Algorithmic Mechanism Design (1999; Nisan, Ronen)

Ron Lavi
The Technion – Israel Institute of Technology
http://ie.technion.ac.il/~ronlavi

INDEX TERMS: mechanism design, multi-item auctions, incentives for strategic agents.

1 PROBLEM DEFINITION

Mechanism design is a sub-field of economics and game theory that studies the construction of social mechanisms in the presence of selfish agents. The nature of the agents dictates a basic contrast between the social planner, that aims to reach a socially desirable outcome, and the agents, that care only about their own private utility. The underlying question is how to incentivize the agents to cooperate, in order to reach the desirable social outcomes.

In the Internet era, where computers act and interact on behalf of selfish entities, the connection of the above to algorithmic design suggests itself: suppose that the input to an algorithm is kept by selfish agents, who aim to maximize their own utility. How can one design the algorithm so that the agents will find it in their best interest to cooperate, and a close-to-optimal outcome will be outputted? This is different than classic distributed computing models, where agents are either "good" (meaning obedient) or "bad" (meaning faulty, or malicious, depending on the context). Here, no such partition is possible. It is simply assumed that all agents are utility maximizers. To illustrate this, let us describe a motivating example:

A motivating example: shortest paths Given a weighted graph, the goal is to find a shortest path (with respect to the edge weights) between a given source and target nodes. Each edge is controlled by a selfish entity, and the weight of the edge, w_e is private information of that edge. If an edge is chosen by the algorithm to be included in the shortest path, it will incur a cost which is minus its weight (the cost of communication). Payments to the edges are allowed, and the total utility of an edge that participates in the shortest path and gets a payment p_e is assumed to be $u_e = p_e - w_e$. Notice that the shortest path is *with respect to the true weights of the agents, although these are not known to the designer.*

Assuming that each edge will act in order to maximize its utility, how can one choose the path and the payments? One option is to ignore the strategic issue all together, ask the edges to simply report their weights, and compute the shortest path. In this case, however, an edge dislikes being selected, and will therefore prefer to report a very high weight (much higher than its true weight) in order to decrease the chances of being selected. Another option is to pay each selected edge its reported weight, or its reported weight plus a small fixed "bonus". However in such a case all edges will report lower weights, as being selected will imply a positive gain.

Although this example is written in an algorithmic language, it is actually a mechanism design problem, and the solution, which is now a classic, was suggested in the 70's. The chapter continues as follows: First, the abstract formulation for such problems is given, the classic solution from economics is described, and its advantages and disadvantages for algorithmic purposes are discussed. The next section then describes the new results that algorithmic mechanism design offers.

Abstract Formulation The framework consists of a set A of alternatives, or outcomes, and n players, or agents. Each player i has a valuation function $v_i : A \to \Re$ that assigns a value to each possible alternative. This valuation function belongs to a domain V_i of all possible valuation functions. Let $V = V_1 \times \cdots \times V_n$, and $V_{-i} = \prod_{j \neq i} V_j$. Observe that this generalizes the shortest path example of above: A is all the possible $s - t$ paths in the given graph, $v_e(a)$ for some path $a \in A$ is either $-w_e$ (if $e \in a$) or zero.

A *social choice function* $f : V \to A$ assigns a socially desirable alternative to any given profile of players' valuations. This parallels the notion of an algorithm. A *mechanism* is a tuple $M = (f, p_1, ..., p_n)$, where f is a social choice function, and $p_i : V \to \Re$ (for $i = 1, ..., n$) is the price charged from player i. The interpretation is that the social planner asks the players to reveal their true valuations, chooses the alternative according to f as if the players have indeed acted truthfully, and in addition rewards/punishes the players with the prices. These prices should induce "truthfulness" in the following strong sense: no matter what the other players declare, it is always in the best interest of player i to reveal her true valuation, as this will maximize her utility. Formally, this translates to:

Definition 1 (Truthfulness)**.** *M is "truthful" (in dominant strategies) if, for any player i, any profile of valuations of the other players $v_{-i} \in V_{-i}$, and any two valuations of player i $v_i, v_i' \in V_i$,*

$$v_i(a) - p_i(v_i, v_{-i}) \geq v_i(b) - p_i(v_i', v_{-i})$$

where $f(v_i, v_{-i}) = a$ and $f(v_i', v_{-i}) = b$.

Truthfulness is quite strong: a player need not know anything about the other players, even not that they are rational, and still determine the best strategy for her. Quite remarkably, there exists a truthful mechanism, even under the current level of abstraction. This mechanism suits all problem domains, where the social goal is to maximize the "social welfare":

Definition 2 (social welfare maximization)**.** *A social choice function $f : V \to A$ maximizes the social welfare if $f(v) \in argmax_{a \in A} \sum_i v_i(a)$, for any $v \in V$.*

Notice that the social goal in the shortest path domain is indeed welfare maximization, and, in general, this is a natural and important economic goal. Quite remarkably, there exists a general technique to construct truthful mechanisms that implement this goal:

Theorem 1 (Vickrey-Clarke-Groves (VCG))**.** *Fix any alternatives set A and any domain V, and suppose that $f : V \to A$ maximizes the social welfare. Then there exist prices p such that the mechanism (f, p) is truthful.*

This gives "for free" a solution to the shortest path problem, and to many other algorithmic problems. The great advantage of the VCG scheme is its generality: it suits *all* problem domains. The disadvantage, however, is that the method is tailored to social welfare maximization. This turns out to be restrictive, especially for algorithmic and computational settings, due to several reasons: (i) different algorithmic goals: the algorithmic literature considers a variety of goals, including many that cannot be translated to welfare maximization. VCG does not help us in such cases. (ii) computational complexity: even if the goal is welfare maximization, in many settings achieving exactly the optimum is computationally hard. The CS discipline usually overcomes this by using approximation algorithms, but VCG will not work with such algorithm – reaching exact optimality is a necessary requirement of VCG. (iii) different algorithmic models: common CS models change "the basic setup", hence cause unexpected difficulties when one tries to use VCG (for example, an online model, where the input is revealed over time; this is common in CS, but changes the implicit setting that VCG requires). This is true even if welfare maximization is still the goal.

Answering any one of these difficulties requires the design of a non-VCG mechanism. What analysis tools should be used for this purpose? In economics and classic mechanism design, average-case analysis, that relies on the knowledge of the underlying distribution, is the standard. Computer science, on the other hand, usually prefers to avoid strong distributional assumptions, and to use worst-case analysis. This difference is another cause to the uniqueness of the answers provided by algorithmic mechanism design. Some of the new results that have emerged as a consequence of this integration between Computer Science and Economics is next described. Many other research topics that use the tools of algorithmic mechanism design are described in the entries on Adword Pricing, Competitive Auctions, False Name Proof Auctions, Generalized Vickrey Auction, Incentive Compatible Ranking, Mechanism for One Parameter Agents Single Buyer/Seller, Multiple Item Auctions, Position Auctions, and Truthful Multicast.

There are two different but closely related research topics that should be mentioned in the context of this entry. The first is the line of works that studies the "price of anarchy" of a given system. These works analyze *existing* systems, trying to quantify the loss of social efficiency due to the selfish nature of the participants, while the approach of algorithmic mechanism design is to understand how new systems should be designed. For more details on this topic the reader is referred to the entry on Price of Anarchy. The second topic regards the algorithmic study of various equilibria computation. These works bring computational aspects into economics and game theory, as they ask what equilibria notions are reasonable to assume, if one requires computational efficiency, while the works described here bring game theory and economics into computer science and algorithmic theory, as they ask what algorithms are reasonable to design, if one requires the resilience to selfish behavior. For more details on this topic the reader is referred (for example) to the entry on Algorithms for Nash Equilibrium and to the entry on General Equilibrium.

2 KEY RESULTS

2.1 Problem Domain 1: job scheduling.

Job scheduling is a classic algorithmic setting: n jobs are to be assigned to m machines, where job j requires processing time p_{ij} on machine i. In the game-theoretic setting, it is assumed that each machine i is a selfish entity, that incurs a cost p_{ij} from processing job j. Note that the payments in this setting (and in general) may be negative, offsetting such costs. A popular algorithmic goal is to assign jobs to machines in order to minimize the "makespan": $\max_i \sum_{j \text{ is assigned to } i} p_{ij}$. This is different than welfare maximization, which translates in this setting to the minimization of $\sum_i \sum_{j \text{ is assigned to } i} p_{ij}$, further illustrating the problem of different algorithmic goals. Thus the VCG scheme cannot be used, and new methods must be developed.

Results for this problem domain depend on the specific assumptions about the structure of the processing time vectors. In the *related machines* case, $p_{ij} = p_j/s_i$ for any i, j, where the p_j's are public knowledge, and the only secret parameter of player i is its *speed*, s_i.

Theorem 2 ([3], [22]). *For job scheduling on related machines, there exists a truthful exponential-time mechanism that obtains the optimal makespan, and a truthful polynomial-time mechanism that obtains a 3-approximation to the optimal makespan.*

More details on this result are given in the entry on Mechanism for One Parameter Agents Single Buyer. The bottom line conclusion is that, although the social goal is different than welfare maximization, there still exists a truthful mechanism for this goal. A non-trivial approximation guarantee is achieved, even under the additional requirement of computational efficiency. However, this guarantee does not match the best possible without the truthfulness requirement, since in this case a PTAS is known.

Open question 1. *Is there a truthful PTAS for makespan minimization in related machines?*

If the number of machines is fixed then [2] give such a truthful PTAS.

The above picture completely changes in the move to the more general case of *unrelated machines*, where the p_{ij}'s are allowed to be arbitrary:

Theorem 3 ([13], [30]). *Any truthful scheduling mechanism for unrelated machines cannot approximate the optimal makespan by a factor better than $1+\sqrt{2}$ (for deterministic mechanisms) and $2-1/m$ (for randomized mechanisms).*

Note that this holds regardless of computational considerations. In this case, switching from welfare maximization to makespan minimization results in a strong impossibility. On the possibilities side, virtually nothing (!) is known. The VCG mechanism (which minimizes the total social cost) is an m-approximation of the optimal makespan [32], and, in fact, nothing better is currently known:

Open question 2. *What is the best possible approximation for truthful makespan minimization in unrelated machines?*

What caused the switch from "mostly possibilities" to "mostly impossibilities"? Related machines is a single-dimensional domain (players hold only one secret number), for which truthfulness is characterized by a simple monotonicity condition, that leaves ample flexibility for algorithmic design. Unrelated machines, on the other hand, are a multi-dimensional domain, and the algorithmic conditions implied by truthfulness in such a case are harder to work with. It is still unclear whether these conditions imply real mathematical impossibilities, or perhaps just pose harder obstacles that can be in principle solved. One multi-dimensional scheduling domain for which possibility results are known is the case where $p_{ij} \in \{L_j, H_j\}$, where the "low"'s and "high"'s are fixed and known. This case generalizes the classic multi-dimensional model of restricted machines ($p_{ij} \in \{p_j, \infty\}$), and admits a truthful 3-approximation [27].

2.2 Problem Domain 2: digital goods and revenue maximization.

In the E-commerce era, a new kind of "digital goods" have evolved: goods with no marginal production cost, or, in other words, goods with unlimited supply. One example is songs being sold on the Internet. There is a sunk cost of producing the song, but after that, additional electronic copies incur no additional cost. How should such items be sold? One possibility is to conduct an *auction*. An auction is a one-sided market, where a monopolistic entity (the auctioneer) wishes to sell one or more items to a set of buyers.

In this setting, each buyer has a privately known value for obtaining one copy of the good. Welfare maximization simply implies the allocation of one good to every buyer, but a more interesting question is the question of revenue maximization. How should the auctioneer design the auction in order to maximize his profit? Standard tools from the study of revenue-maximizing auctions[1] suggest to simply declare a price-per-buyer, determined by the probability distribution of the buyer's value, and make a take-it-or-leave-it offer. However, such a mechanism needs to know the underlying distribution. Algorithmic mechanism design suggests an alternative, worst-case result, in the spirit of CS-type models and analysis.

Suppose that the auctioneer is required to sell all items in the same price, as is the case for many "real-life" monopolists, and denote by $F(\vec{v})$ the maximal revenue from a fixed-price sale to bidders with values $\vec{v} = v_1, ...v_n$, *assuming that all values are known*. Reordering indexes so that $v_1 \geq v_2 \geq ... \geq v_n$, let $F(\vec{v}) = \max_i i \cdot v_i$. The problem is, of-course, that in fact *nothing* about the values is known. Therefore, a truthful auction that extracts the players' values is in place. Can such an auction obtain a profit that is a constant fraction of $F(\vec{v})$, for any $\vec{v}$ (i.e. in the worst case)?

[1] This model was not explicitly studied in classic auction theory, but standard results from there can be easily adjusted to this setting.

Unfortunately, the answer is provably no [17]. The proof makes use of situations where the entire profit comes from the highest bidder. Since there is no potential for competition among bidders, a truthful auction cannot force this single bidder to reveal her value.

Luckily, a small relaxation in the optimality criteria significantly helps. Specifically, denote by $F^{(2)}(\vec{v}) = \max_{i \geq 2} i \cdot v_i$ (i.e. the benchmark is the auction that sells to at least two buyers).

Theorem 4 ([17, 20])**.** *There exists a truthful randomized auction that obtains an expected revenue of at least $F^{(2)}/3.25$, even in the worst-case. On the other hand, no truthful auction can approximate $F^{(2)}$ within a factor better than 2.42.*

Several interesting formats of distribution-free revenue-maximizing auctions have been considered in the literature. The common building block in all of them is the random partitioning of the set of buyers to random subsets, analyzing each set separately, and using the results on the other sets. Each auction utilizes a different analysis on the two subsets, which yields slightly different approximation guarantees. [1] describe an elegant method to derandomize these type of auctions, while losing another factor of 4 in the approximation. More details on this problem domain can be found in the entry on Competitive Auctions.

2.3 Problem Domain 3: combinatorial auctions.

Combinatorial auctions (CAs) are a central model with theoretical importance and practical relevance. It generalizes many theoretical algorithmic settings, like job scheduling and network routing, and is evident in many real-life situations. This new model has various pure computational aspects, and, additionally, exhibits interesting game theoretic challenges. While each aspect is important on its own, obviously only the integration of the two provides an acceptable solution.

A combinatorial auction is a multi-item auction in which players are interested in *bundles* of items. Such a valuation structure can represent substitutabilities among items, complementarities among items, or a combination of both. More formally, m items (Ω) are to be allocated to n players. Players value subsets of items, and $v_i(S)$ denotes i's value of a bundle $S \subseteq \Omega$. Valuations additionally satisfy: (i) monotonicity, i.e $v_i(S) \leq v_i(T)$ for $S \subseteq T$, and (ii) normalization, i.e. $v_i(\emptyset) = 0$. The literature has mostly considered the goal of maximizing the social welfare: find an allocation $(S_1, ..., S_n)$ that maximizes $\sum_i v_i(S_i)$.

Since a general valuation has size exponential in n and m, the representation issue must be taken into account. Two models are usually considered (see [11] for more details). In the *bidding languages* model, the bid of a player represents his valuation is a concise way. For this model it is NP-hard to approximate the social welfare within a ratio of $\Omega(m^{1/2-\epsilon})$, for any $\epsilon > 0$ (if "single-minded" bids are allowed; the exact definition is given below). In the *query access* model, the mechanism iteratively queries the players in the course of computation. For this model, any algorithm with polynomial communication cannot obtain an approximation ratio of $\Omega(m^{1/2-\epsilon})$ for any $\epsilon > 0$. These bounds are tight, as there exist a deterministic $\sqrt{m}$-approximation with polynomial computation and communication. Thus, for the general valuation structure, the computational status by itself is well-understood.

The basic incentives issue is again well-understood: VCG obtains truthfulness. Since VCG requires the exact optimum, which is NP-hard to compute, the two considerations therefore clash, when attempting to use classic techniques. Algorithmic mechanism design aims to develop new techniques, to integrate these two desirable aspects.

The first positive result for this integration challenge was given by [29], for the special case of "single-minded bidders": each bidder, i, is interested in a specific bundle S_i, for a value v_i (any bundle that contains S_i is worth v_i, and other bundles have zero value). Both v_i, S_i are private to the player i.

Theorem 5 ([29])**.** *There exists a truthful and polynomial-time deterministic combinatorial auction for single-minded bidders, which obtains a $\sqrt{m}$-approximation to the optimal social welfare.*

A possible generalization of the basic model is to assume that each item has B copies, and each player still desires at most one copy from each item. This is termed "multi-unit CA". As B grows, the integrality constraint of the problem reduces, and so one could hope for better solutions. Indeed, the next result exploits this idea:

Theorem 6 ([7]). *There exists a truthful and polynomial-time deterministic multi-unit CA, for $B \geq 3$ copies of each item, that obtains $O(B \cdot m^{\frac{1}{B-2}})$-approximation to the optimal social welfare.*

This auction copes with the representation issue (since general valuations are assumed) by accessing the valuations through a "demand oracle": given per-item prices $\{p_x\}_{x\in\Omega}$, specify a bundle S that maximizes $v_i(S) - \sum_{x\in S} p_x$.

Two main drawbacks of this auction motivate further research on the issue. First, as B gets larger it is reasonable to expect the approximation to approach 1 (indeed polynomial-time algorithms with such an approximation guarantee do exist). However here the approximation ratio does not decrease below $O(\log m)$ (this ratio is achieved for $B = O(\log m)$). Second, this auction does not provide a solution to the original setting, where $B = 1$, and, in general for small B's the approximation factor is rather high. One way to cope with these problems is to introduce randomness:

Theorem 7 ([26]). *There exists a truthful-in-expectation and polynomial-time randomized multi-unit CA, for any $B \geq 1$ copies of each item, that obtains $O(m^{\frac{1}{B+1}})$-approximation to the optimal social welfare.*

Thus, by allowing randomness, the gap from the standard computational status is being completely closed. The definition of truthfulness-in-expectation is the natural extension of truthfulness to a randomized environment: the *expected* utility of a player is maximized by being truthful.

However, this notion is strictly weaker than the deterministic notion, as this implicitly implies that players care only about the expectation of their utility (and not, for example, about the variance). This is termed "the risk-neutrality" assumption in the economics literature. An intermediate notion for randomized mechanisms is that of "universal truthfulness": the mechanism is truthful given any fixed result of the coin toss. Here, risk-neutrality is no longer needed. [15] give a universally truthful CA for $B = 1$ that obtains an $O(\sqrt{m})$-approximation. Universally truthful mechanisms are still weaker than deterministic truthful mechanisms, due to two reasons: (i) It is not clear how to actually create the correct and exact probability distribution with a deterministic computer. The situation here is different than in "regular" algorithmic settings, where various derandomization techniques can be employed, since these in general does not carry through the truthfulness property. (ii) Even if a natural randomness source exists, one cannot improve the quality of the actual output by repeating the computation several times (using the the law of large numbers). Such a repetition will again destroy truthfulness. Thus, exactly because the game-theoretic issues are being considered in parallel to the computational ones, the importance of determinism increases.

Open question 3. *What is the best-possible approximation ratio that deterministic and truthful combinatorial auctions can obtain, in polynomial-time?*

There are many valuation classes, that restrict the possible valuations to some reasonable format (see [28] for more details). For example, sub-additive valuations are such that, for any two bundles $S, T, \subseteq \Omega$, $v(S \cup T) \leq v(S) + v(T)$. Such classes exhibit much better approximation guarantees, e.g. for sub-additive valuation a polynomial-time 2-approximation is known [16]. However, no polynomial-time truthful mechanism (be it randomized, or deterministic) with a constant approximation ratio, is known for any of these classes.

Open question 4. *Does there exist polynomial-time truthful constant-factor approximations for special cases of CAs that are NP-hard?*

Revenue maximization in CAs is of-course another important goal. This topic is still mostly unexplored, with few exceptions. The mechanism [7] obtains the same guarantees with respect to the optimal revenue. Improved approximations exist for multi-unit auctions (where all items are identical) with budget constrained players [12], and for unlimited-supply CAs with single-minded bidders [6].

The topic of Combinatorial Auctions is discussed also in the entry on Multiple Item Auctions.

2.4 Problem Domain 4: online auctions.

In the classic CS setting of "online computation", the input to an algorithm is not revealed all at once, before the computation begins, but gradually, over time (for a detailed discussion see the many entries on online problems in this book). This structure suits the auction world, especially in the new electronic environments. What happens when players arrive over time, and the auctioneer must make decisions facing only a subset of the players at any given time?

The integration of online settings, worst-case analysis, and auction theory, was suggested by [24]. They considered the case where players arrive one at a time, and the auctioneer must provide an answer to each player *as it arrives*, without knowing the future bids. There are k identical items, and each bidder may have a distinct value for every possible quantity of the item. These values are assumed to be marginally decreasing, where each marginal value lies in the interval $[\underline{v}, \bar{v}]$. The private information of a bidder includes both her valuation function, and her arrival time, and so a truthful auction need to incentivize the players to arrive on time (and not later on), and to reveal their true values. The most interesting result in this setting is for a large k, so that in fact there is a continuum of items:

Theorem 8 ([24])**.** *There exists a truthful online auction that simultaneously approximates, within a factor of $O(\log(\bar{v}/\underline{v}))$, the optimal offline welfare, and the offline revenue of VCG. Furthermore, no truthful online auction can obtain a better approximation ratio to either one of these criteria (separately).*

This auction has the interesting property of being a "posted price" auction. Each bidder is not required to reveal his valuation function, but, rather, he is given a price for each possible quantity, and then simply reports the desired quantity under these prices.

Ideas from this construction were later used by [10] to construct *two*-sided online auction markets, where multiple sellers and buyers arrive online.

This approximation ratio can be dramatically improved, to be a constant, 4, if one assumes that (i) there is only one item, and (ii) player values are i.i.d from some fixed distribution. No a-priori knowledge of this distribution is needed, as neither the mechanism nor the players are required to make any use of it. This work, [19], analyzes this by making an interesting connection to the class of "secretary problems".

A general method to convert online algorithms to online mechanisms is given by [4]. This is done for one item auctions, and, more generally, for one parameter domains. This method is competitive both with respect to the welfare and the revenue.

The revenue that the online auction of theorem 8 manages to raise is competitive only with respect to VCG's revenue, which may be far from optimal. A parallel line of works is concerned with revenue maximizing auctions. To achieve good results, two assumptions need to be made: (i) there exists an unlimited supply of items (and recall from section 2.2 that $F(v)$ is the offline optimal monopolistic fixed-price revenue), and (ii) players cannot lie about their arrival time, only about their value. This last assumption is very strong, but apparently needed. Such auctions are termed here "value-truthful", indicating that "time-truthfulness" is missing.

Theorem 9 ([9])**.** *For any $\epsilon > 0$, there exists a value-truthful online auction, for the unlimited supply case, with expected revenue of at least $\frac{F(v)}{1+\epsilon} - O(h/\epsilon^2)$.*

The construction exploits principles from learning theory in an elegant way. Posted price auctions for this case are also possible, in which case the additive loss increases to $O(h \log\log h)$. [19] consider fully-truthful online auctions for revenue maximization, but manage to obtain only very high (although fixed) competitive ratios. Constructing fully-truthful online auctions with a close-to-optimal revenue remains an open question. Another interesting open question involves multi-dimensional valuations. The work [24] remains the only work for players that may demand multiple items. However their competitive guarantees are quite high, and achieving better approximation guarantees (especially with respect to the revenue) is a challenging task.

2.5 Advanced Issues

Monotonicity. What is the general way for designing a truthful mechanism? The straightforward way is to check, for a given social choice function f, whether truthful prices exist. If not, try to "fix" f. It turns out, however, that there exists a more structured way, an *algorithmic* condition that will imply the *existence* of truthful prices. Such a condition shifts the designer back to the familiar territory of algorithmic design. Luckily, such a condition do exist, and is best described in the abstract social choice setting of section 1:

Definition 3 ([23], [8])**.** *A social choice function $f : V \to A$ is "weakly monotone" (W-MON) if for any i, $v_{-i} \in V_{-i}$, and any $v_i, v'_i \in V_i$, the following holds. Suppose that $f(v_i, v_{-i}) = a$, and $f(v'_i, v_{-i}) = b$. Then $v'_i(b) - v_i(b) \geq v'_i(a) - v_i(a)$.*

In words, this condition states the following. Suppose that player i changes her declaration from v_i to v'_i, and this causes the social choice to change from a to b. Then it must be the case that i's value for b has increased in the transition from v_i to v'_i no-less than i's value for a.

Theorem 10 ([35])**.** *Fix a social choice function $f : V \to A$, where V is convex, and A is finite. Then there exist prices p such that $M = (f, p)$ is truthful if and only if f is weakly monotone.*

Furthermore, given a weakly monotone f, there exists an explicit way to determine the appropriate prices p (see [18] for details).

Thus, the designer should aim for weakly monotone algorithms, and need not worry about actual prices. But how difficult is this? For single-dimensional domains, it turns out that W-MON leaves ample flexibility for the algorithm designer. Consider for example the case where every alternative has a value of either 0 (the player "loses") or some $v_i \in \Re$ (the player "wins" and obtains a value v_i). In such a case, it is not hard to show that W-MON reduces to the following monotonicity condition: if a player wins with v_i, and increases her value to $v'_i > v_i$ (while v_{-i} remains fixed), then she must win with v'_i as well. Furthermore, in such a case, the price of a winning player must be set to the infimum over all winning values.

Impossibilities of truthful design. It is fairly simple to construct algorithms that satisfy W-MON for single-dimensional domains, and a variety of positive results were obtained for such domains, in classic mechanism design, as well as in algorithmic mechanism design. But how hard is it to satisfy W-MON for multi-dimensional domains? This question is yet unclear, and seems to be one of the challenges of algorithmic mechanism design. The contrast between single-dimensionality and multi-dimensionality appears in all problem domains that were surveyed here, and seems to reflect some inherent difficulty that is not exactly understood yet. Given a social choice function f, call f *implementable* (in dominant strategies) if there exist prices p such that $M = (f, p)$ is truthful. The basic question is then *what forms of social choice functions are implementable.*

As detailed in the beginning, the welfare maximizing social choice function is implementable. This specific function can be slightly generalized to allow weights, in the following way: fix some non-negative real constants $\{w_i\}_{i=1}^n$ (not all are zero) and $\{\gamma_a\}_{a \in A}$, and choose an alternative that maximizes the *weighted* social welfare, i.e. $f(v) \in \text{argmax}_{a \in A} \sum_i w_i v_i(a) + \gamma_a$. This class of functions is sometimes termed "affine maximizers". It turns out that these functions are also implementable, with prices similar in spirit to VCG. In the context of the above characterization question, one sharp result stands out:

Theorem 11 ([34])**.** *Fix a social choice function $f : V \rightarrow A$, such that (i) A is finite, $|A| \geq 3$, and f is onto A, and (ii) $V_i = \Re^A$ for all i. Then f is implementable (in dominant strategies) if and only if it is an affine maximizer.*

The domain V that satisfies $V_i = \Re^A$ for all i is term an "unrestricted domain". The theorem states that, if the domain is unrestricted, at least three alternatives are chosen, and the set A of alternatives is finite, then nothing besides affine maximizers can be implemented!

However, the assumption that the domain is unrestricted is very restrictive. All the above example domains exhibit some basic combinatorial structure, and are therefore restricted in some way. And as discussed above, for many restricted domains the theorem is simply not true. So what *is* the possibilities – impossibilities border? As mentioned above, this is an unsolved challenge. Lavi, Mu'alem, and Nisan [23] explore this question for Combinatorial Auctions and similar restricted domains, and reach partial answers. For example:

Theorem 12 ([23])**.** *Any truthful combinatorial auction or multi-unit auction among two players, that must always allocate all items, and that approximates the welfare by a factor better than 2, must be an affine maximizer.*

Of-course, this is far from being a complete answer. What happens if there are more than two players? And what happens if it is possible to "throw away" part of the items? These questions, and the more general and abstract characterization question, are all still open.

Alternative solution concepts. In light of the conclusions of the previous section, a natural thought would be to re-examine the *solution concept* that is being used. Truthfulness relies on the strong concept of dominant strategies: for each player there is a unique strategy that maximizes her utility, no matter what the other players are doing. This is very strong, but it fits very well the worst-case way of thinking in CS. What other solution concepts can be used? As described above, randomization, and truthfulness-in-expectation, can help. A related concept, again for randomized mechanisms, is truthfulness with high probability. Another direction is to consider mechanisms where players cannot improve their utility *too much* by deviating from the truth-telling strategy [21].

Algorithm designers do not care so much about actually reaching an equilibrium point, or finding out what will the players play – the major concern is to guarantee the optimality of the solution, taking into account the strategic behavior of the players. Indeed, one way of doing this is to guarantee a good equilibrium point. But there is no reason to rule out mechanisms where several acceptable strategic choices for the players exist, provided that the approximation will be achieved *in each of these choices.*

As a first attempt, one is tempted to simply let the players try and improve the basic result by allowing them to lie. However, this can cause unexpected dynamics, as each player chooses her lies under some assumptions about the lies of the others, etc. etc. To avoid such an unpredictable situation, it is important to insist on using rigorous game theoretic reasoning to explain exactly why the outcome will be satisfactory.

The work [31] suggests the notion of "feasibly dominant" strategies, where players reveal the possible lies they consider, and the mechanism takes this into account. By assuming that the players are computationally bounded, one can show that, instead of actually "lying", the players will prefer to reveal their true types plus all the lies they might consider. In such a case, since the mechanism has obtained the true types of the players, a close-to-optimal outcome will be guaranteed.

Another definition tries to capture the initial intuition by using the classic game-theoretic notion of undominated strategies:

Definition 4 ([5]). *A mechanism M is an "algorithmic implementation of a c-approximation (in undominated strategies)" if there exists a set of strategies, D, such that (i) M obtains a c-approximation for any combination of strategies from D, in polynomial time, and (ii) For any strategy not in D, there exists a strategy in D that weakly dominates it, and this transition is polynomial-time computable.*

By the second condition, it is reasonable to assume that a player will indeed play *some* strategy in D, and, by the first condition, it does not matter what tuple of strategies in D will actually be chosen, as any of these will provide the approximation. This transfers some of the burden from the game-theoretic design to the algorithmic design, since now a guarantee on the approximation should bu provided for a larger range of strategies. [5] exploit this notion to design a deterministic CA for multi-dimensional players that achieves a close-to-optimal approximation guarantee. A similar-in-spirit notion, although a weaker one, is the notion of "Set-Nash" [25].

3 APPLICATIONS

One of the popular examples to a "real-life" combinatorial auction is the spectrum auction that the US government conducts, in order to sell spectrum licenses. Typical bids reflect values for different spectrum ranges, to accommodate different geographical and physical needs, where different spectrum ranges may complement or substitute one another. The US government invests research efforts in order to determine the best format for such an auction, and auction theory is heavily exploited. Interestingly, the US law guides the authorities to allocate these spectrum ranges in a way that will maximize *the social welfare*, thus providing a good example for the usefulness of this goal.

Adword auctions are another new and fast-growing application of auction theory in general, and of the new algorithmic auctions in particular. These are auctions that determine the advertisements that web-search engines place close to the search results they show, after the user submits her search keywords. The interested companies compete, for every given keyword, on the right to place their ad on the results' page, and this turns out to be the main source of income for companies like Google. Several entries in this book touch on this topic in more details, including the entries on Adwords Pricing and on Position Auctions.

A third example to a possible application, in the meanwhile implemented only in the academic research labs, is the application of algorithmic mechanism design to pricing and congestion control in communication networks. The existing fixed pricing scheme has many disadvantages, both with respect to the needs of efficiently allocating the available resources, and with respect to the new opportunities of the Internet companies to raise more revenue due to specific types of traffic. Theory suggests solutions to both of these problems.

4 CROSS REFERENCES

Adword Pricing, Competitive Auctions, False Name Proof Auctions, Generalized Vickrey Auction, Incentive Compatible Ranking, Mechanisms for One Parameter Agents Single Buyer/Seller, Multiple Item Auctions, Position Auctions, Truthful Multicast.

5 RECOMMENDED READING

The topics presented here are detailed in the textbook [33]. Section 1 is based on the paper [32], that also coined the term "algorithmic mechanism design". The book [14] covers the various aspects of combinatorial auctions.

[1] G. Aggarwal, A. Fiat, A. Goldberg, N. Immorlica, and M. Sudan. Derandomization of auctions. In *Proc. of the 37th ACM Symposium on Theory of Computing (STOC'05)*, 2005.

[2] N. Andelman, Y. Azar, and M. Sorani. Truthful approximation mechanisms for scheduling selfish related machines. In *Proc. of the 22nd International Symposium on Theoretical Aspects of Computer Science (STACS)*, pages 69–82, 2005.

[3] A. Archer and É. Tardos. Truthful mechanisms for one-parameter agents. In *Proc. 42nd Annual Symposium on Foundations of Computer Science (FOCS)*, pages 482–491, 2001.

[4] B. Awerbuch, Y. Azar, and A. Meyerson. Reducing truth-telling online mechanisms to online optimization. In *Proc. of the 35th ACM Symposium on Theory of Computing (STOC'03)*, 2003.

[5] M. Babaioff, R. Lavi, and E. Pavlov. Single-value combinatorial auctions and implementation in undominated strategies. In *Proc. of the 17th Symposium on Discrete Algorithms (SODA)*, 2006.

[6] M. Balcan, A. Blum, J. Hartline, and Y. Mansour. Mechanism design via machine learning. In *Proc. of the 46th Annual Symposium on Foundations of Computer Science (FOCS'05)*, 2005.

[7] Y. Bartal, R. Gonen, and N. Nisan. Incentive compatible multi-unit combinatorial auctions. In *Proc. of the 9th Conference on Theoretical Aspects of Rationality and Knowledge (TARK'03)*, 2003.

[8] S. Bikhchandani, S. Chatterjee, R. Lavi, A. Mu'alem, N. Nisan, and A. Sen. Weak monotonicity characterizes deterministic dominant-strategy implementation. *Econometrica*, 74:1109–1132, 2006.

[9] A. Blum and J. Hartline. Near-optimal online auctions. In *Proc. of the 16th Symposium on Discrete Algorithms (SODA)*, 2005.

[10] A. Blum, T. Sandholm, and M. Zinkevich. Online algorithms for market clearing. *Journal of the ACM*, 53(5):845–879, 2006.

[11] L. Blumrosen and N. Nisan. On the computational power of iterative auctions. In *Proc. of the 7th ACM Conference on Electronic Commerce (EC'05)*, 2005.

[12] C. Borgs, J. Chayes, N. Immorlica, M. Mahdian, and A. Saberi. Multi-unit auctions with budget-constrained bidders. In *Proc. of the 7th ACM Conference on Electronic Commerce (EC'05)*, 2005.

[13] G. Christodoulou, E. Koutsoupias, and A. Vidali. A lower bound for scheduling mechanisms. In *Proc. 18th Symposium on Discrete Algorithms (SODA)*, 2007.

[14] P. Cramton, Y. Shoham, and R. Steinberg. *Combinatorial Auctions.* MIT Press, 2005.

[15] S. Dobzinski, N. Nisan, and M. Schapira. Truthful randomized mechanisms for combinatorial auctions. In *Proc. of the 38th ACM Symposium on Theory of Computing (STOC'06)*, 2006.

[16] U. Feige. On maximizing welfare when utility functions are subadditive. In *Proc. of the 38th ACM Symposium on Theory of Computing (STOC'06)*, 2006.

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

[18] H. Gui, R. Muller, and R. V. Vohra. Characterizing dominant strategy mechanisms with multi-dimensional types, 2004. Working paper.

[19] M. Hajiaghayi, R. Kleinberg, and D. Parkes. Adaptive limited-supply online auctions. In *Proc. of the 6th ACM Conference on Electronic Commerce (EC'04)*, 2004.

[20] J. Hartline and R. McGrew. From optimal limited to unlimited supply auctions. In *Proc. of the 7th ACM Conference on Electronic Commerce (EC'05)*, 2005.

[21] A. Kothari, D. Parkes, and S. Suri. Approximately-strategyproof and tractable multi-unit auctions. *Decision Support Systems*, 39:105–121, 2005.

[22] A. Kovács. Fast monotone 3-approximation algorithm for scheduling related machines. In *Proc. 13th Annual European Symposium on Algorithms (ESA)*, pages 616–627, 2005.

[23] R. Lavi, A. Mu'alem, and N. Nisan. Towards a characterization of truthful combinatorial auctions. In *Proc. of the 44rd Annual Symposium on Foundations of Computer Science (FOCS'03)*, 2003.

[24] R. Lavi and N. Nisan. Competitive analysis of incentive compatible on-line auctions. *Theoretical Computer Science*, 310:159–180, 2004.

[25] R. Lavi and N. Nisan. Online ascending auctions for gradually expiring items. In *Proc. of the 16th Symposium on Discrete Algorithms (SODA)*, 2005.

[26] R. Lavi and C. Swamy. Truthful and near-optimal mechanism design via linear programming. In *Proc. 46th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 595–604, 2005.

[27] R. Lavi and C. Swamy. Truthful mechanism design for multi-dimensional scheduling via cycle monotonicity, 2007. Working paper.

[28] B. Lehmann, D. Lehmann, and N. Nisan. Combinatorial auctions with decreasing marginal utilities. *Games and Economic Behavior*, 55(2):270–296, 2006.

[29] D. Lehmann, L. O'Callaghan, and Y. Shoham. Truth revelation in approximately efficient combinatorial auctions. *Journal of the ACM*, 49(5):577–602, 2002.

[30] A. Mu'alem and M. Schapira. Setting lower bounds on truthfulness. In *Proc. 18th Symposium on Discrete Algorithms (SODA)*, 2007.

[31] N. Nisan and A. Ronen. Computationally feasible vcg mechanisms. In *Proc. of the 2nd ACM Conference on Electronic Commerce (EC'00)*, 2000.

[32] N. Nisan and A. Ronen. Algorithmic mechanism design. *Games and Economic Behavior*, 35:166–196, 2001.

[33] N. Nisan, T. Roughgarden, E. Tardos, and V. Vazirani. *Algorithmic Game Theory.* Cambridge University Press, 2007. (expected to appear).

[34] K. Roberts. The characterization of implementable choice rules. In J.-J. Laffont, editor, *Aggregation and Revelation of Preferences.*, pages 321–349. North-Holland, 1979.

[35] M. Saks and L. Yu. Weak monotonicity suffices for truthfulness on convex domains. In *Proc. 6th ACM Conference on Electronic Commerce (ACM-EC)*, pages 286–293, 2005.