Noname manuscript No.
(will be inserted by the editor)

A lower bound of $1 + \varphi$ for truthful scheduling mechanisms

Elias Koutsoupias · **Angelina Vidali**

Abstract We study the mechanism design version of the unrelated machines scheduling problem, which is at the core of Algorithmic Game Theory and was first proposed and studied in a seminal paper of Nisan and Ronen. We give an improved lower bound of $1 + \varphi \approx 2.618$ on the approximation ratio of deterministic truthful mechanisms for the makespan. The proof is based on a recursive preprocessing argument which yields a strictly increasing series of new lower bounds for each fixed number of machines $n \geq 4$.

1 Introduction

We study the classical scheduling problem on unrelated machines [1–3] from the mechanism-design point of view. There are n machines and m tasks each with different execution time on each machine. The objective of the mechanism is to choose an allocation of the tasks to the machines that minimizes the makespan, i.e. that minimizes the time we have to wait until all tasks have been executed.

Mechanism-design recasts optimization problems, such as the scheduling problem, by adding an additional requirement: besides the objective of the

Supported in part by IST-15964 (AEOLUS), the project PENED 2003 (co-financed 75% of public expenditure through EC European Social Fund, 25% of public expenditure through Ministry of Development General Secretariat of Research and Technology of Greece and through private sector, under measure 8.3 of Operational Program Competitiveness in the 3rd Community Support Program, and the Greek Ministry of Education (Thales)) and the Vienna Science and Technology Fund WWTF grant ICT10-002.

Elias Koutsoupias
Department of Informatics, University of Athens, Email: `elias@di.uoa.gr`

Angelina Vidali
Theory and Applications of Algorithms Research Group, University of Vienna, Austria, Main part of the work done at the University of Athens. Email: `angelina.vidali@univie.ac.at`

algorithm designer (here to minimize the makespan) each one of the individuals involved in the solution of the problem (here each machine) has his own objective, namely to maximize selfishly his utility. Each one of the individuals involved has a private value, and the collection of these private values comprises the input to the problem. In order to compute or approximate the optimal solution, the mechanism designer needs to elicit the true input of the problem from the players. For this, the mechanism designer uses payments with the goal of changing the utility of the players to have no incentive to lie. The most typical example of mechanism design is auctions, where the most common objectives of the algorithm designer is either to allocate items to players in order to maximize the total welfare of the players or to maximize the revenue of the auctioneer.

The mechanism-design version of the scheduling problem, is essentially an auction for allocating jobs. The machines are selfish players and would refuse to process tasks without payment. With the payments, the objective of each player is to minimize the time of its own tasks minus the payment. Each one of the players has a private type, here the vector of execution times on the particular machine. A mechanism is called *truthful* when telling the truth (i.e. revealing his true type) is a dominant strategy for each player: for all declarations of the other players, an optimal strategy of the player is to tell the truth.

A central question in the area of Algorithmic Mechanism Design is to determine the best approximation ratio of mechanisms for scheduling. This question was raised by Nisan and Ronen in their seminal work [4] and remains wide open today. The current work improves the lower bound on the approximation to $1+\varphi \approx 2.618$, where φ is the golden ratio. A preliminary version of this paper appeared in [5], and although at that time, it seemed to be a result which will be soon subsumed by a better bound, no improvement has appeared yet, despite the fact that the problem has received a lot of attention.

A lower bound on the approximation ratio can be of either computational or game-theoretic nature. A lower bound is computational when it is based on some assumption about the computational resources of the algorithm, most commonly that the algorithm is polynomial-time. It is of game-theoretic nature when the source of difficulty is not computational but is imposed by the restrictions of the mechanism framework and more specifically by the truthfulness condition. Our lower bound is entirely game-theoretic: No (truthful) mechanism—including exponential and even recursive algorithms—can achieve approximation ratio better than 2.618.

When we consider the approximation ratio of a mechanism, we ignore the payments and care only about the allocation part of the mechanism. A natural question then arises: Which scheduling (allocation) algorithms are part of truthful mechanisms? There is an elegant and seemingly simple characterization of these mechanisms: Monotone Algorithms. The characterizing property of these algorithms, the Monotonicity Property (Definition 2), implies that, given the allocation of a player for a specific input, if this player becomes slower in the tasks that were not assigned to him, the player will not get any new tasks. Similarly when the player becomes faster in the tasks that he was already assigned, the player can only get more tasks. In a loose sense, the Monotonicity Property is a combination of these two facts and can be expressed very succinctly.

1.1 Our techniques and related work

There are basically two approaches to obtain lower bounds for this problem. The first approach is to provide a good characterization of all possible mechanisms. With an appropriate characterization, it is usually easy to determine the mechanism with the best approximation ratio. This approach however solves a potentially more difficult problem, the problem of characterizing the truthful mechanisms. There are few results concerning the characterization of mechanisms. For the case of two machines, Dobzinski and Sundararajan [6] showed that every finite approximation mechanism is task-independent, while [7] gave a characterization of all (regardless of approximation ratio) decisive truthful mechanisms in terms of affine minimizers and threshold mechanisms. Until now the only example of a new lower bound obtained by a characterization is the lower bound of 2 for instances with two (or more) tasks [7]. (However for instances with 3 or more tasks it is considerably easier to prove the same lower bound [4] by employing the approach we describe next instead of a characterization.)

The second approach is to use an appropriately selected subset of the input instances. Fix one instance and consider its possible allocations (pick an instance with certain symmetries so that the possible allocations are not too many). Then argue how each one of the possible allocations implies *approximation ratio at least r for some other instance* of the selected set. By applying the Monotonicity Property to pairs of instances, we get restrictions that their corresponding allocations have to satisfy in order to guarantee truthfulness. This approach has been followed in [4,8,9] using a finite set of small instances of 2 and 3 machines respectively and no more than 5 tasks. The approach of the current work is also of the second type. We start we a set of tasks and then we a recursive preprocess the instance in order to reduce it to an instance that satisfies certain desired properties.

The scheduling problem on unrelated machines is one of the most fundamental scheduling problems [1,3]. The problem is NP-complete. Lenstra, Shmoys, and Tardos [2] showed that it can be approximated in polynomial time within a factor of 2 but no better than 3/2, unless P=NP.

Nisan and Ronen introduced the mechanism-design version of the problem in the paper that founded the algorithmic theory of Mechanism Design [4,4]. They showed that the well-known VCG mechanism, which is a polynomial-time algorithm and truthful, has approximation ratio n. They conjectured that there is no deterministic mechanism with approximation ratio less than n. They also showed that no mechanism (polynomial-time or not) can achieve approximation ratio better than 2. This was improved to $1+\sqrt{2}$, in [9]. Here we improve it further to $1+\varphi$. Four years after the appearance of the conference version of this paper this is still the best lower bound known. However, for the special class of "anonymous" mechanisms, Ashlagi, Dobizinski and Lavi [10] proved a tight bound n of the approximation ratio.

Nisan and Ronen [4] also gave a randomized truthful mechanism for two players, that achieves an approximation ratio of $7/4$. Mu'alem and Schapira [8] proved a lower bound of $2-\frac{1}{n}$ for any randomized truthful mechanism for n machines and generalized the mechanism in [4] to give a $7n/8$ upper bound. Lu and Yu [11] gave a 1.67-approximation universally truthful randomized algorithm for the case of 2 machines improving it later on [12] to a 1.59-approximation algorithm.

In another direction, [13] showed that no fractional truthful mechanism can achieve an approximation ratio better than $2-1/n$. It also showed that fractional algorithms that treat each task independently cannot do better than $(n+1)/2$ and this bound is tight.

Cohen et al. [14] studied the envy free version of the scheduling problem on unrelated machines. They devise an envy-free poly-time mechanism that approximates the minimal makespan to within a factor of $O(\log m)$ and show a lower bound of $\Omega(\log m/\log\log m)$. This improved the result of Mu'alem [15] who had given an upper bound of $(m+1)/2$, and a lower bound of $2-1/m$. Christodoulou and Kovács [16] characterization of envy free mechanisms for the case of 2 tasks and n players.

Lavi and Swamy [17] considered the special case of the same problem when the processing times have only two possible values low or high, and devised a deterministic 2-approximation truthful mechanism. Very recently, Yu [18] generalized their results constructing a randomized $7(1+\epsilon)$-approximation algorithm for the case when the processing times belong to $[L_j, L_j(1+\epsilon)] \cup [H_j, H_j(1+\epsilon)]$ where $L_j < H_j$ and $\epsilon < 1/16mn$.

Another special case of the problem is the problem on related machines in which there is a single private value (instead of a vector) for every machine, its speed. Myerson [19] gave a characterization of truthful algorithms for this kind of problems (single-parameter problems), in terms of a monotonicity condition. Archer and Tardos [20] found a similar characterization and using it obtained a variant of the optimal algorithm which is truthful (albeit exponential-time). They also gave a polynomial-time randomized 3-approximation mechanism, which was later improved to a 2-approximation, in [21], and to a PTAS by Dhangwatnotai, Dobzinski, Dughmi and Roughgarden [22]. These mechanisms are truthful in expectation. Auletta De Prisco, Penna, and Persiano [23] provided a deterministic, monotone $(4+\epsilon)$ approximation algorithm for the case of constant number of machines m. Andelman, Azar, and Sorani [24] improved this to a FPTAS and additionally gave a 5-approximation algorithm for arbitrary m. Kovács improved the approximation ratio to 3 [25] and to 2.8 [26]. Finally, the definite answer for the related machines problem was given by Christodoulou and Kovács who gave a deterministic PTAS for the problem [27].

Much more work has been done for the more general problem of combinatorial auctions (see [28], Chapter 11). The mechanisms for the scheduling problem and the combinatorial auctions problem may be closely related, as it was suggested in [29] for the case of 2 players.

Saks and Yu [30] proved that, for mechanism design problems with convex domains of finitely many outcomes, which includes the scheduling problem, the monotonicity property is also sufficient for truthfulness, generalizing results of [31,32]. Monderer [33] showed that this result cannot be essentially extended to a larger class of domains. Both these results concern domains of finitely many outcomes. There are however cases, like the fractional version of the scheduling problem, when the set of all possible allocations is infinite. For these, Archer and Kleinberg [34] provided a necessary and sufficient condition for truthfulness which generalizes the results of [30]. A geometrical characterization of truthfulness for the case of three items was given in [35].

2 Problem definition

We recall here the definitions of the scheduling problem, of the concept of mechanisms, as well as some of their fundamental properties.

Definition 1 (The unrelated machines scheduling problem) *The input to the scheduling problem is a nonnegative matrix t of n rows, one for each machine-player, and m columns, one for each task. The entry t_{ij} (of the i-th row and j-th column) is the time it takes for machine i to execute task j. Let t_i denote the times for machine i, which is the vector of the i-th row. The output is an allocation $x = x(t)$, which partitions the tasks into the n machines. We describe the partition using indicator values $x_{ij} \in \{0, 1\}$: $x_{ij} = 1$ iff task j is allocated to machine i. We should allocate each task to exactly one machine, or more formally $\sum_{i=1}^{n} x_{ij} = 1$.*

In the mechanism-design version of the problem we consider direct-revelation mechanisms. That is, we consider mechanisms that work according to the following protocol:

- Each player i declares the values in row t_i, which is known only to player i.
- The mechanism, based on the declared values, decides how to allocate the tasks to the players.
- The mechanism, based on the declared values, and the allocation of the previous step, decides how much to pay each player.

The mechanism consists of two algorithms, an allocation algorithm and a payment algorithm. The cost of a player (machine) is the sum of the times of the tasks allocated to it minus the payment. Think of the players as being lazy and not wanting to execute tasks, and the mechanism pays them enough to induce them to execute the tasks. On the other hand, the players know both the allocation and the payment algorithm and may have an incentive to lie in the first step. The class of mechanisms for which the players have no incentive to lie are called truthful mechanisms. Here we consider the strictest version of truthfulness which is the class of dominant truthful mechanisms: In these mechanisms truth telling is a dominant strategy, i.e., for every possible declaration of the other players, an optimal strategy of a player is to reveal its true values.

A classical result in mechanism design, the Revelation Principle (see [28] page 224), states that for every mechanism, in which each player has a dominant strategy, there is a truthful mechanism which achieves the same objective. The Revelation Principle allows us to concentrate on truthful mechanisms (at least for the class of centralized mechanisms). Since every mechanism with dominant strategies can be turned into an equivalent truthful one, we can concentrate only on truthful mechanisms.

Here we care only about the approximation ratio of the allocation part of the mechanisms. So when we refer to the approximation ratio of a mechanism, we mean the approximation ratio of its allocation part. Since payments are of no importance in this consideration, it would be helpful if we could find a necessary and sufficient condition that characterizes which allocation algorithms are ingredients of truthful mechanisms. Fortunately such a condition exists:

Definition 2 (Monotonicity Property) *An allocation algorithm is called monotone if it satisfies the following property: for every two sets of tasks t and t', which differ only on some machine i (i.e., on the i-th row), the associated allocations x and x' satisfy $\sum_{j=1}^{m}(x_{ij}-x'_{ij})(t_{ij}-t'_{ij}) \leq 0$, which can be written more succinctly as a dot product:*

$$(x_i - x'_i) \cdot (t_i - t'_i) \leq 0.$$

Proposition 1 *Every truthful mechanism satisfies the Monotonicity Property.*

The Monotonicity Property characterizes the allocation part of truthful mechanisms. The fact that it is necessary and sufficient was shown in [4] and [30] respectively. Although this is a complete characterization, it is not easy to use it, because it is a local property for each player separately, and because it involves two inputs. One fundamental open problem is to find a better characterization of truthful mechanisms for the scheduling problem. For the problem of mechanism design in unrestricted domains (i.e. when the possible valuations are unrestricted), there is a simple characterization by Roberts [36]: The only truthful mechanisms are affine minimizers [37–39]. In the scheduling problem the valuations are restricted to additive valuations, so the scheduling problem is at the other end of the spectrum, where the domain is restricted yet general enough to admit interesting mechanisms.

Lacking such a nice characterization as the characterization by Roberts for the domain of the scheduling problem, we employ the Monotonicity Property in order to argue about how the allocation of one instance of the problem affects the allocation of another instance. In particular, the following lemma from [9], which will be the main ingredient of our proof, gives a way to change the values of one player without changing his allocation. For completeness, we also include its proof here.

Lemma 1 *Let t be a matrix of processing times and let $x = x(t)$ be the allocation produced by a truthful mechanism.*

a. *Suppose that we change only the processing times of machine i and in such a way that $t'_{ij} > t_{ij}$ when $x_{ij} = 0$, and $t'_{ij} < t_{ij}$ when $x_{ij} = 1$. The mechanism does not change the allocation to machine i, i.e., $x_i(t') = x_i(t)$. (However, it may change the allocation of other machines).*
b. *Fix now a mechanism with approximation ratio r and consider an instance whose optimal allocation has cost c. Suppose that for a task j we have $t_{ij} = 0$ for machine i, and $t_{i'j} = \infty$ for every other machine i', where ∞ denotes a very large real number, greater than $r \cdot (c+u)$, for some constant u. If we change the times of machine i for all other tasks as in the first part of the lemma but raise the time for task j to $t'_{ij} = u$, the mechanism again does not change the allocation vector of machine i.*

Proof By the Monotonicity Property, we have

$$\sum_{j=1}^{m}(t_{ij} - t'_{ij})(x_{ij}(t) - x_{ij}(t')) \leq 0.$$

a. Observe that all terms of the sum are nonnegative (by the premises of the lemma). The only way to satisfy the inequality is to have all terms equal to 0, that is, $x_{ij}(t) = x_{ij}(t')$.
b. When we change the value t_{ij} to u, the optimum makespan becomes at most $c+u$. If the mechanism allocates task j to a machine different than i, the approximation ratio is greater than r, which contradicts the hypothesis about the mechanism. Therefore $x_{ij}(t) = x_{ij}(t') = 1$, which makes the term corresponding to task j in the sum $\sum_{j=1}^{m}(t_{ij} - t'_{ij})(x_{ij}(t) - x_{ij}(t')) \leq 0$ vanish. For the rest of the terms we repeat the argument of the first part.

Remark 1 To simplify the presentation, when we apply Lemma 1, we will increase or decrease only some values of a machine, not all its values. The understanding will be that *the rest of the values increase or decrease appropriately by a tiny amount which we omit* to keep the expressions simple.

3 A lower bound of $1 + \varphi$ for $n \to \infty$ machines

The main result of this work is

Theorem 1 *There is no deterministic mechanism for the scheduling problem with $n \to \infty$ machines with approximation ratio less than $1 + \varphi$.*

Moreover for any fixed number of players n, the solution of the equation

$$1 + \frac{1}{a} + \frac{1}{a^2} + \ldots + \frac{1}{a^{n-1}} = 1 + a.$$

n	2	3	4	5	6	7	8	...	∞
Lower bound	2	2.324	2.465	2.534	2.570	2.590	2.601	...	2.618

Table 1 The lower bound given by Theorem 1 for few machines.

is a lower bound for the approximation ratio (Table 3.)

We shall build the proof of the theorem around the instance

$$\begin{pmatrix} 0 & \infty & \cdots & \infty & \infty & 1 & a & \cdots & a^{n-2} \\ \infty & 0 & \cdots & \infty & \infty & a & a^2 & \cdots & a^{n-1} \\ & & \ddots & & & & & & \\ \infty & \infty & \cdots & 0 & \infty & a^{n-2} & a^{n-1} & \cdots & a^{2n-4} \\ \infty & \infty & \cdots & \infty & 0 & a^{n-1} & a^{n} & \cdots & a^{2n-3} \end{pmatrix},$$

where $a \geq 1$ is a parameter and ∞ denotes an arbitrarily high value. Eventually, we will set $a = \varphi$ when $n \to \infty$. We let however a to be a parameter for clarity and for obtaining better bounds for small n.

The lower bound will follow from the fact (which we will eventually prove) that every truthful mechanism with approximation ratio less than $1 + a$ must allocate all $n - 1$ rightmost tasks to the first player. However, in order to be able to prove the statement for arbitrary n we need to prove a stronger statement which involves instances of the form

$$T(i_1, \ldots, i_k) = \begin{pmatrix} 0 & \infty & \cdots & \infty & a^{i_1} & a^{i_2} & \cdots & a^{i_k} \\ \infty & 0 & \cdots & \infty & a^{i_1+1} & a^{i_2+1} & \cdots & a^{i_k+1} \\ \vdots & & \ddots & & \vdots & & \ddots & \vdots \\ \infty & \infty & \cdots & 0 & a^{i_1+n-1} & a^{i_2+n-1} & \cdots & a^{i_k+n-1} \end{pmatrix},$$

where $0 \leq i_1 < i_2 < \ldots < i_k$ are natural numbers and $k \leq n - 1$. We call the tasks indexed from 1 to k *regular*. We allow these instances to have additional tasks for which some value is 0, i.e., additional columns with at least one 0 entry in each one and we call these tasks *redundant*. This is only for technical reasons, and will play no significant role in the proof (and it definitely does not affect the optimal cost).

We will call the first n tasks *dummy*. Observe that every mechanism with bounded approximation ratio must allocate the i-th dummy task to player i.

Remark 2 Notice that the optimal allocation for $T(i_1, \ldots, i_k)$ has cost a^{i_k}. Furthermore, if $i_1, i_2, \ldots, i_k$ are all successive natural numbers, then the optimal allocation is unique and coincides with the diagonal assignment and has cost a^{i_k}. Also notice that in this case all values of the last $n - k$ players are at least a^{i_k+1}. For example, the following instance has a unique optimal allocation indicated by stars. We will employ this convention of using stars to indicate an allocation throughout.

$$\begin{pmatrix} 0* & \infty & \infty & \infty & \infty & 1 & a & a^2* \\ \infty & 0* & \infty & \infty & \infty & a & a^2* & a^3 \\ \infty & \infty & 0* & \infty & \infty & a^2* & a^3 & a^4 \\ \infty & \infty & \infty & 0* & \infty & a^3 & a^4 & a^5 \\ \infty & \infty & \infty & \infty & 0* & a^4 & a^5 & a^6 \end{pmatrix}$$

Otherwise there are more than one allocations with optimal cost. For example the allocations below:

$$\begin{pmatrix} 0* & \infty & \infty & \infty & \infty & 1 & a & a^3* \\ \infty & 0* & \infty & \infty & \infty & a & a^2* & a^4 \\ \infty & \infty & 0* & \infty & \infty & a^2* & a^3 & a^5 \\ \infty & \infty & \infty & 0* & \infty & a^3 & a^4 & a^6 \\ \infty & \infty & \infty & \infty & 0* & a^4 & a^5 & a^7 \end{pmatrix}, \quad \begin{pmatrix} 0* & \infty & \infty & \infty & \infty & 1 & a & a^3* \\ \infty & 0* & \infty & \infty & \infty & a & a^2 & a^4 \\ \infty & \infty & 0* & \infty & \infty & a^2 & a^3* & a^5 \\ \infty & \infty & \infty & 0* & \infty & a^3* & a^4 & a^6 \\ \infty & \infty & \infty & \infty & 0* & a^4 & a^5 & a^7 \end{pmatrix}$$

both have the optimal cost a^3.

We will now show the main technical lemma of the proof.

Lemma 2 *Suppose that a truthful mechanism on $T(i_1, \ldots, i_k)$, does not allocate all non-dummy tasks to the first player. Then we can find another instance for which the approximation ratio is at least $1 + a$.*

Proof Fix a truthful mechanism and suppose that the first player does not get all regular tasks. In the first part we do a preprocessing of $T(i_1, \ldots, i_k)$. We recursively manipulate the tasks in such a way that we obtain a smaller instance $T(i'_1, \ldots, i'_{k'})$ with $1 \leq k' \leq k$ whose allocation satisfies the following properties:

– the first player gets no regular task, and
– every other player gets at most one regular task.

In the second part, we show that instances which satisfy the above two conditions, can be changed to obtain an instance with approximation ratio at least $1 + a$.

1st part: Suppose that the first of the above conditions is not satisfied. That is, suppose that the first player gets some regular task. We can then decrease its value (for the first player) to 0. By the Monotonicity Property and in particular by Lemma 1 (keep also in mind Remark 1), the same set of tasks will be allocated to the first player, so he still does not get all non-redundant tasks.

Suppose that the second condition is not satisfied, i.e., there is a player in $\{2, \ldots, n\}$ who gets at least two tasks. We can then lower all the non-zero values allocated to this player to 0 except for one. By the Monotonicity Property and in particular by Lemma 1, the same tasks will be allocated to the player. This guarantees that the first player still does not get all non-dummy tasks. Whenever we change the value of a task 0 it becomes redundant.

Redundant tasks remain part of the instance but they will play no particular role in the proof.

By repeating the above operations we decrease the number of regular tasks. We will end up with an instance that contains at least one regular task in which the first player gets no regular task and every other player gets at most one regular task.

2nd part: We can now assume that there is some $T(i_1, \dots, i_k)$ with $k \geq 1$ for which the above two conditions are satisfied, i.e, the mechanism allocates no regular task to the first player and at most one regular task to each of the other players. For clarity, we can assume that there are no redundant tasks; they play no essential role in the rest of the argument because they do not affect the cost of the optimal solution and they can only increase the cost of the mechanism. Specifically, there is only one place in the argument where the redundant tasks may affect the cost of the mechanism and we point it out below.

The optimum cost is a^{i_k}. Our aim is to find a regular task, which is allocated to some player j, with value at least a^{i_k+1}; we will then increase player j's dummy 0 value to a^{i_k}. By Lemma 1, player j will get both tasks with total value at least $a^{i_k+1} + a^{i_k}$. This is the only place where the redundant tasks may play a role. Specifically, when we increase the value of the dummy task, some redundant tasks that were allocated to player j may move to another player. However, the cost of the mechanism is again at least $a^{i_k+1} + a^{i_k}$.

If the optimum cost is still a^{i_k}, then the approximation ratio is at least $1+a$. However, when we raise the dummy 0 to a^{i_k} we may increase the optimum cost. The crux of the proof is that there is always an allocated value greater or equal to a^{i_k+1} for which this bad case does not occur. To find such a value we consider two cases:

Case 1: The algorithm assigns a task with value at least a^{i_k+1} to one of the last $n-k$ players. This is the easy case, because we can increase the dummy 0 value of this player to a^{i_k} without affecting the optimum. The reason is that we can allocate the non-dummy tasks to the first k players with cost a^{i_k} (see Remark 2).

Example 1 Consider the following instance with $n = 5$ and $k = 3$. Suppose that the mechanism has the allocation indicated by the stars.

$$\begin{pmatrix} 0* & \infty & \infty & \infty & \infty & 1 & a & a^3 \\ \infty & 0* & \infty & \infty & \infty & a & a^2 & a^4* \\ \infty & \infty & 0* & \infty & \infty & a^2* & a^3 & a^5 \\ \infty & \infty & \infty & 0* & \infty & a^3 & a^4* & a^6 \\ \infty & \infty & \infty & \infty & 0* & a^4 & a^5 & a^7 \end{pmatrix}$$

Then we can raise the dummy 0 of the 4-th player to a^3. This does not affect the optimum (which is a^3) but raises the cost of the 4-th player to $a^4 + a^3$.

$$\begin{pmatrix} 0* & \infty & \infty & \infty & \infty & 1 & a & a^3 \\ \infty & 0* & \infty & \infty & \infty & a & a^2 & a^4 \\ \infty & \infty & 0* & \infty & \infty & a^2 & a^3 & a^5 \\ \infty & \infty & \infty & a^3* & \infty & a^3 & a^4* & a^6 \\ \infty & \infty & \infty & \infty & 0* & a^4 & a^5 & a^7 \end{pmatrix}$$

The allocation of the 3-rd player is indicated by the stars, the rest of the players might exchange their non-dummy tasks but it doesn't affect our argument.

Case 2: The value of all tasks assigned to the last $n - k$ players is at most a^{i_k}. Consequently the indices $i_1, i_2, \ldots, i_k$ are not successive integers (see Remark 2). Let q be the length of the last block of successive indices, i.e., $k - q$ is the maximum index where there is a gap in the sequence $i_1, i_2, \ldots, i_k$. More precisely, let $k - q$ be the maximum index for which $i_{k-q} + 1 < i_{k-q+1}$. Since player 1 gets no non-dummy task, there is a player $p \in \{q + 1, \ldots, n\}$ such that some of the last q tasks is allocated to p. We raise the dummy 0 value of player p to a^{i_k}.

We have to show two properties: Firstly that the value allocated to p was at least a^{i_k+1} and secondly that the optimum cost is not affected. Indeed, the first property follows from the fact that $p > q$ (and by the observation that all values of the last q tasks for the players in $\{q+1, \ldots, n\}$ are at least a^{i_k+1}). To show that the optimal solution is not affected consider the optimal allocation which assigns

- the ℓ-th from the end non-dummy task to the ℓ-player, for $\ell < p$
- the ℓ-th from the end non-dummy task to the $(\ell + 1)$-player, for $\ell \geq p$

Notice that this allocation assigns no non-dummy task to the p-th player, as it should. The p-th player is allocated the dummy task, which was raised from 0 to a^{i_k}. Also, since there is a gap in position $k - q$, the total processing time of each player is at most a^{i_k}.

Example 2 Consider the following instance with $n = 5$, $k = 3$, and $q = 2$. Suppose that the mechanism has the allocation indicated by the stars.

$$\begin{pmatrix} 0* & \infty & \infty & \infty & \infty & 1 & a^2 & a^3 \\ \infty & 0* & \infty & \infty & \infty & a & a^3* & a^4 \\ \infty & \infty & 0* & \infty & \infty & a^2 & a^4 & a^5* \\ \infty & \infty & \infty & 0* & \infty & a^3* & a^5 & a^6 \\ \infty & \infty & \infty & \infty & 0* & a^4 & a^6 & a^7 \end{pmatrix}$$

Then $p = 3$, and we can raise the dummy 0 of the 3-rd player to a^3. This does not affect the optimum (which allocates the a^3 values), but raises the cost of the 3-rd player to $a^5 + a^3 \geq a^4 + a^3$.

With the above lemma, we can easily prove the main result:

Proof (Proof of Theorem 1) Consider the instance

$$\begin{pmatrix} 0 & \infty & \cdots & \infty & \infty & 1 & a & \cdots & a^{n-2} \\ \infty & 0 & \cdots & \infty & \infty & a & a^2 & \cdots & a^{n-1} \\ & & \ddots & & & & & & \\ \infty & \infty & \cdots & 0 & \infty & a^{n-2} & a^{n-1} & \cdots & a^{2n-4} \\ \infty & \infty & \cdots & \infty & 0 & a^{n-1} & a^n & \cdots & a^{2n-3} \end{pmatrix}.$$

By the previous lemma, either the approximation ratio is at least $1 + a$ or all non-dummy tasks are allocated to the first player. In the latter case, we raise the dummy 0 of the 1-st player to a^{n-1}. The optimal cost becomes a^{n-1} while the cost of the first player is $1 + a + a^2 + \ldots + a^{n-1}$.

The approximation ratio is at least

$$\min\{1 + \frac{1}{a} + \frac{1}{a^2} + \ldots + \frac{1}{a^{n-1}}, a + 1\}.$$

We select a so that

$$1 + \frac{1}{a} + \frac{1}{a^2} + \ldots + \frac{1}{a^{n-1}} = 1 + a. \tag{1}$$

For $n \to \infty$, this gives

$$\frac{1}{1 - \frac{1}{a}} = 1 + a.$$

Thus $a^2 = 1 + a$, and the solution to this equation is $a = \varphi$. So the approximation ratio of any mechanism is at least $1 + \varphi$. For a fixed number of players n, the solution of Equation 1 determines a lower bound for the approximation ratio. For small values of n, the approximation ratio is less than $1 + \varphi$ but it converges to it rapidly, as shown in Table 3.

4 Conclusion

An observation that might help improving the lower bound, but only to a better constant, is the following: An essential element of the proof of the lower bound of $1 + \sqrt{2}$ [9] was a geometric characterization, equivalent to the monotonicity property, about the way a truthful mechanism for 2 tasks partitions the input space of a player in regions corresponding to the possible allocations. We now have in our machinery the analogous geometric understanding for the case of 3 tasks [35]. Since in this work, we only employ Lemma 1, which is a restricted version of the monotonicity property, it is conceivable that a full geometric argument similar to the one used in [9] may give an improved lower bound.

A better, albeit much more challenging, approach is to obtain a characterization of all possible truthful mechanisms for the case of n players. In fact, to bound the approximation ratio, it suffices to characterize only the decisive mechanisms as it was shown in [6]. Also characterizations for special cases, like the characterization of envy free mechanisms for the case of 2 tasks and n players [16], are of particular interest since they can give us a better understanding of the problem structure.

Acknowledgments

We would like to thank the anonymous reviewers for their thoughtful comments and helpful suggestions.

References

1. Hochbaum, D.: Approximation algorithms for NP-hard problems. PWS Publishing Co. Boston, MA, USA (1996)
2. Lenstra, J., Shmoys, D., Tardos, É.: Approximation algorithms for scheduling unrelated parallel machines. Mathematical Programming **46** (1990) 259–271
3. Horowitz, E., Sahni, S.: Exact and approximate algorithms for scheduling nonidentical processors. J. ACM **23** (1976) 317–327
4. Nisan, N., Ronen, A.: Algorithmic mechanism design. Games and Economic Behavior **35** (2001) 166–196
5. Koutsoupias, E., Vidali, A.: A lower bound of $1+\phi$ for truthful scheduling mechanisms. In: MFCS. (2007) 454–464
6. Dobzinski, S., Sundararajan, M.: On characterizations of truthful mechanisms for combinatorial auctions and scheduling. In: ACM Conference on Electronic Commerce (EC). (2008) 38–47
7. Christodoulou, G., Koutsoupias, E., Vidali, A.: A characterization of 2-player mechanisms for scheduling. In: Algorithms - ESA, 16th Annual European Symposium. (2008) 297–307
8. Mu'alem, A., Schapira, M.: Setting lower bounds on truthfulness. In: Proceedings of the Eighteenth Annual ACM-SIAM Symposium on Discrete Algorithms (SODA). (2007) 1143–1152
9. Christodoulou, G., Koutsoupias, E., Vidali, A.: A characterization of 2-player mechanisms for scheduling. Algorithmica **55** (2009) 729–740
10. Ashlagi, I., Dobzinski, S., Lavi, R.: An optimal lower bound for anonymous scheduling mechanisms. In: ACM Conference on Electronic Commerce (EC). (2009) 169–176
11. Lu, P., Yu, C.: An improved randomized truthful mechanism for scheduling unrelated machines. In: STACS, 25th Annual Symposium on Theoretical Aspects of Computer Science. (2008) 527–538
12. Lu, P., Yu, C.: Randomized truthful mechanisms for scheduling unrelated machines. In: Internet and Network Economics, 4th International Workshop, WINE. (2008) 402–413
13. Christodoulou, G., Koutsoupias, E., Kovács, A.: Mechanism design for fractional scheduling on unrelated machines. ACM Transactions on Algorithms **6** (2010)
14. Cohen, E., Feldman, M., Fiat, A., Kaplan, H., Olonetsky, S.: Envy-free makespan approximation: extended abstract. In: ACM Conference on Electronic Commerce (EC). (2010) 159–166
15. Mu'alem, A.: On multi-dimensional envy-free mechanisms. In: Algorithmic Decision Theory, First International Conference, ADT. (2009) 120–131
16. Christodoulou, G., Kovács, A.: A global characterization of envy-free truthful scheduling of two tasks. In: Internet and Network Economics, 7th International Workshop, WINE. (2011) 84–96
17. Lavi, R., Swamy, C.: Truthful mechanism design for multi-dimensional scheduling via cycle monotonicity. In: ACM Conference on Electronic Commerce (EC). (2007) 252–261
18. Yu, C.: Truthful mechanisms for two-range-values variant of unrelated scheduling. Theoretical Computer Science **410** (2009) 2196–2206

19. Myerson, R.B.: Optimal auction design. Mathematics of Operations Research **6** (1981) 58–73
20. Archer, A., Tardos, É.: Truthful mechanisms for one-parameter agents. In: 42nd Annual Symposium on Foundations of Computer Science (FOCS). (2001) 482–491
21. Archer, A.: Mechanisms for Discrete Optimization with Rational Agents. PhD thesis, Cornell University (2004)
22. Dhangwatnotai, P., Dobzinski, S., Dughmi, S., Roughgarden, T.: Truthful approximation schemes for single-parameter agents. SIAM J. Comput. **40** (2011) 915–933
23. Auletta, V., Prisco, R.D., Penna, P., Persiano, G.: Deterministic truthful approximation mechanisms for scheduling related machines. In: 21st Annual Symposium on Theoretical Aspects of Computer Science (STACS), volume 2996 of LNCS. (2004) 608–619
24. Andelman, N., Azar, Y., Sorani, M.: Truthful approximation mechanisms for scheduling selfish related machines. In: 22nd Annual Symposium on Theoretical Aspects of Computer Science (STACS). (2005) 69–82
25. Kovács, A.: Fast monotone 3-approximation algorithm for scheduling related machines. In: Algorithms - ESA 2005: 13th Annual European Symposium. (2005) 616–627
26. Kovács, A.: Fast Algorithms for Two Scheduling Problems. PhD thesis, Universität des Saarlandes (2007)
27. Christodoulou, G., Kovács, A.: A deterministic truthful ptas for scheduling related machines. In: Twenty-First Annual ACM-SIAM Symposium on Discrete Algorithms, SODA. (2010) 1005–1016
28. Nisan, N., Roughgarden, T., Tardos, E., Vazirani, V.: Algorithmic Game Theory. Cambridge University Press (2007)
29. Vidali, A.: Extending characterizations of truthful mechanisms from subdomains to domains. In: Internet and Network Economics - 7th International Workshop, WINE. (2011) 408–414
30. Saks, M.E., Yu, L.: Weak monotonicity suffices for truthfulness on convex domains. In: Proceedings 6th ACM Conference on Electronic Commerce (EC). (2005) 286–293
31. Gui, H., Müller, R., Vohra, R.V.: Dominant strategy mechanisms with multidimensional types. In: Computing and Markets. Dagstuhl Seminar Proceedings (2005)
32. Lavi, R., Mu'alem, A., Nisan, N.: Towards a characterization of truthful combinatorial auctions. In: 44th Symposium on Foundations of Computer Science (FOCS). (2003) 574–583
33. Monderer, D.: Monotonicity and implementability. In: ACM Conference on Electronic Commerce (EC). (2008) 48
34. Archer, A., Kleinberg, R.: Truthful germs are contagious: A local to global characterization of truthfulness. In: ACM Conference on Electronic Commerce (EC). (2008) 21–30
35. Vidali, A.: The geometry of truthfulness. In: Internet and Network Economics, 5th International Workshop, WINE. (2009) 340–350
36. Kevin, R.: The characterization of implementable choice rules. Aggregation and Revelation of Preferences (1979) 321–348
37. Vickrey, W.: Counterspeculations, auctions and competitive sealed tenders. Journal of Finance **16** (1961) 8–37
38. Clarke, E.: Multipart pricing of public goods. Public Choice **8** (1971) 17–33
39. Groves, T.: Incentives in teams. Econometrica **41** (1973) 617–631