{A}$, and $v \in \Re^n_{++}$: if $G(v, A) \in A_i$ and $v'_i > v_i$ then $G(v', A) \in A_i$, where $v' = (v'_i, v_{-i})$.*

The definition of critical value of [12] is phrased, in our notation, as:

Observation 1 (Critical value)**.** *If an algorithm G is value monotonic then for all $i \in N$, $A \in \mathcal{A}, v_{-i} \in \Re^{n-1}_{++}$, there exists a* critical value *$c_i(A_i|v_{-i}, A) \geq 0$, such that,*

- *if $v_i > c_i(A_i|v_{-i}, A)$ then $G(v, A) \in A_i$ ("i is a winner").*
- *if $v_i < c_i(A_i|v_{-i}, A)$ then $G(v, A) \notin A_i$ ("i is a loser").*

The critical value $c_i(A_i|v_{-i}, A)$ is the "minimal" value player i needs to report in order for an outcome in A_i to be chosen, when the bids are $A \in \mathcal{A}$ and $v_{-i} \in \Re^{n-1}_{++}$. In a *normalized* mechanism for single-parameter domain, the payment rule ensures that a loser i pays 0.

Theorem 1 ([12, 2, 13], etc...)**.** *A direct revelation normalized mechanism for a KSP domain is truthful if and only if its allocation rule G is value monotonic, and any winner i pays $c_i(\bar{A}_i|v_{-i}, \bar{A})$.*

Note that the critical value of each winner can be calculated efficiently using a binary search.

As an example for a monotonic allocation rule for a general single parameter domain, we analyze the EDP problem. Although this is a special case of the more general combinatorial auction problem, we cannot use any of the existing algorithms for that problem, as (i) they are designed for single minded bidders, and, (ii) when transformed to a multi minded setting in the straightforward way, they require each player to declare all his bundles. In Figure 1 we present a polynomial time allocation algorithm for the EDP problem[11]. This is a modification to the greedy algorithm of [12] for CA with unknown single minded bidders. Our modification is polynomial even though the number of satisfying bundles of a player might be exponentially large, as there may be an exponential number of paths between the source and target nodes.

Theorem 2. *For the "known" EDP domain, the EDP greedy allocation algorithm is truthful, and obtains a $(\sqrt{m}+1)$-approximation in polynomial time.*

Unfortunately, designing truthful mechanisms for unknown domains seems to be much harder. The greedy algorithm of [12] is one of the rare examples for truthfulness in unknown domains. Somewhat surprisingly, it turns out that the above modification to it, for the EDP problem, stops being truthful for the unknown case:

Proposition 1. *The EDP Greedy Mechanism is not truthful for the Unknown EDP problem.*

[11] Figure 6 in the Appendix presents a similar mechanism for the general SP multi-minded CA domain. Unlike the EDP mechanism, this mechanism running time is dependent on the number of desired bundles.

The EDP greedy allocation algorithm:
Input: For each player i, a value v_i and a pair of source-target nodes (s_i, t_i).
Allocation:

1. Let $E_0 = E, j = 0, W = \emptyset$.
2. Calculate the all pairs shortest path for the graph $G_j = (V, E_j)$.
   Let d_i^j be the distance (number of edges) of the shortest path p_i^j from s_i to t_i in G_j.
3. If there exist a player $i \notin W$ with $d_i^j \leq |V|$ (finite distance), then
   let $W = W \cup \{i\}$ where player i has the maximal value of $v_i / \sqrt{d_i^j}$.
   Let $E_{j+1} = E_j \setminus p_i^j, j = j + 1$. Return to stage 2.

Figure 1: The EDP greedy allocation algorithm. It is value monotonic, thus can be made truthful for the KSP domain in which players only report their value. It is not truthful for "Unknown" players.

This is quite surprising, as one can show that the following claim still holds: if a player is satisfied with some type declaration then he will still be satisfied with his true type declaration. Although this condition characterizes truthful mechanisms for "unknown" single minded bidders, it is, apparently, not enough in general. Indeed, the necessary and sufficient condition is a bit more subtle:

Definition 4. *An algorithm G* **ensures minimal payments** *if it is value monotonic, and, for all $i \in N, A \in \mathcal{A}, v \in \Re^n_{++}, \bar{A}_i \in \mathcal{A}_i$, if i bids $v_i > c_i(A_i|v_{-i}, A)$ and A_i such that $G(v, A) \in A_i \cap \bar{A}_i$ (i is satisfied and he wins) then $c_i(\bar{A}_i|v_{-i}, (\bar{A}_i, A_{-i})) \leq c_i(A_i|v_{-i}, A)$.*

We also need a technical, but important, condition, which is w.l.o.g for combinatorial auctions and all their special cases:

Definition 5. *An algorithm G* **ensures losers unsatisfaction** *if for all $i \in N, A \in \mathcal{A}, v \in \Re^n_{++}, \bar{A}_i \in \mathcal{A}_i$, if $G(v, A) \notin A_i$ then $G(v, A) \notin \bar{A}_i$.*

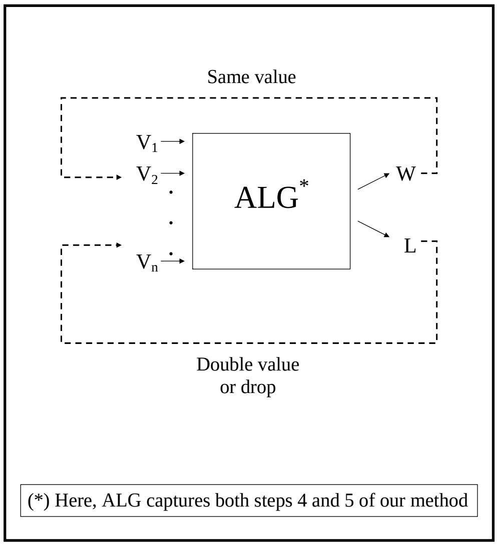
Theorem 3. *An allocation rule G is truthfully implementable (in dominant strategies) for USP domains if and only if it ensures minimal payments and it ensures losers unsatisfaction.*

As in the KSP domain, the appropriate payments are by critical value.

3 The Japanese Wrapper Mechanism for KSP domains

In this section we present a method to obtain a truthful approximation mechanism from any algorithmic procedure (given as a black-box), for a KSP domain. This method converts a given procedure to an ascending Japanese auction (players compete by gradually raising their bids; winners pay their bids). If the number of iterations is small then the original approximation ratio does not deteriorate too much. This method will also be used as basic building blocks for our constructions for Unknown Single Parameter domains in section 4 below.

A formal description of our method is given in Figure 2. Let us now describe it informally: Suppose that *ALG* is an algorithmic procedure for some Known Single Parameter domain, that, when given as input a set of player values, outputs a c-approximation to the social welfare. The Japanese Wrapper Mechanism is a simple wrapper to *ALG*: A vector of player values, initialized to



The Japanese Wrapper Mechanism:

Allocation:

1: Let $j = 0$, $W_j = \emptyset$, $L_j = \emptyset$.
2: For any player $i \in N$ let $v_i^0 = 1$.
3: While $(W_j \cup L_j \neq N)$
4: $\quad W_{j+1} = ALG(v^j)$
5: $\quad$ if $v(W_{j+1}, v^j) < v(W_j, v^j)$ then $W_{j+1} = W_j$.
6: $\quad$ For any $i \notin W_{j+1}$,
$\qquad i$ chooses $v_i^{j+1} = ev_i^j$ or $v_i^j = 0$ (retire).
7: $\quad L_j = \{i \in N | v_i^j = 0\}$, j=j+1
8: J=j

Payments:

Each winner $i \in W_J$ pays his reported value v_i^J, losers pay 0.

Figure 2: The Japanese Wrapper Mechanism for KSP domains

$v^0 = \vec{1}$, is iteratively handed in as input to ALG, who, in return, outputs a set of winners ω_j (where j is the iteration number). The new allocation W_{j+1} is taken to be the one with the maximal value among W_j, ω_j (step 5). Every loser is then required to either increase his value by a factor of e (e is the base of the natural logarithm) or to permanently drop (this is denoted by $v_i^j = 0$). This is iterated until all remaining players are declared winners by ALG. This set of players is the set of winners of the mechanism. Each of the winners pays his last bid, v_i^J, where J denotes the total number of iterations. It can be easily verified that, if ALG has polynomial running time then the Japanese Wrapper Mechanism also has a polynomial running time.

Proposition 2. *The dominant strategy of any player is to increase his reported value when asked, as long as the increased value is lower then his true value $\bar{v}_i$.*

Note that this is true even if the underlying algorithm, ALG, is **not** value monotonic. Indeed, many natural methods are not value monotonic, e.g. taking maximum over two algorithms is not monotonic in general.

In order to analyze the approximation bounds of this resulting mechanism, the crucial point is that the number of iterations will be low:

Lemma 1. *Assume that ALG is a c-approximation algorithm. Then the Japanese Wrapper Mechanism achieves an $(e \cdot c \cdot J + 1)$-approximation in dominant strategies, where J is the number of iterations performed.*

For single-minded CA domain, we are able to bound the maximal number of iterations: even in the worst case, $J \leq 2 \cdot \ln \bar{v}_{max} + 1$. [12] Interestingly, this bound on the number of iterations is independent of the number of players. Thus we have:

Theorem 4. *Given any c-approximation algorithm for single-minded CA, the Japanese Wrapper Mechanism obtains an $O(\ln \bar{v}_{max} \cdot c)$-approximation in dominant strategies, in polynomial time.*[13]

[12]This assumes, without loss of generality, that the algorithm always outputs a set of winners that is maximal to containment.

[13]With minor changes to the algorithm the mechanism also becomes an $O(min(m, n))$-approximation.

In Appendix B.1 we give an abstract condition that enables us to bound the maximal number of iterations in single-minded CA, and derive results for other domains. Unfortunately, we also show, by example, that in combinatorial auctions with double minded players, the number of iterations that our mechanism may perform may be even $n \ln \bar{v}_{max}$. However, by refining our techniques, we show a similar method for the general single parameter (multi-minded) CA domain, that achieves $O(\ln^2 \bar{v}_{max} \cdot \sqrt{m})$-approximation using some specific $\sqrt{m}$-approximation algorithm.

4 "Unknown" Single-Parameter Combinatorial Auctions and Undominated Strategies Implementation

In this section we present our main results – implementations in undominated strategies for unknown general single parameter players. We begin with a formal discussion of the concept of implementation in undominated strategies.

Definition 6. *A strategy $s_i \in \mathcal{S}_i$* **dominates** *the strategy $s'_i \in \mathcal{S}_i$ of player i with type $\bar{t}_i$, if for any $s_{-i} \in \mathcal{S}_{-i}$, $u_i((s_i, s_{-i}), \bar{t}_i) \geq u_i((s'_i, s_{-i}), \bar{t}_i)$, and this inequality is strict for at least one instance of $s_{-i} \in \mathcal{S}_{-i}$. A strategy $s_i \in \mathcal{S}_i$ is* **undominated** *for player i if it is not dominated by any other strategy of player i.*

Clearly, a player that aims to maximize his utility will not play a dominated strategy, if he is able to quickly find a strategy that dominates it:

Definition 7. *A strategic mechanism has the* **fast undominance recognition property** *if, given any strategy $s'_i \in \mathcal{S}_i$ of any player i, there exists a polynomial-time procedure that determines if s'_i is undominated, and, if not, finds an undominated strategy $s_i \in \mathcal{S}_i$ that dominates s'_i.*

We will also require that a player will never regret taking part in the mechanism, i.e. that *any* undominated strategy will *ensure* a non-negative utility:

Definition 8. *A mechanism M is* **Ex-post individually rational** *if for any $i \in N, \bar{t}_i \in \mathcal{T}_i$, if player i has type $\bar{t}_i$ then (1) there exists at least one undominated strategy for player i.[14] (2) for any undominated strategy $s_i \in \mathcal{S}_i$, $u_i((s_i, s_{-i}), \bar{t}_i) \geq 0$ for all $s_{-i} \in \mathcal{S}_{-i}$.*

We are now ready the define the main new concept we suggest in this paper.

Definition 9. *A strategic mechanism M* **implements a c-approximation in undominated strategies** *if for any types $\bar{t} \in \mathcal{T}$:*

- *For* any *tuple of undominated strategies $s \in \mathcal{S}$ (i.e. s_i is undominated for any i), M outputs a c-approximation outcome with respect to $\bar{t}$ in polynomial time.*
- *The mechanism has the fast undominance recognition property.*
- *The mechanism is Ex-post individually rational.*

[14]if the number of strategies is finite, this always holds.

The Japanese Wrapper Mechanism for "Unknown" CAs:

Allocation:

1. Let $j = 0$, $W_j = \emptyset$, $L_j = \emptyset$. For any player $i \in N$ let $v_i^0 = 1$, $s_i^0 = G$.
2. While $(W_j \cup L_j \neq N)$:
   - $(W_j, s^{j+1}) = Proper\_Sub\_Game(v^j, s^j, W_{j-1})$.
   - For any $i \notin W_{j+1}$, i chooses if $v_i^{j+1} = ev_i^j$ or $v_i^j = 0$ (retire).
   - $L_j = \{i \in N | v_i^j = 0\}$, j=j+1.
3. J=j.

Payments:

Each winner $i \in W_J$ receives the bundle s_i^J and pays his last bid v_i^J, a loser receives $\emptyset$ and pays 0.

Figure 3: The Japanese Wrapper Mechanism for combinatorial auctions with multi-minded players.

Mechanisms in dominant strategies are a special case of this definition, as their set of undominated strategies contains exactly one strategy: the dominant strategy. This game theoretic relaxation allows the mechanism designer to "leave in" several undominated strategies, from which the actual strategy that the player will choose is not known. However, it also implies an increasing algorithmic difficulty, as the outcome must be an approximately optimal one, for any choice of undominated strategies that the players make. The definition also includes a direct computational explanation as to why the players will indeed choose one of their undominated strategies: given any *dominated* strategy that some player considers as an option, it is computationally easy to find another, *undominated*, strategy. That strategy is guaranteed to make the player better-off, *in the worst case.*

4.1 The Japanese Wrapper Mechanism revisited

In this section, we enhance our Japanese Wrapper Mechanism in order to implement an $O(log^2(\bar{v}_{max}) \cdot \sqrt{m})$ approximation in undominated strategies, for combinatorial auctions with unknown (single-parameter) multi-minded players.

The first step to handle the "unknown" situation is to maintain an "active bundle" for each player. In a similar manner to the current value maintained by the Japanese Wrapper Mechanism, this "active bundle" will keep approaching to one of the true bundles of the player. Figure 3 essentially describes the original Japanese Wrapper Mechanism, with this additional "active bundle" feature. Since every winner will get his active bundle and will pay his last bid, players will bid of to their value and will never report a bundle that they do not value.

However, it turns out that we cannot just "plug-in" any algorithm to this modified wrapping procedure. To ensure the approximation ratio, we require the following condition on the active bundle: if a player is about to retire, and knows "for sure" that he will lose if his bundle will not change, then, if he has some desired bundle which is available, he will bid on it. This rather weak condition still enables us to upper bound the value of the losers by the value of the winners. Unfortunately, most procedures will not satisfy it. This is because, even in the unknown single-minded case, a player with a true value has sometimes an incentive to declare a bundle that strictly contains his own. Some types of "proper-subgames" (see Appendix C), however, does enable us

The 1-CA-SUB sub-game:

Input:

A vector of values v^j and a vector of bundles s^j (with one element for each player).
An allocation W_{j-1} which is legal w.r.t. s^j.

Procedure:

1. $W_j(1) = \text{argmax}_{i \in N}\{ v_i^j \}$; $W_j(2) = \emptyset$.
2. Go over the players with $v_i^j > 0$ in descending order of values. Let player i be the current player.
   - If $i \notin W_{j-1}$ allow him to pick a bundle[a] $s_i^{j+1} \subseteq Free(W_j(2), s^{j+1}) \cap s_i^j$ such that $|s_i^{j+1}| \leq \sqrt{m}$, in any other case ($i \in W_{j-1}$ or i does not pick) $s_i^{j+1} = s_i^j$.
   - If $|s_i^{j+1}| \leq \sqrt{m}$, add i to any of the allocations $W \in \{W_{j-1}, W_j(1), W_j(2)\}$ for which $s_i^{j+1} \subseteq Free(W, s^{j+1})$.

Output:

The vector of bundles s^{j+1} and an allocation $W \in \{W_{j-1}, W_j(1), W_j(2)\}$ with maximal value of $\sum_{i \in W} v_i^j$.

[a] $Free(W_j(2), s^{j+1})$ denotes the set of free goods, goods that are not in $\cup_{i \in W_j(2)} s_i^{j+1}$

Figure 4: The 1-CA-SUB sub-game for combinatorial auctions. This sub-game is a "proper" sub-game.

to proceed. A main example for this is the procedure in Figure 4. This is essentially the 1-CA procedure[15] with the following twist: players are not required to declare their bundles in advance, but are rather approached in a certain carefully chosen time point in the middle of the run time of the procedure (this is step $2a$ in the figure). As we show in the appendix, this time point does satisfy the above requirement.

For the case of unknown single minded bidders we can show that the 1-CA Japanese Mechanism essentially reduces to the "known" case discussed in section 3:

Theorem 5. *The 1-CA Japanese Mechanism implements an $O(log(\bar{v}_{max}) \cdot \sqrt{m})$ approximation in undominated strategies for unknown single-minded combinatorial auctions.*[16]

For the case of unknown multi-minded bidders, the use of an active bundle forces a (temporary) winning player to focus on one specific bundle; if that specific bundle turns out to be highly demanded, he cannot switch and insert new items to his bundle. We are able to show that this will not cause the approximation ratio to deteriorate by more than an additional $O(log(\bar{v}_{max}))$ factor:

Theorem 6. *The 1-CA Japanese Mechanism implements an $O(log^2(\bar{v}_{max}) \cdot \sqrt{m})$-approximation in undominated strategies for combinatorial auctions with unknown single parameter multi minded players.*[17]

As a result, we get "for free" a solution to the EDP problem, which is a special case of a combinatorial auction with unknown single parameter multi minded players. Notice that our method also solves the problematic situation where a player has an exponential number of bundles (paths), as he need not declare all of them, but only to gradually focus on one of them.

[15] The basic algorithm is by [8]. It was used by [13] to create a truthful mechanism for known single-minded CA.
[16] The mechanism is also an $O(min(m, n))$-approximation.
[17] The mechanism is also an $O(min(m, n))$-approximation.

Impersonation-Based Mechanism for the USM CA model
Given a direct revelation allocation rule G for single minded CA (each player i bids a value v_i and a bundle s_i), and a positive integer K.

Strategy space:
Each player i submits a value v_i and a sequence of $k_i \leq K$ bundles, $s_i^1 \subset s_i^2 \subset \ldots \subset s_i^{k_i}$.

Allocation:
Run the allocation rule G on the input $\{(v_i, s_i^k)\}_{i \in N, k=1\ldots,k_i}$ and get a set of winners W. Each winner $i \in W$ receives s_i^1 (the minimal bundle), other players lose, and get $\emptyset$.

Payments (assuming the Impersonation-Based allocation is value monotonic):
Losers pay 0. Each winner pays his critical value for winning.

Figure 5: The Impersonation-Based Mechanism for unknown single-minded CA.

This result extends to the general (not single-parameter) Combinatorial Auctions domain, where each player i has a possibly different valuation for each bundle $b \subseteq G$. We define the *maximal value difference* δ to be $\delta = \max_{i \in N} v_i(G)/v_i(b_i^{min})$, where $v_i(G)$ is the valuation of player i for G, and $v_i(b_i^{min})$ is the value of the lowest positive valued bundle of player i (still $v_i(b_i^{min}) \geq 1$). Our proof shows that the "1-CA Japanese Mechanism" mechanism implements an $O(\delta \cdot log^2(\bar{v}_{max}) \cdot \sqrt{m})$ approximation in undominated strategies for general CA domain.

4.2 An Impersonation-Based Mechanism for Unknown Single Minded CA

For some allocation rules in the case of unknown single minded bidders, we can have an implementation in undominated strategies that actually keeps the algorithm's original approximation ratio. As a by-product to this, we demonstrate a completely different technique to create undominated strategies mechanisms for combinatorial auctions with single minded players. This technique is given in figure 5. The main idea is to allow players to impersonate to several different single minded players, all with the same value, but with different bundles. It turns out that, for some carefully designed allocation rules which we demonstrate below, the players will still reveal their true value and their true bundle (coupled with extra "false" information), and, furthermore, this will enable us to maintain the original approximation ratio.

Let $M(G, K)$ denote the mechanism created by running the Impersonation-Based Mechanism with an allocation rule G and an integer K. We note that it is well defined if and only if it is value monotonic.

Proposition 3. *Assume that for some allocation rule G and an integer K, the $M(G, K)$ mechanism is value monotonic. Then for any player i, in any undominated strategy, $s_i^1 \supseteq \bar{s}_i$ and $v_i = \bar{v}_i$.*

Theorem 7. *Suppose that an allocation rule G is a c-approximation for KSM-CA. If the $M(G, K)$ mechanism is value monotonic and in any undominated strategy, each player i reveals his true bundle ($s_i^1 = \bar{s}_i$), then $M(G, K)$ implements a c-approximation in undominated strategies.*

We next give two examples for an allocation rules that satisfy both these conditions[18].

[18] It is interesting to note that is not sufficient that the allocation rule itself will be value monotonic for the impersonation-based mechanism to be monotonic. See the appendix for details.

An axis-parallel CA of the plane. In an axis-parallel rectangles CA, the set of goods is the set of points in the plane $\Re^2$, and each bundle a player may desire is an axis-parallel rectangle. Babaioff and Blumrosen [4] used an algorithm by [10] to give a truthful $O(log(R))$ approximation mechanism for known single minded bidders, where R is the ratio between the smallest width and the largest height of the given rectangles. However, for the "unknown" case they were only able to give an $O(R)$ approximation. With the Impersonation-Based mechanism technique, we achieve the original $O(\log(R))$ approximation ratio in undominated strategies: Let $Shifting$ be the original shifting allocation algorithm of [10], and $M(Shifting, \log(R))$ be the Impersonation-Based mechanism based on $Shifting$, with $K = \log(R)$. By showing that $M(Shifting, \log(R))$ is value monotonic and encourages minimal bundle bidding, we have:

Theorem 8. *The $M(Shifting, \log(R))$ mechanism implements an $O(\log(R))$-approximation in undominated strategies.*

A modified k-CA[19] allocation rule for general USM-CA. The Impersonation Based mechanism with the k-CA auction is not value monotonic. Based on it, we define the $IA - k - CA$ allocation rule, which is suitable for our impersonation technique. It has an approximation ratio of $(e \cdot \epsilon \cdot \sqrt{m} \cdot \ln \bar{v}_{max})$ for any fixed $\epsilon > 0$. We then have:

Theorem 9. *The $M(IA{-}k{-}CA, \sqrt{m})$ mechanism implements an $(e{\cdot}\epsilon{\cdot}\sqrt{m}{\cdot}\ln \bar{v}_{max})$-approximation in undominated strategies.*

Acknowledgements

We are grateful to Noam Nisan for many valuable suggestions and advice. We also thank Daniel Lehmann for many helpful suggestions.

References

[1] A. Archer, C. Papadimitriou, K. Talwar, and E. Tardos. An approximate truthful mechanism for combinatorial auctions with single parameter agent. In *Proceedings of SODA*, pages 205–214, 2003.

[2] Aaron Archer and Eva Tardos. Truthful mechanisms for one-parameter agents. In *Proceedings of FOCS*, pages 482–491, 2001.

[3] B. Awerbuch, Y. Azar, and A. Meyerson. Reducing truth-telling online mechanisms to online optimization. In *Proceedings of STOC*, pages 503–510, 2003.

[4] Moshe Babaioff and Liad Blumrosen. Computationally feasible auctions for convex bundles. In *7th. International Workshop on Approximation Algorithms for Combinatorial Optimization Problems (APPROX). LNCS Vol. 3122*, pages 27–38, 2004.

[5] E. H. Clarke. Multipart pricing of public goods. *Public Choice*, pages 17–33, 1971.

[6] T. Groves. Incentives in teams. *Econometrica*, pages 617–631, 1973.

[19] the k-CA is a generalization of the 1-CA algorithm.

[7] Venkatesan Guruswami, Sanjeev Khanna, Rajmohan Rajaraman, Bruce Shepherd, and Mihalis Yannakakis. Near-optimal hardness results and approximation algorithms for edge-disjoint paths and related problems. *Journal of Computer and System Sciences*, 67(3):473–496, 2003.

[8] Magnus M. Halldorsson. Approximation of weighted independent set and hereditary subset problems. *Journal of Graph Algorithms and Applications*, 4(1):1–16, 2000.

[9] Matthew O. Jackson. Implementation in undominated strategies: A look at bounded mechanisms. *Review of Economic Studies*, 59:757–775, 1992.

[10] S. Khanna, S. Muthukrishnan, and M. Paterson. On approximating rectangle tiling and packing. In *Symposium on Discrete Algorithms*, pages 384–393, 1998.

[11] Ron Lavi and Noam Nisan. Online ascending auctions for gradually expiring items. In *SODA'05*, 2005.

[12] Daniel Lehmann, Liadan Ita O'Callaghan, and Yoav Shoham. Truth revelation in approximately efficient combinatorial auctions. *Journal of the ACM*, 49(5):1–26, 2002.

[13] A. Mu'alem and N. Nisan. Truthful approximation mechanisms for restricted combinatorial auctions. In AAAI (poster) 2002.

[14] N. Nisan and I. Segal. The communication requirements of efficient allocations and supporting lindahl prices.

[15] Noam Nisan and Amir Ronen. Algorithmic mechanism design. *Games and Economic Behavior*, 35:166–196, 2001.

[16] David Parkes. Iterative combinatorial auctions, 2005.

[17] Tuomas Sandholm. Algorithm for optimal winner determination in combinatorial auctions. *Artificial Intelligence*, 135(1-2):1–54, 2002.

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

A Revelation Mechanisms

Some of our mechanisms assume a specific structure on the strategy space $\mathcal{S}$. In a *direct revelation mechanism* the strategy space $\mathcal{S}$ is equal to the type space $\mathcal{T}$, which means that each agent reports a value and a set of outcomes from its predefined family of possible sets. A direct revelation mechanism is *incentive-compatible in dominant strategies* if for any agents types $\bar{t} \in \mathcal{T}$ each agent has a dominant strategy to bid his true type (strategy $s_i \in \mathcal{S}_i$ is a *dominant strategy* for agent i with type $\bar{t}_i$, if it dominates any other strategy $s'_i \in \mathcal{S}_i$ of agent i.)

We define the *value of an outcome ω with respect to types $\bar{t} \in \mathcal{T}$* to be $v(\omega, \bar{t}) = \sum_{i \in N} v_i(\omega, \bar{t}_i) = \sum_{i:\omega \in \bar{A}_i} \bar{v}_i$. An outcome $\omega^*(\bar{t}) \in \Omega$ is *efficient with respect to* $\bar{t}$, if it has maximal value, that is $v(\omega^*(\bar{t}), \bar{t}) \geq v(\omega, \bar{t})$ for any $\omega \in \Omega$. An outcome $\omega \in \Omega$ is *c-approximation* with respect to $\bar{t}$ if $v(\omega^*(\bar{t}), \bar{t}) \leq cv(\omega, \bar{t})$, where $\omega^*(\bar{t})$ is an efficient outcome with respect to $\bar{t}$. An incentive-compatible direct revelation mechanism achieves a *c-approximation* if for any types $\bar{t} \in \mathcal{T}$, it outputs a c-approximation outcome with respect to $\bar{t}$ in polynomial time.

For single-parameter agents we suggest the following generalization of the direct revelation mechanism concept. In a *direct value revelation mechanism* (also referred as mechanism) the strategy space $\mathcal{S}$ is equal to the product of the value space $\Re^n_{++}$ and some *outcome bid space* $\mathcal{B} = \mathcal{B}_1 \times \cdots \times \mathcal{B}_n$, where $\mathcal{B}_i$ is the outcome bid space of agent i. A bid of agent i is constructed from a reported value v_i and an outcome bid $B_i \in \mathcal{B}_i$ (the outcome bid is independent of the agent's value, which means that the strategy space is separable). We denote the vector of values reported by all agents by v and the vector of outcome bids by B. Assume that the agents report values v and outcome bids B. The mechanism picks an outcome $G(v, B) = \omega \in \Omega$ and charges agent i with $P_i(v, B)$ monetary units. If an outcome $\omega \in \bar{A}_i$ is chosen then we say that i is *satisfied*.

Definition 10. *An allocation rule G is* value monotonic *if for all $i \in N, B \in \mathcal{B}, v \in \Re^n_{++}, A_i \in \mathcal{A}_i$, if $G(v, B) \in A_i$ and $v'_i > v_i$ then $G(v', B) \in A_i$, where $v' = (v'_i, v_{-i})$.*

We say that a mechanism M is value monotonic if the allocation rule G is value monotonic.

Observation 2. *If an allocation rule G is value monotonic then for all $i \in N, B \in \mathcal{B}, v_{-i} \in \Re^{n-1}_{++}, A_i \in \mathcal{A}_i$, there exists a critical value $c_i(A_i|v_{-i}, B) \geq 0$, such that*

- *if $v_i > c_i(A_i|v_{-i}, B)$ then $G(v, B) \in A_i$*
- *if $v_i < c_i(A_i|v_{-i}, B)$ then $G(v, B) \notin A_i$*

$c_i(A_i|v_{-i}, B)$ is the infimum of all values that will cause the mechanism to pick an outcome in A_i when the agents bids B and the other agents report the values v_{-i}.

We want the agents to participate in our mechanisms voluntarily (ex post individually rational mechanisms). In a truth telling mechanism unsatisfied agents should pay zero. We concentrate on mechanisms that are *satisfy guessing*, such mechanisms guess the satisfying set of an agent from its outcome bid. For each agent i, a satisfy guessing mechanism defines a function $S_i : \mathcal{B}_i \to \mathcal{A}_i$, which is given the outcome bid of agent i and calculates the *alleged satisfying set* $S_i(B_i) \in \mathcal{A}_i$ of the agent. We require that S_i be **on** $\mathcal{A}_i$, that is for any $\bar{A}_i \in \mathcal{A}_i$ there exists an outcome bid $\bar{B}_i \in \mathcal{B}_i$ such that $S_i(\bar{B}_i) = \bar{A}_i$. We denote $S(B) = \{S_i(B_i)\}_{i \in N}$.

For an outcome ω, let $Sat(\omega, A) = \{i \in N | \omega \in A_i\}$, $Sat(\omega, A)$ is the set of agents that are satisfied when agent i has a satisfying set A_i (the set of agents that are actually satisfied by the outcome ω is $Sat(\omega, \bar{A})$). A satisfying guessing mechanism partition the agents to *winners* and *losers*. The set of winners is the set of agents that M assumes that are satisfied by the outcome, $W(v,B) = Sat(\omega, S(B)) = \{i \in N | G(v,B) \in S_i(B_i)\}$, the rest of the agents are losers. Note that a winner might not be satisfied by the outcome, this is the case if the guess of the mechanism regarding the chosen outcome is wrong. A satisfying guessing mechanism is *normalized* if each loser pays zero.

In a mechanism with payments by critical values, winners pay their critical value for winning and losers pay zero.

Definition 11. *A satisfying guessing mechanism M has* payments by critical values *if for all $i \in N, B \in \mathcal{B}, v \in \Re^n_{++}$, if $i \in W(v,B)$ then $P_i(v,B) = c_i(S_i(B_i)|v_{-i}, B)$, otherwise $P_i(v,B) = 0$.*

Note that each agent's payment is non negative, so a winner that is not satisfied has non positive utility.

The next Lemma shows that a mechanism is individually rational if it is value monotonic and the payments are by critical values.

Lemma 2. *Assume that the satisfying guessing mechanism M is value monotonic, the payments are by critical values. Then for all $i \in N, B_{-i} \in \mathcal{B}_{-i}, v_{-i} \in \Re^{n-1}_{++}, \bar{A}_i \in \mathcal{A}_i$, if i bids his true value $\bar{v}_i$ and B_i such that $S_i(B_i) \subseteq \bar{A}_i$, then his utility is non negative.*

Proof. If i loses ($i \notin W(v,B)$) he pays zero and his utility is non negative. If i wins ($i \in W(v,B)$) then by monotonicity, $G(v,B) \in S_i(B_i)$. Since $S_i(B_i) \subseteq \bar{A}_i$, i wins and is satisfied. Since the payments are by critical values agent i wins thus $\bar{v}_i \geq c_i(S_i(B_i)|v_{-i}, B)$, and i pays $P_i(v,B) = c_i(S_i(B_i)|v_{-i}, B)$. Thus his utility is $\bar{v}_i - c_i(S_i(B_i)|v_{-i}, B) \geq 0$. □

Definition 12. *A satisfying guessing mechanism M* ensures losers unsatisfaction *if for all $i \in N, B \in \mathcal{B}, v \in \Re^n_{++}, \bar{A}_i \in \mathcal{A}_i$, if $i \notin W(v,B)$ then $G(v,B) \notin \bar{A}_i$.*

A mechanism that ensures losers unsatisfaction chooses an outcome that does not satisfy any loser, for any satisfying set of each loser (In combinatorial auctions this is trivially done by assigning a loser with the empty bundle, and this can be done for any allocation to the other agents).

Lemma 3. *Assume that the satisfying guessing mechanism M is value monotonic, the payments are by critical values and it ensures losers unsatisfaction. Then for all $i \in N, B_{-i} \in \mathcal{B}_{-i}, v_{-i} \in \Re^{n-1}_{++}, \bar{A}_i \in \mathcal{A}_i$, if i bids v_i and B_i such that $S_i(B_i) \subseteq \bar{A}_i$, then his utility will not decrease if he bids $\bar{v}_i$ and B_i.*

Proof. First assume that i wins ($i \in W(v,B)$), that is $G(v,B) \in S_i(B_i)$. Since $S_i(B_i) \subseteq \bar{A}_i$, i is satisfied. Since M is value monotonic and the payments are by critical values $v_i \geq c_i(S_i(B_i)|v_{-i}, B)$. Agent i pays $P_i(v,B) = c_i(S_i(B_i)|v_{-i}, B)$ and has utility $\bar{v}_i - c_i(S_i(B_i)|v_{-i}, B)$. Assume that i changes his reported value to $\bar{v}_i$. If $\bar{v}_i \geq c_i(S_i(B_i)|v_{-i}, B)$, i will still win, be satisfied and pay the same, so his utility will not change. If $\bar{v}_i < c_i(S_i(B_i)|v_{-i}, B)$, i utility will improve from negative to zero.

Next assume that i loses ($i \notin W(v,B)$) by bidding v_i. Since M ensures losers unsatisfaction and he pays zero, his utility is zero. Assume that i changes his reported value to $\bar{v}_i$. If $\bar{v}_i < c_i(S_i(B_i)|v_{-i}, B)$, i utility will stay zero. If $\bar{v}_i \geq c_i(S_i(B_i)|v_{-i}, B)$, i will win, be satisfied, pay $c_i(S_i(B_i)|v_{-i}, B)$ and his utility will be non negative. □

The KMM CA Greedy Mechanism:
Input: For each player i, a value v_i. The set of desired bundles $\{s_i^k\}_{k=1}^{k_i}$ is known.
Allocation:

1. Sort the bids by descending order of the value $v_i/\sqrt{s_i^k}$.
2. While the list is not empty, pick the first bid on the list (of agent i with bundle s_i^k) agent i wins. Remove all other bids of i from the list, as well as any other bid that intersects s_i^k.

Payments:
Each winner pays his critical value for winning, losers pay 0.

Figure 6: The KMM CA Greedy Mechanism. It is value monotonic, thus it is incentive-compatible for "Known" single-parameter multi-minded agents. It is not truthful for "Unknown" agents.

B Dominant Strategies Mechanisms

B.1 Known Single Parameter Domains

B.1.1 Known Single Parameter Combinatorial Auctions

The results in this section hold even if we do not assume that $\bar{v}_i \geq 1$ for any agent i. It is sufficient to assume that $\bar{v}_i > 0$ for any i.

Theorem 10. *The KMM CA Greedy Mechanism (see Figure 6) is truthful for multi-minded bidders and achieves a $(\sqrt{m}+1)$-approximation in polynomial time.*

Proof. The mechanism is clearly normalized, thus by showing that it is value monotonic, we conclude by Theorem 1 that it is truthful. It is value monotonic since if agent i wins (and is satisfied) with some bundle s_i^k and increases his value, all his bids only improve, and the allocation before s_i^k is considered does not change unless one of agent i's bundles win. If this is not the case, s_i^k is considered at a prior stage to its original stage, thus must win.

Lehmann et al. [12] proved that this algorithm is a $\sqrt{m}$-approximation for USM agents (and in particular for KSM agents). Let W be the set of winners in the allocation that the mechanism outputs, and let OPT be the set of winners in some efficient allocation. For the KMM agents case, we can partition the agents in OPT to two types, agents that are also in W (note that they might win with different bundles in the two allocations), and agents that are not in W. Clearly the value of the agents in W bounds the value of the agents of the second type (all bundles winning in the mechanism's allocation are disjoint), and is a $\sqrt{m}$-approximation to the value of the agents of the first type (since all bundles of the same agents can be regarded as bundles of different agents, and they all still lose). Thus W is a $\sqrt{m}+1$-approximation to OPT. □

The EDP Greedy Mechanism uses the EDP Greedy allocation algorithm of Figure 1 and critical value payments to create a mechanism for the Edge Disjoint Paths problem for "known" single-parameter agents.

Theorem 11. *The EDP Greedy Mechanism is truthful for KSP agents in the EDP model, it achieves a $(\sqrt{m}+1)$-approximation in polynomial time.*

Proof. Assume that we run the The KMM CA Greedy Mechanism on all paths of each agent as his desired bundles. By Theorem 10 this creates a truthful mechanism that achieves a $(\sqrt{m}+1)$-

approximation. It is easy to verify that the EDP Greedy allocation always produces the same allocation and runs in polynomial time. □

Somewhat surprisingly, it turns out that the EDP Greedy Mechanism is not truthful for the unknown case:

Claim 1. *In the USP EDP problem, where each player has to report both his value and his source-target node pair, the EDP Greedy Mechanism is not truthful.*

Proof. Consider the undirected cycle graph on 5 nodes, $n_1, n_2, \ldots, n_5$. Suppose player 1 has type $(\$10, (n_1, n_2))$ (has a value of \$10 for paths from n_1 to n_2), and player 2 has type $(\$5, (n_1, n_5))$, and that they both bid truthfully. Player 3 has type $(\$100, (n_1, n_2))$. If he bids truthfully he wins (receive the edge (n_1, n_2)), and pays 10, as this is his critical value to win. However, if he bids $(\$100, (n_5, n_2))$, then he still wins and receives the edges (n_5, n_1) and (n_1, n_2) (so he is satisfied). But, his critical value for winning will now be zero, as he would have been able to win and receive the edges $(n_2, n_3), (n_3, n_4), (n_4, n_5)$ even if he would have declared any value greater than 0 (in this case he would still win, but, he would not be satisfied according to his true type). Thus player 3 can increase his utility by declaring a false type. □

B.1.2 The Japanese Wrapper Mechanism for KSP domains

In this section we present a generic way to create a dominant strategy IC mechanism for the KSP domain from any allocation algorithm for single parameter agents.

Let $\bar{v}_{max} = \max_{i \in N} \bar{v}_i$, $\bar{v}_{max}$ is the **observed** maximal value of any agent (it is not an a-priori bound).

An algorithm is an *allocation algorithm for single parameter agents*, if it gets as input a set of agents N and a set of outcomes Ω. For any agent i it also get a satisfying set A_i and a value v_i. The allocation algorithm outputs an outcome $\omega \in \Omega$. An allocation algorithm for single parameter agents is a *c-approximation algorithm* if for any $N, \Omega, \bar{A}$ and values $\bar{v}$ it outputs an outcome that is c-approximation to the efficient outcome with respect to $\bar{v}$ in polynomial time. Assume that we are given an allocation algorithm ALG for single parameter agents. Since the satisfying sets of the agents are known, the input for the algorithm will be the set of agents N, the set of outcomes Ω and the satisfying set of the agents $\bar{A}$. We denote by $ALG(v)$ the outcome of the algorithm ALG on values v.

Algorithm ALG *ensures losers unsatisfaction* if it always outputs an outcome $\omega \in \Omega$ such that $\omega \notin \bar{A}_i$ for any agent i with $v_i = 0$.

To simplify notation we use W_j to denote the set agents satisfied by the allocation W_j, that is the set $Sat(W_j, \bar{A})$.

The Japanese Wrapper Mechanism for KSP agents is described in Figure 2. In Line 1, j denote the iteration and initialized to 0, W_j is the set of provisional winners and L_j is the set of losers (non active agents). Next (line 2) each agent reported value at iteration 0, v_i^0 is initialized to 1. The iterations (line 3) run as long as there are active agents (agents with reported value greater than 0) that are not winners. Line 4 runs the algorithm ALG on the current vector of reported values v^j to get an allocation. If this allocation is better than the previous one, the current allocation is updated (line 5). Next, each loser can either increase his reported value by a factor of e (e denotes the Euler number $e = 2.718...$), or retire (become inactive). Finally, line 7 updates the set of losers (inactive agents) and the iteration counter. The total number of iterations is denotes by J at line

8. At the end, the set of active agents equals the set of winners, and each winner pays his final reported value v_i^J.

Observation 3. *If the allocation algorithm ALG runs in polynomial time then the Japanese Wrapper Mechanism runs in polynomial time.*

Lemma 4. *Assume that the Japanese Wrapper Mechanism uses an algorithm ALG that ensures losers unsatisfaction. It is a dominant strategy for any agent i to increase his reported value as long as the increased value is lower then his value $\bar{v}_i$. That is, at each stage j, if $\omega_j \notin \bar{A}_i$ and $ev_i^j < \bar{v}_i$, i reports $v_i^{j+1} = ev_i^j$, otherwise he reports $v_i^{j+1} = 0$.*

Proof. If agent i bids any value larger than $\bar{v}_i$ his utility will be non-positive (zero if he loses, negative if he wins), this can be improved by i retiring at a value not larger than $\bar{v}_i$.

If agent i retires at a value smaller than $\bar{v}_i/e$ his utility is zero. If he continue to bid and retires at the largest possible value smaller than $\bar{v}_i$, his utility is non negative. We conclude that a dominant strategy for any agent i to increase his reported value as long as the increased value is lower then his value $\bar{v}_i$. □

Lemma 5. *For any allocation algorithm ALG used by the Japanese Wrapper Mechanism, the mechanism is individually rational and incentive compatible in dominant strategies.*

Proof. Lemma 4 shows that it is a dominant strategy for any agent i to increase his reported value as long as the increased value is lower then his value $\bar{v}_i$. Clearly if each agent i retires at a value not larger than $\bar{v}_i$ his utility is non negative, so the mechanism is individually rational. □

Each losing agent retires at a value that it at least $\bar{v}_i/e$ (for each loser i, $v_i^J \geq \bar{v}_i/e$). Each agent can increase his value at most $ln(\bar{v}_{max})$ times. Since there are n agents and at each stage at least one increases his value, $J \leq n \ln \bar{v}_{max}$. Note that basic calculus shows that increasing the value by the factor of e is optimal over all constants.

Lemma 6. *If ALG is a c-approximation allocation algorithm and the Japanese Wrapper Mechanism runs J stages, then it achieves an $(ecJ + 1)$-approximation.*

Proof. Let us denote the value of an efficient outcome for the set of agents T with values v by $\omega^*(T, v)$. To simplify notation we use W_J to denote the set of agents satisfied by the allocation W_J. Let $R_j = L_j \setminus L_{j-1}$ denote the set of agents that retired at iteration j. Note that $L_J = \cup_{j=1}^J R_j$ and that this is a union of disjoint sets. Let r_i denote the last non zero reported value of agent i (this is the value that losing agent retired at, and the value that a winner pays), and let r denote the vector of these values.

$$\omega^*(N, \bar{v}) = \omega^*(W_J \cup L_J, \bar{v}) \leq \omega^*(W_J, \bar{v}) + \omega^*(L_J, \bar{v}) \leq \omega^*(W_J, \bar{v}) + e\omega^*(L_J, r)$$

$$\omega^*(L_J, r) \leq \sum_{j=1}^{J} \omega^*(R_j, r) = \sum_{j=1}^{J} \omega^*(R_j, v^j) \leq \sum_{j=1}^{J} c\omega^*(W_j, v^j) \leq c \sum_{j=1}^{J} \omega^*(W_J, v^J) =$$

$$cJ\omega^*(W_J, v^J) \leq cJ\omega^*(W_J, \bar{v})$$

So we conclude:

$$\omega^*(N, \bar{v}) \leq \omega^*(W_J, \bar{v}) + e\omega^*(L_J, r) \leq \omega^*(W_J, \bar{v}) + ecJ\omega^*(W_J, \bar{v}) = \omega^*(W_J, \bar{v})(ecJ + 1)$$

□

Note that if the algorithm ALG never outputs an outcome with a value that is worse than the highest agent's value at each stage, then the mechanism is also an (en)-approximation.

Next, we present a condition on the algorithm used by the KSP Japanese Wrapper Mechanism that is sufficient to ensure that the maximal number of iteration J is low, thus ensuring good approximation.

The KSP Japanese Wrapper Mechanism with an algorithm ALG has the *d-wise property*, if for any strategies of the agents, any stage j and any agent $i \in L_j$, there exists a set $X \subseteq W_j$, $|X| \leq d-1$ such that at no stage prior to j, ALG outputs an outcome that satisfies $X \cup \{i\}$.

Lemma 7. *If the KSP Japanese Wrapper Mechanism with an algorithm ALG has the d-wise property then the mechanism stops after at most $(d \ln \bar{v}_{max} + 1)$ stages.*

Proof. We look at a loser i that retired at the last round (round J). Since KSP Japanese Wrapper Mechanism with an algorithm ALG has the d-wise property, there exists a set $X \subseteq W_J$ of at most $d-1$ agents that such that at least one of the agents in $X \cup \{i\}$ loses at each stage. Each of the agents in $X \cup \{i\}$ can be a loser and multiply its value at most $\ln \bar{v}_{max}$ times, so all the agents in $X \cup \{i\}$ can multiply there value at different stages at most $d \ln \bar{v}_{max}$ consecutive times. We conclude that $J - 1 \leq d \ln \bar{v}_{max}$. □

We now present conditions on the outcome space and the algorithm that are sufficient to ensure that KSP Japanese Wrapper Mechanism with an algorithm ALG has the d-wise property.

A set of agents $T \subseteq N$ *get along* if there exist an outcome $\omega \in \Omega$ such that $T \subseteq Sat(\omega, \bar{A})$ (all agent in T can be satisfied together by some outcome ω). An outcome $\omega \in \Omega$ *pareto dominates* outcome $\omega' \in \Omega$ if $Sat(\omega, \bar{A}) \subset Sat(\omega', \bar{A})$ (the set of satisfied agents by the outcome ω' is a proper subset of the set of agents that are satisfied by ω). An outcome $\omega \in \Omega$ is *pareto efficient* if there is no outcome that pareto dominates ω. An outcomes space Ω is a *d-wise space* if for any set T that get along such that $|T| \geq d-1$, and any $i \notin T$: $T \cup \{i\}$ get along if and only if $X \cup \{i\}$ get along for any $X \subseteq T$ such that $|X| = d-1$.

A *pareto efficient algorithm* always outputs an outcome that is pareto efficient.

Lemma 8. *If the Japanese Wrapper Mechanism runs on a d-wise space using a pareto efficient algorithm ALG then the KSP Japanese Wrapper Mechanism with an algorithm ALG has the d-wise property.*

Proof. We look at the execution of the mechanism for some strategies of the agents. For any stage j and any agent $i \in L_j$, we need to show a set $X \subseteq W_j$ of size at most $d-1$ that does not get along with i.

Since the algorithm is pareto efficient, W_j does not get along with i. There are two cases. Either $|W_j| \geq d-1$ and there exists a set $X \subset W_j$, $|X| = d-1$ that does not get along with i (since the space is d-wise), or $|W_j| < d-1$ (in that case $X = W_j$). In both cases we have shown that there is a set $X \subseteq W_j$ of size at most $d-1$ that does not get along with i, thus at no stage prior to j, ALG outputs an outcome that satisfies $X \cup \{i\}$. □

Next, we present an example for d-wise spaces.

Lemma 9. *The single-minded CA domain is a 2-wise space. Combinatorial auctions with n agents that are l-minded for any $l > 1$, is not a t-wise space for any $t < n$.*

Proof. The single-minded CA domain is a 2-wise space, since if we take any set of single-minded agents with disjoint desired bundles and an agent with desired bundle that is disjoint from each of them, then he is disjoint from all of them.

Next we show that if agents are 2-minded, then it is possible that any $n-1$ agents get along, but all n agent do not. Assume that there are $n-1$ goods, ordered in a cycle. Agent $i \in \{1, n-1\}$ is double minded with bundles $\{i\}$ and $\{(i+1) \ mod \ (n-1)\}$. Agent n wants the bundle $\{1\}$. Clearly, the $n-1$ goods cannot satisfy all these n agents. Any set $T \subset N$ (the size of T is smaller then n) can be satisfied. If agent $n \notin T$, assign each agent i the bundle $\{i\}$. If agent $n \in T$, assign agent n the bundle $\{1\}$, and starting from him assign the goods greedily in both directions. □

In some d-wise spaces it is computationally feasible to take an outcome and calculate a pareto efficient outcome that pareto dominates it. In such spaces any algorithm ALG that achieves a c-approximation can be efficiently converted to an algorithm that is pareto efficient and achieves a c-approximation (since the pareto dominating outcome has a higher value). For combinatorial auctions it is computationally feasible to find a pareto efficient outcome that dominates another outcome (just go over all losers one by one and check if the current candidate can be added to the allocation).

From Lemma 6 and Lemma 7:

Corollary 1. *If the KSP Japanese Wrapper Mechanism with an algorithm ALG has the d-wise property, and ALG is a c-approximation allocation algorithm, then it achieves an $(ec(d \ln \bar{v}_{max} + 1) + 1)$-approximation.*

Let us now concentrate on combinatorial auctions for single minded agents. By Lemma 9 this space is a 2-wise space, and we can turn any algorithm into pareto efficient algorithm in computationally efficient way. We get the following result.

Theorem 12. *If the Japanese Wrapper Mechanism runs on a known single-minded CA domain using a pareto efficient c-approximation polynomial algorithm ALG, then it achieves an $(ec(2\ln(\bar{v}_{max}) + 1) + 1)$-approximation in dominant strategies, and the mechanism runs in polynomial time.*

There exists a $\sqrt{m}$-approximation pareto efficient algorithm for combinatorial auctions ([12]). For such an algorithm, the Japanese Wrapper Mechanism achieves an $(e\sqrt{m}(2\ln(\bar{v}_{max}) + 1) + 1)$-approximation in dominant strategies, by an ascending auction.

Next, we explain how to modify any allocation algorithm to ensure that the Japanese Wrapper Mechanism approximation is never worse then $e \min(m, n)$. Given an allocation algorithm $\widehat{ALG}$ we create an "improved" allocation algorithm ALG as follows. If the value of the highest loser is higher then the value of the algorithm output, then the algorithm output a pareto efficient allocation that includes the highest bidder and in which any agent with zero value receives the empty bundle. Note that if $\widehat{ALG}$ is a c-approximation algorithm then ALG is also a c-approximation algorithm. So we conclude the following.

Corollary 2. *If the Japanese Wrapper Mechanism runs a combinatorial auction for single minded agents using algorithm ALG that is an "improved" c-approximation allocation algorithm (always output an allocation that is at least as good as the highest agent's value), then it achieves a $min(em, en, ec(2\ln(\bar{v}_{max}) + 1) + 1)$-approximation in dominant strategies.*

Up to a constant factor, our analysis is tight.

Proposition 4. *There exists an example with $2m$ agents, $\bar{v}_{max} = m + 1$ and an "improved" e-approximation algorithm for the KSM CA domain, for which the Japanese Wrapper Mechanism is an $\Omega(m)$-approximation.*

Proof. For $i = 1, \ldots, m$, agent i wants the bundle $\{i\}$ with a value of $\bar{v}_{max}$. For $k = 1, \ldots, m$, agent $i = m + k$ wants the bundle $\{1, 2, \ldots, k\}$ with a value of e^{k+1}.

In the efficient allocation all the first m agents win and the allocation has a value of $m\bar{v}_{max}$. Next, we turn to analyze the execution of the mechanism and show that there exists an e-approximation algorithm that achieves a total value of $\bar{v}_{max}$ and no more.

At the first iteration ($j = 0$), all the first m agents win. If the algorithm pick the efficient allocation at the following m iterations, then at iteration $j \in \{1, \ldots, m\}$, agent $m + j$ wins with all agents $j + 1, \ldots, m$.

After these iterations, agent $i \in \{1, \ldots, m\}$ has a value of $\bar{v}_{max}/e^i$. All agents $i \in \{m, \ldots, 2m-1\}$ retired, and agent $2m$ reported value is $\bar{v}_{max}$. At iteration $m + 1$, agent $i \in \{1, \ldots, m\}$ has a value of $\bar{v}_{max}/e^{i-1}$. The sum of values of all these agents is less than $e\bar{v}_{max}$ (e times the value of agent $2m$), so the algorithm can still pick agent $2m$ as the sole winner. Now, agent 2 retires, all the rest multiply their value by e, but the same argument shows that the algorithm can still pick agent $2m$. At each following stage the next agent retires, till none remains. □

B.1.3 Mechanism based on a Bitonic Unweighted Allocation Rule

In this section we show how a bitonic allocation rule for unweighted KSP can be used to create a dominant strategies incentive-compatibility mechanism for weighted KSP. Clearly, creating bitonic unweighted algorithms is easier than creating monotonic algorithms, so this technique is useful in many cases.

An *unweighted* allocation algorithm for single parameter agents is an allocation rule for single parameter agents that is defined when the agents values are either zero or one. Note that any allocation algorithm is also an unweighted allocation algorithm.

Bitonic allocation rules were defined by Mu'alem and Nisan [13]. A monotonic allocation rule is *bitonic* if the value of the outcome does not increase when a losing agent increase its value and remains a loser, and does not decrease when a winning agent increase its value.

Assume that in an unweighted allocation algorithm, the set of agents with reported value of 0 are losers and are not satisfied. This means that only agents with a value of 1 can be winners, thus bitonicity only means the following. If a loser with value 0 changes his value to 1 and remains a loser then the value of the allocation does not increase.

The mechanism presented in Figure 7 takes any allocation rule that is bitonic for unweighted single-parameter agents and creates a mechanism that is incentive compatible in dominant strategies for KSP agents.

Lemma 10. *If the unweighted allocation rule BR is a bitonic rule then the resulting mechanism based on BR is individually rational and incentive compatible in dominant strategies for the KSP model.*

Proof. Since losers pay 0, by Theorem 1 we only need to show that the mechanism is value monotonic. Assume that agent i wins, so he appears with value 1 in class C^*. If i increases his value, the only change in the input is that its value is changed from 0 to 1 in some classes after class C^*. There is no change in class C^* and all prior classes. In all other classes, either there is no

Mechanism based on a Bitonic Unweighted Allocation Rule:

Allocation:

Step 1: Each agent i report a value v_i.

Step 2: Divide the given input to $\ln(\bar{v}_{max})$ classes according to their value. Agent i that bids v_i appears with value 1 in any class C such that $v_i \geq e^{C-1}$, and with value 0 in all other classes.

Step 3: Perform a bitonic allocation rule BR per each class C. Denote the number of winners in class C by $n(C)$.

Step 4: Output the class C^* for which $e^{C-1}n(C)$ is maximal.

Payments:

Each winner pays his critical value for winning, losers pay 0.

Figure 7: A mechanism that uses a bitonic allocation rule BR that is a c-approximation for the KSP unweighted case. This mechanism is incentive compatible in the KSP model and achieves an $(ec \ln \bar{v}_{max})$-approximation in dominant strategies.

change, or i remains a loser and the value of the class decreases (since the rule is bitonic), or i wins in that class and the value increases. We conclude that either class C^* remains the winning class or an higher class with i winning in it become the winning class. In any case i remains a winner. □

Theorem 13. *If the unweighted allocation rule BR is a bitonic rule that achieves a c-approximation for unweighted single-parameter agents then the resulting mechanism based on BR is individually rational and incentive compatible in dominant strategies for the KSP model, The mechanism achieves an $(ec \ln \bar{v}_{max})$-approximation in dominant strategies.*

Proof. By Lemma 10 the resulting mechanism is individually rational and incentive compatible in dominant strategies.

Next we prove the approximation ratio. Let $\omega^*(N, \bar{v})$ denotes both the value of the efficient outcome with respect to the set of agents N and values $\bar{v}$, as well as the set of satisfied agents in that outcome. Let $A(C) = \{i \in N | v_i \in [e^{C-1}, e^C)\}$ denote the set of agents for which the last class they appear in is C.

Let $W(C) = \omega^*(N, \bar{v}) \cap A(C)$ be the set of agents in the efficient allocation and with value in $[e^{C-1}, e^C)$. Let C_m be the class with the maximal value of $\omega^*(W(C), \bar{v})$. It holds that $\omega^*(W(C_m), \bar{v}) \geq \frac{\omega^*(N,\bar{v})}{\ln \bar{v}_{max}}$.

Let $\hat{v}_i$ be the value of agent i rounded down to an integral power of e, and let $\hat{v}$ denotes the vector of such values. Since agents bid at least an e fraction of their true values, $\omega^*(W(C_m), \bar{v}) \leq e\ \omega^*(W(C_m), \hat{v})$. Since the unweighted rule is a c-approximation allocation rule, it holds that $\omega^*(W(C_m), \hat{v}) \leq c\ e^{C_m - 1} n(C_m) \leq c\ e^{C^*-1} n(C^*)$, where the last inequality is since the mechanism chooses the class C^* that maximizes $e^{C-1}n(C)$.

We conclude that

$$\omega^*(N, \bar{v}) \leq (\ln \bar{v}_{max})\omega^*(W(C_m), \bar{v}) \leq (\ln \bar{v}_{max})e\ \omega^*(W(C_m), \hat{v}) \leq (\ln \bar{v}_{max})e\ c\ e^{C^*-1}n(C^*)$$

which means that the value of at least $e^{C^*-1}n(C^*)$ achieved by the mechanism is an $(ec \ln \bar{v}_{max})$-approximation to the efficient allocation $\omega^*(N, \bar{v})$. □

B.2 Unknown Single Parameter (USP) Domains

In the USP model, all information is public except the value of each agent and his satisfying set, which are private information. In a direct revelation USP mechanism, each agent's outcome bid space is the family of satisfying sets of that agent ($\mathcal{B}_i = \mathcal{A}_i$). We denote the outcome bid (reported satisfying set) of agent i by $\tilde{A}_i \in \mathcal{A}_i$. The alleged satisfying set of an agent is its reported satisfying set, that is $S_i(B_i) = \tilde{A}_i$. This means that a winner might not be satisfied. This also means that a loser might be satisfied and pay zero (this will be prevented by requiring that the mechanism ensures losers unsatisfaction).

Definition 13. *A monotonic direct revelation mechanism M* ensures minimal payments *if for all $i \in N, A \in \mathcal{A}, v \in \Re^n_{++}, \bar{A}_i \in \mathcal{A}_i$, if i bids $v_i > c_i(A_i|v_{-i}, A)$ and A_i such that $G(v, A) \in A_i \cap \bar{A}_i$ (i is satisfied and he wins) then $c_i(\bar{A}_i|v_{-i}, (\bar{A}_i, A_{-i})) \leq c_i(A_i|v_{-i}, A)$.*

Lemma 11. *A monotonic direct revelation mechanism M ensures minimal payments if and only if for all $i \in N, A \in \mathcal{A}, v_{-i} \in \Re^{n-1}_{++}, \bar{A}_i \in \mathcal{A}_i$, if there exists a value v_i for agent i such that $G(v, A) \in A_i \cap \bar{A}_i$ (i is satisfied and he wins) then for any $v'_i > c_i(A_i|v_{-i}, A)$ (i wins with a bid v'_i and A_i, but he does not bid his exact critical value), agent i wins and is satisfied if he bids v'_i and $\bar{A}_i$.*

Proof. Since M is value monotonic, the critical values are well defined.

Case if: Assume that for all $i \in N, A \in \mathcal{A}, v_{-i} \in \Re^{n-1}_{++}, \bar{A}_i \in \mathcal{A}_i$, if there exists a value v_i for agent i such that $G(v, A) \in A_i \cap \bar{A}_i$ then for any $v'_i > c_i(A_i|v_{-i}, A)$, agent i wins and is satisfied if he bids v'_i and $\bar{A}_i$. Assume in contradiction that $c_i(\bar{A}_i|v_{-i}, (\bar{A}_i, A_{-i})) > c_i(A_i|v_{-i}, A)$. If i bids v'_i such that $c_i(\bar{A}_i|v_{-i}, (\bar{A}_i, A_{-i})) > v'_i > c_i(A_i|v_{-i}, A)$, then he wins with the bid v'_i and A_i, but loses with the bid v'_i and $\bar{A}_i$, which is a contradiction. This shows that the M ensures minimal payments.

Case only if: Assume that M ensures minimal payments, this means that for all $i \in N, A \in \mathcal{A}, v \in \Re^n_{++}, \bar{A}_i \in \mathcal{A}_i$, if i bids $v_i > c_i(A_i|v_{-i}, A)$ and A_i such that $G(v, A) \in A_i \cap \bar{A}_i$ (i is satisfied and he wins) then $c_i(\bar{A}_i|v_{-i}, (\bar{A}_i, A_{-i})) \leq c_i(A_i|v_{-i}, A)$.

If i bids v_i and A_i such that $G(v, A) \in A_i \cap \bar{A}_i$ this means that there exists a value which causes i to win and be satisfied. For and $v'_i > c_i(A_i|v_{-i}, A)$ it holds that $v'_i > c_i(\bar{A}_i|v_{-i}, (\bar{A}_i, A_{-i}))$, and since the mechanism is value monotonic, this means that if i bids v'_i and $\bar{A}_i$, he wins and is satisfied. ☐

If mechanism M ensures minimal payments this means that in any case that the agent is satisfied by untruthfully reporting his satisfying set, he will remain satisfied if he reports his satisfying set truthfully (and pay less).

Theorem 14. *Assume that in the USP model the value of each agent is at least* 1*. A normalized direct revelation mechanism M for USP model is incentive-compatible in dominant strategies and individually rational if and only if M is value monotonic, the payments are by critical values, the mechanism ensures losers unsatisfaction and ensures minimal payments.*

Proof. Case if: By Lemma 2 if agent i bids $\bar{v}_i$ and $\bar{A}_i$ ($S_i(B_i) = \bar{A}_i$) his utility is non negative so the mechanism is ex post individually rational.

To prove incentive-compatibility we need to show that for all $i \in N, A_{-i} \in \mathcal{A}_{-i}, v_{-i} \in \Re^{n-1}_{++}, \bar{A}_i \in \mathcal{A}_i$, if i changes his bid from v_i and A_i to $\bar{v}_i$ and $\bar{A}_i$ his utility does not decrease.

If i has a negative utility by bidding v_i and A_i, by Lemma 2 he can improve his utility to zero by bidding $\bar{v}_i$ and $\bar{A}_i$. Therefore we only need to consider the cases when i has non negative utility.

If i is a loser ($i \notin W(v, B)$) he pays zero, and since M ensures losers unsatisfaction his utility is zero. By Lemma 2 his utility will stay non negative by bidding $\bar{v}_i$ and $\bar{A}_i$.

Next we assume that i wins ($i \in W(v, B)$), that is $\omega = G(v, B) \in A_i$. Since i pays a non negative payment, for him to have positive utility, it must be the case that i is satisfied, that is $\omega \in A_i \cap \bar{A}_i$. In that case he pays $c_i(A_i|v_{-i}, A) \leq v_i$ and has utility $\bar{v}_i - c_i(A_i|v_{-i}, A) \geq 0$. Now assume that i bids $\bar{A}_i$ and v_i instead. Since M ensures minimal payments, $G(v, (\bar{A}_i, A_{-i})) \in \bar{A}_i$ (i is still satisfied if he reports $\bar{A}_i$ instead of A_i). i pays $c_i(\bar{A}_i|v_{-i}, (\bar{A}_i, A_{-i})) \leq c_i(A_i|v_{-i}, A)$, so his utility does not decrease. By Lemma 3 his utility will not decrease if he bids $\bar{v}_i$ instead of v_i and $\bar{A}_i$.

Case only if: Assume that the USP mechanism is ex post individually-rational and incentive-compatible. This implies that it is also individually rational and incentive-compatible for the KSP model, therefore by Theorem 1 it must be value monotonic and the payments must be by critical values.

Next, we show that the mechanism ensures losers unsatisfaction. Assume in contradiction that the mechanism does not ensure losers unsatisfaction. This means that there exists $i \in N, A \in \mathcal{A}, v \in \Re^n_{++}, \bar{A}_i \in \mathcal{A}_i$, such that i loses ($G(v, A) \notin A_i$) but is satisfied ($G(v, A) \in \bar{A}_i$). Assume that in this case $\bar{v}_i = v_i$. In this case i has reported $\bar{v}_i$ and $A_i \neq \bar{A}_i$ (untruthful bidding), pays zero (in a normalized mechanism losers pay zero) and has a utility of $\bar{v}_i > 0$ (since he is satisfied). If i bids $\bar{v}_i$ and $\bar{A}_i$ (truthfully) there are two cases. If i is a loser, he is not satisfied and has utility of zero. If i is a winner, he is satisfied, he pays his critical value which is at least 1 (since its value is at least 1) and has utility smaller than $\bar{v}_i$. We conclude that i has improved his utility by reporting his satisfying set untruthfully, contradicting incentive-compatibility.

Finally we show that the mechanism ensures minimal payments. Assume in contradicting that it does not. Then for some $i \in N, A \in \mathcal{A}, v \in \Re^n_{++}, \bar{A}_i \in \mathcal{A}_i$, i bids $v_i > c_i(A_i|v_{-i}, A)$ and A_i such that $G(v, A) \in A_i \cap \bar{A}_i$ (i is satisfied and he wins) but $c_i(\bar{A}_i|v_{-i}, (\bar{A}_i, A_{-i})) > c_i(A_i|v_{-i}, A)$. Assume that in this case $\bar{v}_i = v_i$. In this case i has reported $\bar{v}_i$ and $A_i \neq \bar{A}_i$ (untruthful bidding), and has a utility of $\bar{v}_i - c_i(A_i|v_{-i}, A) > 0$ (since he is satisfied). If i bids $\bar{v}_i$ and $\bar{A}_i$ (truthfully) there are two cases. If i is a loser, he is not satisfied and has utility of zero. If i is a winner, he is satisfied, he pays $c_i(\bar{A}_i|v_{-i}, (\bar{A}_i, A_{-i})) > c_i(A_i|v_{-i}, A)$, so his utility is smaller than $\bar{v}_i - c_i(A_i|v_{-i}, A)$. We conclude that i has improved his utility by reporting his satisfying set untruthfully, contradicting incentive-compatibility. $\square$

C Undominated Strategies Mechanisms

In this section we concentrate on a specific single-parameter space, the combinatorial auctions space. We present two general frameworks for the creation of mechanisms that achieve good approximations in undominated strategies.

C.1 The Japanese Mechanism for General Combinatorial Auctions

In this section we show that the 1-CA Japanese Mechanism (the Japanese Wrapper of Figure 3 on top of the 1-CA-SUB sub-game of Figure 4) works for general Combinatorial Auctions, and in particular it works for single-parameter CA domains. In the general (not single-parameter)

Combinatorial Auctions domain, each agent i has a valuation for each bundle $b \subseteq G$. Let $v_i(b)$ be the valuation of agent i for the bundle $b \subseteq G$. Assume that for any agent i and bundle b, $v_i(b)$ is either 0 or at least 1. If i has a positive value for bundle b, we say that i *desires* b. Let b_i^{min} be a desired bundle which i values the least. We define the *maximal value difference* δ to be $\delta = \max_{i \in N} v_i(G)/v_i(b_i^{min})$.

Lemma 12. *Assume that we run the 1-CA Japanese Mechanism for general combinatorial auctions. Then, for any agent i, in any undominated strategy:*

1. *In any stage j in which agent i is active (did not retire), $v_i(s_i^j) \geq v_i^j$. Since $v_i^j \geq 1$ it implies that s_i^j contains some desired bundle (a bundle with non zero value).*
2. *If i retires with value v_i^j and bundle s_i^j at stage j, then $v_i^j \geq v_i(s_i^j)/e$.*
3. *If in some stage j it holds that i will retire if losing ($v_i^j \in [v_i(s_i^j)/e, v_i(s_i^j)]$), and i will surely lose if he does not change his bundle ($i \notin W_{j-1}$ and $i \notin W_j(1)$ and $s_i^j \cap (\cup_{i \in W_{j-1}} s_i) \neq \emptyset$ and $s_i^j \cap (\cup_{i \in W_{j(1)}} s_i) \neq \emptyset$), then if there exists some set $s_i^{j+1} \subseteq Free(W_j(2), s^{j+1}) \cap s_i^j$, such that $|s_i^{j+1}| \leq \sqrt{m}$ and $v_i(s_i^{j+1}) \in [v_i^j/e, v_i^j)$ he will report such a bundle (in Line 4).*

and any such strategy is undominated.

Proof. The first claim is derived from the fact that a winner must pay his last reported value. The second claim is derived from the fact that an agent can only gain by not retiring at a lower value (has no risk if he stays active). The third claim is derived from the fact that an agent can only gain by declaring such a bundle.

Next, we show that any strategy that satisfies the above properties is undominated. We look at two such strategies s_i and s'_i of agent i and show that neither dominates the other. This holds since in the first point that they differ (bid different bundles), we can build strategies of the others that will cause one to win with that value and the other to become a loser. □

Observation 4. *The 1-CA Iterative Mechanism for general combinatorial auctions is Ex-post individually rational.*

Proof. The observation is a direct result of the two first properties of any undominated strategies presented in Lemma 12, and the fact that winners pay their final reported value. □

Observation 5. *The 1-CA Iterative Mechanism for general combinatorial auctions has the fast undominance recognition property.*

Proof. Clearly, any agent i can check if his strategy satisfies the properties that any undominated strategies must have by Lemma 12, and if not, modify his strategy to an undominated strategy that dominates the original one, in polynomial time. □

Lemma 13. *The 1-CA Iterative Mechanism for general combinatorial auctions stops after at most* $2 \ln \bar{v}_{max} + 1$ *stages.*

Proof. The allocation is pareto efficient allocation in any stage, so if agent i loses at stage j, he intersects a winner $k \in W_j$. This implies that $s_i^j \cap s_k^j \neq \emptyset$. For any $j' \leq j$ it holds that $s_i^j \subseteq s_i^{j'}$ and

$s_k^j \subseteq s_k^{j'}$, thus $s_i^{j'} \cap s_k^{j'} \neq \emptyset$. We conclude that i and k never win together, therefore the mechanism has the 2-*wise property*.

The proof the 2-wise property implies that the mechanism stops after at most $2 \ln \bar{v}_{max} + 1$ stages is similar to the proof of Lemma 7. □

For a set of agents T, let $\omega^*(T, s)$ denote the value of the efficient outcome (w.r.t. the true valuation functions) for the set of agents T when each agent i is restricted to get a bundle that is a subset of s_i.

For a set of agents T we denote $V(T, v) = \sum_{i \in T} v_i$.

Lemma 14. *For general Combinatorial Auctions, in the 1-CA Japanese Mechanism it holds that* $\omega^*(L_J, s^J) \leq (e\hat{c}J)\omega^*(W_J, s^J)$*, where* $\hat{c} = 2\sqrt{m} + 2$.

Proof. Let R_j be the set of agents that retired at stage j, then it holds that $\cup_{j=1}^J R_j = L_J$. Thus $\omega^*(L_J, s^J) \leq \sum_{i=1}^J \omega^*(R_j, s^J)$. To prove the claim it is thus sufficient to show that for any stage j, $\omega^*(R_j, s^J) \leq e\hat{c}\omega^*(W_J, s^J)$. This is a direct result of the following two claims.

Claim 2. $V(W_j, v^j) \leq \omega^*(W_J, s^J)$.

Proof. Clearly $V(W_j, v^j) \leq V(W_{j+1}, v^{j+1})$ for any stage $j \in \{1, J-1\}$, thus $V(W_j, v^j) \leq V(W_J, v^J)$. By the undominated strategies characterization of Lemma 12 for each agent $i \in W_J$, $v_i(s_i^J) \geq v_i^J$, and all bundles of s^J that belong to agents in W_J are disjoint, thus $V(W_j, v^j) = \sum_{i \in W_j} v_i^j \leq \sum_{i \in W_j} v_i(s_i^J) = \omega^*(W_J, s^J)$. □

Claim 3. $\omega^*(R_j, s^J) \leq e\hat{c}V(W_j, v^j)$*, where* $\hat{c} = 2\sqrt{m} + 2$.

Proof. We look at some efficient allocation for the set R_j and partition the winners to two types. (note that for $i \in R_j$, $s_i^j = s_i^J$.) First, there are agents that are at either W_{j-1}, $W_j(1)$ or $W_j(2)$ but are not in W_j (because of the maximum taken at the output picking stage). Let $T \neq W_j$, $T \in \{W_{j-1}, W_j(1), W_j(2)\}$ be any one of the two sets of such agents. By the undominated strategies characterization of Lemma 12 for each agent $i \in T$, $v_i(s_i^j) \leq ev_i^j$, and all bundles of s^j that belong to agents in T are disjoint, thus $V(W_j, v^j) \geq V(T, v^j) = \sum_{i \in T} v_i^j \geq (1/e) \sum_{i \in T} v_i(s_i^j) = (1/e)\omega^*(T, s^j)$. Thus the value of both such T's is bounded by $2eV(W_j, v^j)$.

Secondly, there are agents that are not in any of W_{j-1}, $W_j(1)$ and $W_j(2)$. Since any allocation has at most $\sqrt{m}$ bundles of size at least $\sqrt{m}$, $eV(W_j(1), v^j)$ is an $e\sqrt{m}$ approximation for the set of agents X such that any $i \in X$ does not desired any bundle in s_i^j of size at most $\sqrt{m}$.

For agents not in X (agent with some desired bundle in s_i^j of size at most $\sqrt{m}$), $V(W_j(2), v^j)$ is a $(e\sqrt{m})$-approximation. This follows from the undominated strategies characterization, which implies that any such agent i does not have a declaration of s_i^{j+1} which will cause him to be in $W_j(2)$. Any winner in the greedy has a higher value (up to e factor) than any loser that intersects him, and he intersects at most $\sqrt{m}$ such losers.

We conclude that $V(W_j, v^j)$ is a $2e$-approximation to the agents of the first type, and $2e\sqrt{m}$-approximation to the agents of the second type, and this concludes the proof. □

Using all above claims:

$$\omega^*(L_J, s^J) \leq \sum_{i=1}^J \omega^*(R_j, s^J) \leq \sum_{i=1}^J e\hat{c}\omega^*(W_J, s^J) = (e\hat{c}J)\omega^*(W_J, s^J)$$

which concludes the proof of the lemma. $\square$

Observation 6. *For general Combinatorial Auctions, the 1-CA Japanese Wrapper Mechanism implements an $e\delta \min(m,n)$-approximation in undominated strategies.*

Proof. Let k be the number of winners in some efficient allocation. Clearly $k \leq \min(m,n)$, since each winner receives at least one item. The mechanism allocation is at least as good as the $\bar{v}_{max}/(e\delta)$ (a lower bound on the value reported by the agent with the highest value, if he loses), and the value of the efficient allocation is at most $k\bar{v}_{max}$, since any winning agent has a value of at most $\bar{v}_{max}$. $\square$

From the above we conclude the following.

Theorem 15. *For Unknown Single-Minded CA, the 1-CA Japanese Wrapper Mechanism implements a $(e\hat{c}J+1)$-approximation in undominated strategies, where $J \leq 2\ln\bar{v}_{max}+1$ is the number of iterations of the mechanism and $\hat{c} = 2\sqrt{m}+2$.*

It also implements an $e\min(m,n)$-approximation.

Proof. Let O be the set of winners in some efficient allocation. We partition them to two - agents that also win in the mechanism, and agents that lose. For the first set of agents ($O \cap W_J$), $\omega^*(O\cap W_J, s^J) \leq \omega^*(W_J, s^J)$. For the second set of agents ($O \cap L_J$), by Lemma 14, $\omega^*(O\cap L_J, s^J) \leq \omega^*(L_J, s^J) \leq (e\hat{c}J)\omega^*(W_J, s^J)$ (since by undominated strategies characterization, for single-minded agents, the desired bundle of each agent i is a subset of s_i^J).

The $e\min(m,n)$-approximation is derived from Observation 6 and the fact that $\delta = 1$ for single-minded agents. $\square$

Theorem 16. *For general Combinatorial Auctions, the 1-CA Japanese Wrapper Mechanism implements a $(e\delta\hat{c}J^2+e\hat{c}J+\delta)$-approximation in undominated strategies, where $J \leq 2\ln\bar{v}_{max}+1$ is the number of iterations of the mechanism and $\hat{c} = 2\sqrt{m}+2$. Thus, it implements $O(\delta(\ln\bar{v}_{max})^2\sqrt{m})$-approximation.*

It also implements an $e\delta\min(m,n)$-approximation.

Proof. To prove the theorem we need an additional claim. Assume that the winners in the efficient allocation win with bundles s^O.

Lemma 15. *We define the set of agents Q to be the set of losers in the 1-CA Japanese Mechanism with the following property. Agent $i \in Q$ if and only if i wins in the efficient allocation with bundle s_j^O which is not contained in the bundle s_i^J (the bundle agent i retired with).*

Then,

$$(e\delta\hat{c}J^2)\omega^*(W_J, s^J) \geq \omega^*(Q, s^O)$$

for $\hat{c} = \sqrt{m}+2$.

Proof. For each agent $i \in Q$, $s_i^j \neq G$. Let $Q_j = \{i \in N | s_i^j = G \text{ and } s_i^{j+1} \neq G\}$, this is the set of agents that first declared a bundle that is not all the goods at stage j. Note that $Q = \cup_{j=1}^J Q_j$.

All bundles s^{j+1} of agents in Q_j are disjoint. Additionally, $s_i^J \subseteq s_i^{j+1}$ thus all bundles s^J of agents in Q_j are disjoint. $Q_j \subseteq L_J$ thus $\omega^*(L_J, s^J) \geq \omega^*(Q_j, s^J)$.

Additionally, by the first property of undominated strategies presented in Lemma 12 (and the definition of δ), for any $i \in Q_j$, $\delta v_i(s_i^J) \geq v_i(s_i^O)$. We conclude that $\delta\omega^*(L_J, s^J) \geq \delta\omega^*(Q_j, s^J) \geq \omega^*(Q_j, s^O)$.

By Lemma 14, $(e\hat{c}J)\omega^*(W_J, s^J) \geq \omega^*(L_J, s^J)$, therefore $(e\delta\hat{c}J)\omega^*(W_J, s^J) \geq \omega^*(Q_j, s^O)$. We conclude that

$$\omega^*(Q, s^O) \leq \sum_{j=1}^{J} \omega^*(Q_j, s^O) \leq \sum_{j=1}^{J} (e\delta\hat{c}J)\omega^*(W_J, s^J) \leq (e\delta\hat{c}J^2)\omega^*(W_J, s^J)$$

□

The winners in the efficient allocation win with bundles s^O. We partition them to three and bound each separately: Agents that are also in W_J, agents in Q and agents in $L_J \setminus Q$. Agent in W_J might win other bundles in the efficient allocation, so $\omega^*(W_J, s^O) \leq \delta\omega^*(W_J, s^J)$. By definition of Q, all agents in $L_J \setminus Q$ win bundles in s^J, with Lemma 14 this implies that $\omega^*(L_J \setminus Q, s^O) \leq \omega^*(L_J \setminus Q, s^J) \leq \omega^*(L_J, s^J) \leq (e\hat{c}J)\omega^*(W_J, s^J)$. By Lemma 15, $\omega^*(Q, s^O) \leq (e\delta\hat{c}J^2)\omega^*(W_J, s^J)$. So we conclude that the mechanism is a $(e\delta\hat{c}J^2 + e\hat{c}J + \delta)$-approximation. By Lemma 13 it holds that $J \leq 2\ln\bar{v}_{max} + 1$ and the mechanism runs in polynomial time, this concludes the proof. □

Since unknown single-parameter multi-minded CA is a special case of general CA with $\delta = 1$, we conclude:

Corollary 3. *For USP multi-minded CA, the 1-CA Japanese Wrapper Mechanism implements a* $(e\hat{c}J^2 + e\hat{c}J + 1)$*-approximation in undominated strategies, where* $J \leq 2\ln\bar{v}_{max} + 1$ *is the number of iterations of the mechanism and* $\hat{c} = 2\sqrt{m} + 2$*. Thus, it implements* $O((\ln\bar{v}_{max})^2\sqrt{m})$*-approximation.*
It also implements a $\min(em, en)$*-approximation.*

We note that if the agents in a general CA are "almost" single-parameter (if δ is a constant), then the approximation is asymptotically the same as for the single-parameter multi-minded CA domain.

C.2 An Impersonation-Based Mechanism for Unknown Single Minded CA

Next, we demonstrate a completely different technique for implementation in undominated strategies for the single minded CA domain. This technique is given in figure 5. The main idea is to allow players to impersonate several different single minded agents. The resulting mechanism is an example for a "direct value mechanism": players are required to state their value, plus some extra information about their bundle. It turns out that, for some carefully designed allocation rules which we demonstrate below, the players will still reveal their true value and their true bundle (coupled with extra "false" information), and, furthermore, this will enable us to maintain the original approximation ratio.

Let $M(G, K)$ denote the mechanism created by running the Impersonation-Based Mechanism with an allocation rule G and an integer K (in any case that its allocation is value monotonic the mechanism is well defined).

Lemma 16. *Assume that for some allocation algorithm* G *and a number* K*, the* $M(G, K)$ *mechanism is value monotonic. Then for any agent* i*, in any undominated strategy,* $s_i^1 \supseteq \bar{s}_i$ *and* $v_i = \bar{v}_i$*.*

Proof. We first show that for any agent i, in any undominated strategy, $\bar{s}_i \subseteq s_i^1$. In any such strategy his utility is always non negative. If this is not the case, when the agent wins, he is not satisfied, thus having negative utility (in any case he wins he pays a positive amount and his valuation is zero). In cases he loses his utility is zero. We conclude that he always has non positive utility, and such a strategy is dominated.

Next we show that for any agent i, in any undominated strategy, $v_i = \bar{v}_i$. The mechanism is value monotonic, the payments are by critical values and it ensures losers unsatisfaction (since any loser gets $\emptyset$). By Lemma 3, for any fixed bundles he reports, bidding $\bar{v}_i$ dominates bidding v_i (it is possible to build cases in which bidding $\bar{v}_i$ strictly dominates bidding v_i). □

Definition 14. *The mechanism $M(G, K)$* encourages minimal bundle bidding *if for any agent i in any undominated strategy $s_i^1 = \bar{s}_i$.*

Theorem 17. *Suppose that an allocation rule G is a c-approximation algorithm for KSM CA, and that the mechanism $M(G, K)$ is value monotonic, encourages minimal bundles bidding and has the fast undominance recognition property, then it implements a c-approximation in undominated strategies.*

Proof. Let t be the true types of the players, and let t' be the sequence of single minded bidders constructed from the players' bids. Since $M(G, K)$ is value monotonic, then, in any undominated strategy, each player will report his true value. Since $M(G, K)$ encourages minimal bundle bidding, s_i^1 will be the true bundle of player i for all i. Therefore $t \subseteq t'$, i.e. all the actual players of t exist in t'. Let $v^*(t), v^*(t')$ be the optimal efficiency according to t, t', respectively. We first claim that $v^*(t) = v^*(t')$: $v^*(t) \leq v^*(t')$ as t' includes all players of t, and perhaps more. On the other hand, $v^*(t) \geq v^*(t')$, since we can convert any allocation in t' to an allocation in t with the same value – choose the same winners and allocate them their bundles in t. This is still a valid allocation as allocated bundles only decreased, and it has the same value. Thus $v^*(t) = v^*(t')$. Since $G(t')$ produces an allocation with value at least $v^*(t')/c$, the theorem follows.

Finally, we note that since G runs in polynomial time, then the mechanism runs in polynomial time. The allocation is clearly polynomial time computable. The payment of each agent can be calculated by running the allocation at most $\log(\bar{v}_{max})$ times using a binary search. □

What allocation rules can we "plug-in" to our mechanism so that it will be value monotonic? It is not sufficient that the allocation rule itself is value monotonic[20] . We demonstrate two allocation rules that can be used.

An axis-parallel CA of the plane. In an axis-parallel rectangles CA, the set of goods is the set of points in the plane $\Re^2$, and each bundle an agent may desire is an axis-parallel rectangle in the plane. With the Impersonation-Based mechanism technique, we are able to achieve the original $O(\log(R))$ approximation ratio of the original shifting algorithm (see Figure 8) in undominated strategies. Throughout this part, we assume that the sub-procedure to find the optimal interval allocation breaks ties in favor of contained rectangles (if a rectangle contains another rectangle, than the later win be chosen if possible). Let $Shifting$ be the shifting allocation algorithm of Figure 8, and $M(Shifting, \log(R))$ be the Impersonation-Based mechanism based on $Shifting$, with $K = \log(R)$.

[20]E.g. the 1-CA of [13] will not do: suppose that in a CA with three agents and 10 goods, agent 1 bids 6 for the set $\{g_1\}$, agent 2 bids 5 for the sets $\{g_2\}$ and $\{g_1, g_2\}$, and agent 3 bids 10 for all the goods. In that case agent 1 wins $\{g_1\}$ and agent 2 wins $\{g_2\}$. If agent 2 increases his reported value to 7, then he become a loser since agent 3 wins alone (since the value of the greedy decreases from 11 to 7, and the maximal value remains 10) .

The Shifting Algorithm for KSM agents:
Input:
A vector of values v^j and a vector of axis-parallel rectangles s^j (with one element for each agent).
Procedure:

1. Divide the given rectangles to $\log(R)$ classes such that a class $c \in \{1, \ldots, \log(R)\}$ consists of all rectangles with heights in $[W \cdot 2^{c-1}, W \cdot 2^c)$ (where the height of an axis-parallel rectangle is its projection on the y-axis).[a]
2. For each class $c \in \{1, \ldots, \log(R)\}$, run the Class Shifting Algorithm (Figure 9) on the class c, where the input is the vector of values v^j and a vector of axis-parallel rectangles from s^j with height in $[W \cdot 2^{c-1}, W \cdot 2^c)$, to get an allocation $W_j(c)$.

Output:
Output the maximal value solution $W_j(c)$ (with respect to v^j), over all classes $c \in \{1, \ldots, \log(R)\}$.

[a]Assume that the last class contains also the rectangles of height $W \cdot 2^{\log(R)}$.

Figure 8: The Shifting Algorithm for KSM agents.

The Class Shifting Algorithm:
Input:
A class number c.
A vector of values v^j and a vector of axis-parallel rectangles, each of height in $[W \cdot 2^{c-1}, W \cdot 2^c)$.
Procedure:

1. Superimpose a collection of horizontal lines with a distance of $W \cdot 2^{c+1}$ between consecutive lines. Each area between two consecutive lines is called a *slab*. Later, we shift this collection of lines downwards. Each location of these lines is called a *shift* and is denoted by h.
2. For any slab created, and for all the rectangles in this slab which do not intersect any horizontal line, project all the rectangles to the x-axis. Now, find the set of projections (intervals) that maximizes the sum of valuations w.r.t. v^j (this can be done in polynomial time using a dynamic-programming algorithm [4]). Let $V(h, l)$ be the value achieved in slab l of shift h.
3. Sum the efficiency achieved in all slabs in a shift to calculate the welfare of this shift. Denote the welfare achieved in shift h by $V(h) = \sum_{l \in h} V(h, l)$.
4. Shift down the collection of horizontal lines by W, and recalculate the welfare. Repeat the process 2^{c+1} times.

Output:
Output the set of agents winning in the shift with maximal value of $V(h)$, and their winning rectangles.

Figure 9: The Class Shifting Algorithm that computes an allocation that is a 6-approximation for the set of agents in class c (rectangles with height in $[W \cdot 2^{c-1}, W \cdot 2^c)$).

Theorem 18. *The $M(Shifting, \log(R))$ implements an $O(\log(R))$-approximation in undominated strategies.*

Proof. The claims below show that $M(Shifting, \log(R))$ is value monotonic, the mechanism encourages minimal bundle bidding and has the fast undominance recognition property, hence the claim follows by theorem 17.

Claim 4. *The allocation rule of the $M(Shifting, \log(R))$ mechanism is value monotonic.*

Proof. Suppose a winner i increases his value. Agent i remains a winner in the winning shift, and its value increases. The value of any shift (in any class) in which he remains a loser does not change. Thus, the shift with the maximal value (over all classes and shifts in each class) must contain i as a winner. □

Claim 5. *In the $M(Shifting, \log(R))$ mechanism, any agent i, in any undominated strategy, reports one minimal area bundle (minimal height and width) containing $\bar{s}_i$ in any class c for which $\bar{s}_i$ can belong to (a class c for which the height of $\bar{s}_i$ is greater or equal to $W \cdot 2^{c-1}$).*

Additionally, the mechanism has the fast undominance recognition property.

Proof. Since the mechanism is value monotonic (Claim 4), by Lemma 16, for any agent i, in any undominated strategy, $\bar{s}_i \subseteq s_i^1$. This implies that in any undominated strategy, i never bids in a class c for which $W \cdot 2^{c-1}$ is smaller than the height of $\bar{s}_i$.

Next we look only at some class c for which $\bar{s}_i$ can belong to ($W \cdot 2^{c-1}$ is at least the height of $\bar{s}_i$). We show that any strategy in which s_i^q denotes the minimal area bundle agent i bids for in a class c, is dominated by a strategy in which i replaces s_i^q by some minimal area bundle $s_i \supseteq \bar{s}_i$ in class c.

In any case that i wins with a bundle not in class c, replacing the bundles in class c cannot make i a loser (since only allocations with i as a winner are improved).

Next, assume that agent i wins the bundle s_i^q which belongs to class c. Replacing s_i^q by some minimal area bundle s_i in class c, such that $\bar{s}_i \subseteq s_i \subseteq s_i^q$, can never cause i to lose (in any shift from which s_i^q was not removed, also s_i will not be removed. The same shift has i winning and at least the same value as before, and the value of any shift with i losing does not increase).

Agent i can remove any non minimal area bundles in any class, since this will never change the outcome. Additionally, adding bundles can never make him worse off. This implies that since i can bid for up to $\log(R)$ bundles, in any undominated strategy, he will bid (a minimal area bundle, as we have shown) in any class c such that $W \cdot 2^{c-1}$ is at least the height of $\bar{s}_i$.

Finally we show that the mechanism has the computational undominated transfer property. Given a strategy of agent i, he can change it to an undominated strategy as follows. He replaces the value by $\bar{v}_i$. In each class, he removes all bundles of that class and bid for a minimal area bundle in that class that is contained in all these bundles, and contains $\bar{s}_i$. Any such strategy is undominated. □

Corollary 4. *The $M(Shifting, \log(R))$ mechanism encourages minimal bundle bidding.*

Proof. By the above claim, any agent i, in any undominated strategy, bids a minimal area bundle that contains $\bar{s}_i$ in any class that such a bundle exists, and does not bid in any smaller class. The only minimal area bundle that contains $\bar{s}_i$ in the minimal class for which such a bundle exists (the "true" class of i) is $\bar{s}_i$. Thus, in any undominated strategy, i bids $s_i^1 = \bar{s}_i$. □

This completes the proof of the theorem. ☐

A modified k-CA allocation rule for general USM-CA. Unfortunately, the Impersonation Based mechanism with the k-CA auction is not value monotonic. In order to turn it to such, we use our technique of Figure 7 to convert allocation rules to be value monotone. Recall that, given an allocation rule G, this method maintains $\ln \bar{v}_{max}$ classes, where the i'th class "sees" only players with value of at least 2^{i-1}. Each class computes an allocation according to an unweighted version of G (i.e. all players have value of 1), and the allocation of the class with maximal value is chosen.

Definition 15 (The impersonation-adjusted k-CA)**.** The impersonation-adjusted k-CA allocation rule $IA-k-CA$ *is defined to be the Bitonic Unweighted Allocation Rule of Figure 7, on top of the k-CA allocation rule of [13] with the following tie breaking rules:*

- *The greedy algorithm favors larger bundles over smaller bundles.*
- *The exhaustive-k algorithm favors smaller bundles over larger bundles.*

For a fixed ϵ, k is chosen to make the k-CA a $\epsilon\sqrt{m}$-approximation for the KSM CA model. By Theorem 13, the impersonation-adjusted k-CA is a thus a $(e\epsilon\sqrt{m}\ln \bar{v}_{max})$-approximation for the KSM CA model.

We use the impersonation-adjusted k-CA as the allocation rule for our Impersonation-Based mechanism. We set the parameter K of the mechanism to be $\sqrt{m}$. Effectively, this enables any agent to bid for any chain he might desire to bid for, as we later show.

Theorem 19. *The $M(IA-k-CA, \sqrt{m})$ mechanism implements a $(e\epsilon\sqrt{m}\ln \bar{v}_{max})$-approximation in undominated strategies.*

Proof. The following claims show that $M(IA-k-CA, \sqrt{m})$ is value monotonic, it encourages minimal bundle bidding and it has the computational undominated transfer property, hence the claim follows by theorem 17.

Claim 6. *The allocation rule of the $M(IA-k-CA, \sqrt{m})$ mechanism is value monotonic.*

Proof. Mu'alem and Nisan [13] show that for KSM CA, any allocation rule that takes the maximal allocation over a set of bitonic allocation rules is monotonic. We use this and show that each of the $\ln \bar{v}_{max}$ allocation rules (one in each class) is bitonic (when agents bid several bundles). Essentially, each of these unweighted allocation rules is bitonic, since it is bitonic when each agent bids one bundle to the exhaustive-k and some bundle that contains the first in the greedy (it is not the same since large bundles are considered first). Next, we formally prove the claim directly.

Bitonicity means that if an agent bids a value and a set of bundles and wins, he remains a winner if he increases his value (monotonicity), and the value of the allocation does not decrease. If he loses and increases his value and remains a loser, the value of the allocation does not decrease.

Let us concentrate on one class. For an agent to win in that class, he must appear with a value 1, which can not increase, so the allocation is monotonic, and the value never changes if a winner increases his bid.

If an agent i is a loser and can increase his value, this means that he first appears with value 0 and then with value 1. There are two cases, either the greedy wins or the exhaustive-k wins after the value change. If the greedy wins but i loses, this means that i loses in the greedy, thus the value of the greedy does not change by adding i and was maximal before the change (the value of the exhaustive-k can only increase by adding i). So we conclude that the value of the allocation does not change.

Next we consider the case that the exhaustive-k wins (is maximal after the change). Since i loses in the exhaustive-k, the value of the exhaustive-k does not change. If the exhaustive-k was maximal before the change, then the value of the allocation does not change. If the greedy was maximal before the change, it must be the case that its value decreases (since the value of the exhaustive-k is fixed), so the value of the allocation decreases. □

Since the mechanism is value monotonic, in any undominated strategy, agent i will report his true value $\bar{v}_i$.

Claim 7. *In the $M(IA-k-CA, \sqrt{m})$ mechanism, for any agent i, any strategy where i bids the value $\bar{v}_i$ and a sequence of $k_i \leq \sqrt{m}$ bundles, $s_i^1 \subset s_i^2 \subset \ldots \subset s_i^{k_i}$, $s_i^1 \neq \bar{s}_i$, is dominated by the strategy in which i bids $\bar{v}_i$ and*

- *removes all bundles that does not contain $\bar{s}_i$.*
- *adds $\bar{s}_i$ as his first reported bundle.*
- *removes all reported bundles different from $\bar{s}_i$ of size larger than $\sqrt{m}$.*

Thus, the mechanism has the fast undominance recognition property and it encourages minimal bundle bidding.

Proof. Since the mechanism is value monotonic (Claim 6), by Lemma 16, for any agent i, in any undominated strategy, $\bar{s}_i \subseteq s_i^1$. This implies that any strategy in which an agent i bids any bundle that does not contain $\bar{s}_i$, is dominated by a strategy in which such a bundle is removed. So from this point we assume that each reported bundle contains $\bar{s}_i$, so i wins if and only if he is satisfied.

Next, note that it is possible to add $\bar{s}_i$ and remove all bundles larger than $\sqrt{m}$ while keeping $k_i \leq \sqrt{m}$. This holds since there are at most $\sqrt{m}-1$ bundles in any chain of bundles, each of size at most $\sqrt{m}$, and $\bar{s}_i \subset s_i^1$ (the chain does not include $\bar{s}_i$).

We show that in any case that i wins with the current bid, he also wins if he changes his bid by adding $\bar{s}_i$ as the first reported bundle and removing all bundles larger than $\sqrt{m}$.

In any class, in the exhaustive-k, if i wins with some bundle that contains $\bar{s}_i$ before the change, he also wins with $\bar{s}_i$ after the change. Only agents with $|\bar{s}_i| \leq \sqrt{m}$ might win in the greedy algorithm before the change. In any class, in the greedy algorithm, if i wins with some bundle that contains $\bar{s}_i$ before the change, he also wins with the same bundle after the change (this bundle is not removed). Adding $\bar{s}_i$ might only cause i to win in some cases that he lost before the change.

Note that the lemma shows that if $|\bar{s}_i| > \sqrt{m}$, the strategy of reporting $\bar{v}_i$ and $s_i^1 = \bar{s}_i$ ($k_i = 1$) is a dominant strategy for agent i. □

This completes the proof of the theorem. □

C.3 Undominated Strategies in Direct Value Revelation Mechanisms for USP domains

Definition 16. *A mechanism M for USP model is* Ex-ante individually rational *if for each agent i, there* **exists** *an undominated strategy that ensures non negative utility (for any strategies of the other agents). A mechanism M for USP model is* Ex-post individually rational *if for each agent i,*

any *undominated strategy ensures non negative utility (for any strategies of the other agents), and there* **exists** *at least one undominated strategy for agent* i.

In this section we present some general characterizations for undominated strategies in satisfying guessing mechanisms for general USP domains. The mechanisms presented in the Section C.2 are examples for such mechanisms.

Lemma 17. *If a satisfying guessing mechanism* M *for USP model is value monotonic, the payments are by critical values, and for each agent* i *there exists a strategy* $v_i = \bar{v}_i$ *and* B_i *such that* $S_i(B_i) \subseteq \bar{A}_i$ *that is undominated, then* M *is Ex-ante individually rational.*

If any undominated strategy for agent i *is in the form* $v_i = \bar{v}_i$ *and* B_i *such that* $S_i(B_i) \subseteq \bar{A}_i$*, then* M *is Ex-post individually rational.*

Additionally, if i *declares* $v_i = \bar{v}_i$ *and* B_i *such that* $S_i(B_i) \subseteq \bar{A}_i$ *then* i *has non negative utility.*

Proof. We show that if an agent i declares $v_i = \bar{v}_i$ and B_i such that $S_i(B_i) \subseteq \bar{A}_i$ then he has a non negative utility. Since the mechanism is monotonic and the payments are by critical values, agent i loses and pays zero if $\bar{v}_i < c_i(S_i(B_i)|v_{-i}, B)$, and wins and pays $c_i(S_i(B_i)|v_{-i}, B)$ if $\bar{v}_i > c_i(S_i(B_i)|v_{-i}, B)$ (if this holds as equality his utility is zero).

Since $S_i(B_i) \subseteq \bar{A}_i$, if agent i wins then he is satisfied. Therefore, if i is not satisfied then i loses, pays zero and has non negative utility (since his value for that outcome is non negative). If agent i wins then $\bar{v}_i \geq c_i(\bar{A}_i|v_{-i}, B)$, and i pays his critical value $c_i(\bar{A}_i|v_{-i}, B)$ so his utility is $\bar{v}_i - c_i(\bar{A}_i|v_{-i}, B) \geq 0$. □

Lemma 18. *If a satisfying guessing mechanism* M *for USP model is value monotonic, the payments are by critical values and the mechanism ensures losers unsatisfaction, then for each agent* i*, any strategy in which* i *declares* $v_i \neq \bar{v}_i$ *and* B_i *such that* $S_i(B_i) \subseteq \bar{A}_i$ *is dominated.*

Proof. We show that any strategy in which i declares $v_i \neq \bar{v}_i$ and B_i such that $S_i(B_i) \subseteq \bar{A}_i$ is dominated by the strategy $\bar{v}_i$ and B_i.

Since $S_i(B_i) \subseteq \bar{A}_i$, if i is not satisfied then i loses, pays zero and has non negative utility (since his value for that outcome is non negative). If $v_i < c_i(S_i(B_i)|v_{-i}, B)$, i loses and has zero utility when declaring v_i. By Lemma 17 i has non negative utility if he changes his bid to $\bar{v}_i$ and B_i (he will also have zero utility if $v_i = c_i(S_i(B_i)|v_{-i}, B)$).

If $v_i > c_i(S_i(B_i)|v_{-i}, B) \geq \bar{v}_i$, i wins, is satisfied and has negative utility when declaring v_i. This can be improved to zero utility if he changes his bid to $\bar{v}_i$ and B_i (since he will lose or win and pay his value). If $v_i > c_i(S_i(B_i)|v_{-i}, B)$ and $\bar{v}_i > c_i(S_i(B_i)|v_{-i}, B)$, i wins, is satisfied and pays the same with both declarations, so he has the same utility. □

The following condition makes sure that agents alleged satisfying set is contained in their satisfying set.

Definition 17. *A satisfying guessing mechanism* M *for USP model that is value monotonic* prevents unsatisfying outcomes declaration, *if for all* $i \in N, B \in \mathcal{B}, v \in \Re^n_{++}, \bar{A}_i \in \mathcal{A}_i$*, if* i *bids* v_i *and* B_i *such that* $S_i(B_i)$ *is not contained in* $\bar{A}_i$ *and* $G(v, B) \in S_i(B_i) \cap \bar{A}_i$ *(*i *is satisfied and he wins) then for any* $\tilde{B}_i$ *such that* $S_i(\tilde{B}_i) \subseteq \bar{A}_i$*,* $c_i(S_i(\tilde{B}_i)|v_{-i}, (\tilde{B}_i, B_{-i})) \leq c_i(S_i(B_i)|v_{-i}, B)$.

Theorem 20. *If a satisfying guessing mechanism* M *for USP model is value monotonic, the payments are by critical values, the mechanism ensures losers unsatisfaction and it prevents unsatisfying outcomes declaration, then for each agent* i*, in any undominated strategy* i *declares* $\bar{v}_i$ *and* $\tilde{B}_i$ *such that* $S_i(\tilde{B}_i) \subseteq \bar{A}_i$.

Proof. First we show that if i bids v_i and B_i such that $S_i(B_i)$ is not contained in $\bar{A}_i$ then this strategy is dominated by any strategy v_i and $\tilde{B}_i$ such that $S_i(\tilde{B}_i) \subseteq \bar{A}_i$. Next, by Lemma 18 this new strategy is dominated by the strategy $\bar{v}_i$ and $\tilde{B}_i$, which conclude the proof of the claim.

Assume that i bids v_i and B_i such that $S_i(B_i)$ is not contained in $\bar{A}_i$. If i loses its utility is zero (since the mechanism ensures losers unsatisfaction). If i wins he is either satisfied or not ($S_i(B_i)$ is not contained in $\bar{A}_i$). In case he is not satisfied, his utility is non positive since he pays non negative payment and has a valuation of zero for the outcome. In case he is satisfied, his utility is $\bar{v}_i - c_i(S_i(B_i)|v_{-i}, B)$.

Assume that i bids v_i and $\tilde{B}_i$ such that $S_i(\tilde{B}_i) \subseteq \bar{A}_i$ instead of bidding v_i and B_i. By Lemma 17 agent i has non negative utility. The only case that the bid v_i and B_i results with positive utility is the case that the agent wins and is satisfied and his utility is $\bar{v}_i - c_i(S_i(B_i)|v_{-i}, B) > 0$. Since the mechanism prevents unsatisfying outcomes declaration, the utility is improved by bidding v_i and $\tilde{B}_i$. In that case $v_i > c_i(S_i(B_i)|v_{-i}, B)$ therefore $v_i > c_i(S_i(\tilde{B}_i)|v_{-i}, (\tilde{B}_i, B_{-i}))$ so i wins and is satisfied by the new bid (since $S_i(\tilde{B}_i) \subseteq \bar{A}_i$). His utility improves since his valuation for the outcome is the same but he pays no more (since $c_i(S_i(\tilde{B}_i)|v_{-i}, (\tilde{B}_i, B_{-i})) \leq c_i(S_i(B_i)|v_{-i}, B)$). □

The following condition makes sure that agents alleged satisfying set is their satisfying set.

Definition 18. *A satisfying guessing mechanism M for USP model that is value monotonic* prevents untruthful alleged satisfying set, *if for all $i \in N, B \in \mathcal{B}, v \in \Re^n_{++}, \bar{A}_i \in \mathcal{A}_i$, if i bids v_i and B_i such that $S_i(B_i) \neq \bar{A}_i$ and $G(v, B) \in S_i(B_i) \cap \bar{A}_i$ (i is satisfied and he wins) then for any $\bar{B}_i$ such that $S_i(\bar{B}_i) = \bar{A}_i$, $c_i(\bar{A}_i|v_{-i}, (\bar{B}_i, B_{-i})) \leq c_i(S_i(B_i)|v_{-i}, B)$.*

Theorem 21. *If a satisfying guessing mechanism M for USP model is value monotonic, the payments are by critical values, the mechanism ensures losers unsatisfaction and it prevents untruthful alleged satisfying set, then M is individually rational and for each agent i, in any undominated strategy i declares $\bar{v}_i$ and $\bar{B}_i$ such that $S_i(\bar{B}_i) = \bar{A}_i$.*

Proof. The proof is similar to the proof of Theorem 20. □