Scheduling without payments[*]

Elias Koutsoupias[†]

Abstract

We consider mechanisms without payments for the problem of scheduling unrelated machines. Specifically, we consider truthful in expectation randomized mechanisms under the assumption that a machine (player) is bound by its reports: when a machine lies and reports value $\tilde{t}_{ij}$ for a task instead of the actual one t_{ij}, it will execute for time $\tilde{t}_{ij}$ if it gets the task (unless the declared value $\tilde{t}_{ij}$ is less than the actual value t_{ij}, in which case, it will execute for time t_{ij}). Our main technical result is an optimal mechanism for one task and n players which has approximation ratio $(n+1)/2$. We also provide a matching lower bound, showing that no other truthful mechanism can achieve a better approximation ratio. This immediately gives an approximation ratio of $(n+1)/2$ and $n(n+1)/2$ for social cost and makespan minimization, respectively, for any number of tasks. We also study the price of anarchy of natural algorithms.

1   Introduction

A major challenge today is to design algorithms that work well even when the input is reported by selfish agents or when the algorithm runs on a system with selfish components. The classical approach is to use mechanism design [18], that is, to design algorithms that use payments to convince the selfish agents to reveal the truth and then compute the outcome using the reported values. Central to the mechanism design approach is the use of dominant strategies as the equilibrium concept. Mechanism design is a very important framework with many unexpected results and it remains a very active research area trying to address some beautiful and challenging problems. Nevertheless, one major problem with mechanism design with payments is that in many situations, the use of payments may not be feasible.

Partly for this reason, there is a lot of recent interest in mechanisms that use no payments (mechanism design without payments) [18]. Given that in many problems we have obtained very poor results using mechanisms with payments, it will be really surprising if the substantially more restricted class of mechanisms without payments can achieve any positive results. For the general unrestricted domain with at least 3 outcomes and truthful mechanisms without payments, the Gibbard-Satterthwaite theorem [10, 24] states that only dictatorial mechanisms are truthful; dictatorial mechanisms are those in which a particular player determines the outcome. Contrast this

---

[†]University of Oxford, `elias@cs.ox.ac.uk`

to Roberts theorem [23], which states that if we allow payments, the truthful mechanisms are the affine maximizers, a much richer class than the dictatorial mechanisms (yet a very poor class in comparison to the set of all possible algorithms).

When we restrict the domain to scheduling unrelated machines, perhaps the most influential problem in algorithmic game theory, the results so far have also been disappointing. The best approximation ratio for the makespan that we know by truthful in expectation mechanisms is $(n+1)/2$ [6], where n is the number of players. The best known lower bound of 2 [6, 16] leaves the possibility open for improved mechanisms. The situation for deterministic mechanisms is similar: the upper bound is easily n [17], and the best known lower bound is 2.61 [13]. It may seem surprising that in this work we can achieve comparable results using mechanisms without payments if we assume that the players are bound by their declarations: Our main result, Theorem 2, is a truthful in expectation mechanism with approximation ratio $n(n+1)/2$. But a moments thought will reveal that we can get a slightly weaker bound (approximation ratio n for one task and hence approximation ratio n^2 for many tasks) with the natural mechanism which allocates each task independently and with probability inversely proportional to the execution times (Proposition 1). The assumption that the players are bound by their declarations plays a crucial role; without it, no positive result is possible. The main value of our main result is that it gives a definite answer (tight upper and lower bounds, albeit only for one task) for this fundamental problem.

Mechanism design without payments is a major topic in game theory [18], although it has not been studied as intensively as the variant with payments in algorithmic game theory. There are however many recent publications which study such mechanisms. Procaccia and Tennenholtz [22] proposed to study approximate mechanism design without payments for combinatorial problems and they studied facility location problems. This work was substantially extended (see for example [15, 14, 1]). Such mechanisms studied also by Dughmi and Ghosh [8] who consider mechanisms for assignment problems and by Guo and Conitzer [11] who consider selling items without payments to 2 buyers. Conceptually, closer to our approach of assuming that the players pay their declared values is the notion of impositions of Nissim, Smorodinsky, and Tennenholtz [19] which was further pursued by Fotakis and Tzamos [9].

2   Model

We study the problem of scheduling tasks when the machines are selfish. We formulate the problem in its more general form, the unrelated machines version: there are n selfish machines and m tasks; the machines are lazy and prefer not to execute any tasks. Machine i needs time $t_{i,j}$ to execute task j. The tasks are allocated to machines with the objective of minimizing the makespan (or the social welfare which is the negation of the sum of executing times of all machines). Let a be an (optimal) allocation for input t, where $a_{i,j}$ is an indicator variable about the event of allocating task j to machine i. The execution time of machine i is $\sum_{j=1}^{m} a_{i,j} t_{i,j}$ and the makespan is $\max_{i=1,\dots,n} \sum_{j=1}^{m} a_{i,j} t_{i,j}$. For randomized algorithms, the allocation variables $a_{i,j}$ are not integral but real values in $[0,1]$ which is the probability to allocate task j to ma-

chine i[1].

In this work, we consider direct revelation mechanisms; a mechanism is simply a randomized scheduling algorithm S which computes an allocation based on the declarations of the machines. There are no payments. More precisely, every machine i reports its private values $\tilde{t}_{i,j}$, one for each task and we apply the algorithm S on this input. The notation $\tilde{t}_{i,j}$ instead of $t_{i,j}$ is used because the machine may lie and not declare its true values $t_{i,j}$. There is however a very important difference with the standard Nisan-Ronen framework [17] for this mechanism design problem. *We assume that the machines are bound by their reports.* More precisely, if a machine i declares a value $\tilde{t}_{i,j} \geq t_{i,j}$ for task i and is allocated the task, *its actual cost is the declared value $\tilde{t}_{i,j}$ and not $t_{i,j}$*. One justification for this assumption is that in some environments the machines can be observed during the execution of the tasks and cannot afford to be caught lying about the execution times. Similar assumptions have been used for other problems. Our assumption is similar to the notion of imposition [19, 9]; for example, in the facility location problem, the players may be forced to use the facility which is closer to their declared position instead of letting them freely choose between the facilities. To complete our assumptions, we need to specify what happens when a machine declares a smaller value, i.e., $\tilde{t}_{i,j} < t_{i,j}$. In this case, we make the simple assumption that the actual cost is the true value $t_{i,j}$; it would be simpler to assume that the machines are not allowed to lie in this direction, but we prefer the weakest assumption since it does not affect our results. To summarize, the cost of machine i for task j is $\max(t_{i,j}, \tilde{t}_{i,j})$.

Our framework now is simple. We design a randomized scheduling algorithm S. The selfish machines report values $\tilde{t}_{i,j}$ and we apply algorithm S to the reported values. This induces a game among the machines: the pure strategies of machine i are the vectors $\left(\tilde{t}_{i,j}\right)_{j=1}^{m}$ with $\tilde{t}_{i,j} \geq 0$. The cost is the execution time computed by algorithm S on input $\tilde{t}$. To be more precise, if $a = S(\tilde{t})$ is the allocation computed by algorithm S on input $\tilde{t}$, the cost of machine i is $c_i = \sum_{j=1}^{m} a_{i,j} \max(t_{i,j}, \tilde{t}_{i,j})$. We seek mechanisms which minimize the makespan $\max_{i=1,\ldots,n} c_i$ or the social cost $\sum_{i=1}^{n} c_i$.

Our assumption is related to mechanisms with verification [17], in which the mechanism learns the actual execution time of the machines and pays after the execution. Because of the delayed payments, these mechanisms are much more powerful than the mechanisms of our framework; for example, they can enforce that the machines are bound by their declarations by imposing a very high penalty for lying. A similar framework was proposed in [4], but in this case the power of the mechanism is limited: it can only deny payment at the end when a machine is caught lying by not being able to finish the tasks within the declared time. In this model, [20, 21] gave collusion-resistant mechanisms that are optimal. The power of this model is in some sense orthogonal to the power of the model of this work: on the one hand, the model of [4, 20, 21] is more powerful in that it can use payments as rewards for telling the truth, but on the other hand it is less powerful in that the players pay their actual cost, not their declared cost.

Mechanisms without money for the scheduling problem have been considered before: coordination mechanisms introduced in [7] are such mechanisms, but they dif-

---

[1]We may also interpret the probabilities as fractional allocations. Naturally, our results apply to the fractional allocations as well.

fer substantially from the model of this work in that the players are the tasks, not the machines. This and subsequent work [2, 5] considered truthful mechanisms with the assumption, same with the assumption of this work, that the execution time of a task is the maximum of the declared and actual value, a condition which is more easily enforceable when the machines are controlled by the mechanism.

2.1 The case of a single task

We focus on the simple case of one task and n machines. Let $t_1, \ldots, t_n$ be the true values of the machines for the task and let $\tilde{t}_1, \ldots, \tilde{t}_n$ be the declared values; we drop the second subscript since there is only one task. Let $p_i(\tilde{t})$ be the probability of allocating the task to player i. The expected cost of player i is $c_i = c_i(t_i, \tilde{t}) = p_i(\tilde{t}) \max(t_i, \tilde{t}_i)$, while the social cost of the algorithm is $\sum_{i=1}^{n} c_i$ (in the case of the single task, the makespan and social cost are identical).

The mechanism is called truthful if for every t_{-i}, the expected cost c_i of player i is minimized when $\tilde{t}_i = t_i$. This notion of truthfulness, truthful in expectation, is the weakest notion of truthfulness which contains a richer class of mechanisms than the standard notion of truthfulness or universal truthfulness. It is not ex post truthful, meaning that after the players see the outcome of the coins, they may want to change their declarations. In contrast, universal truthfulness is ex post truthfulness, and it is trivial to see that this stronger notion of truthfulness cannot achieve any positive result. Notice however, that for the fractional version of the scheduling problem, in which we can allocate parts of the same tasks to different machines, we can consider deterministic algorithms and consequently the strongest notion of truthfulness.

Lemma 1. *An algorithm is truthful if and only if for every i and t_{-i}, $t_i p_i(t)$ is non-decreasing in t_i and $p_i(t)$ is non-increasing in t_i.*

Proof. Indeed, if $t_i p_i(t)$ is non-decreasing in t_i, the player prefers t_i to higher values, i.e., when $\tilde{t}_i \geq t_i$, the cost $c_i = p_i(\tilde{t}_i, t_{-i})\tilde{t}_i$ is minimized at $\tilde{t}_i = t_i$. Similarly, if $p_i(t_i)$ is non-increasing, the player prefers t_i to smaller values, i.e., when $\tilde{t}_i \leq t_i$, the cost $c_i = p_i(\tilde{t}_i, t_{-i})t_i$ is again minimized at $\tilde{t}_i = t_i$.

Conversely, if there exist $x < y$ with $p_i(t_{-i}, y)y < p_i(t_{-i}, x)x$, then player i gains by lying: when $t_i = x$ he prefers to declare y. Similarly, if there exist $x > y$ with $p_i(t_{-i}, x) > p_i(t_{-i}, y)$, the player would again prefer to declare y when $t_i = x$. $\square$

A mechanism is defined simply by the probability functions $p_i(t)$. The expected makespan is $\sum_i c_i$ and its approximating ratio is $\sum_i c_i / \min_i t_i$. We seek truthful mechanisms with small approximation ratio.

3 Truthful mechanisms for one task

In this section, we study the case of a single task. We first consider a natural mechanism, the *proportional* algorithm: It allocates the task to machine i with probability inversely proportional to the declared value t_i, i.e., $p_i(t) = t_i^{-1} / \sum_k t_k^{-1}$.

Proposition 1. *The proportional algorithm is truthful and achieves approximation ratio n.*

Proof. Indeed, to verify that the mechanism is truthful, it suffices to observe that $p_i(t)$ is non-increasing in t_i and that $t_i p_i(t)$ is non-decreasing it t_i. The expected makespan of this mechanism is $n/\sum_i t_i^{-1}$ while the optimal makespan is $\min_i t_i$. It follows that the approximation ratio is at most n and that it can be arbitrarily close to n (when for example one value is 1 and the other $n-1$ values are arbitrarily high). □

It is natural to ask whether there are better mechanisms than the proportional mechanism. In the next subsection, we give a positive answer by designing an optimal mechanism, albeit with not substantially better approximation ratio.

3.1   An optimal truthful mechanism

In this subsection, we study truthful algorithms that have optimal approximation ratio.

To find an optimal truthful mechanism, we want to find functions $p_i(t)$ such that for every t:

- for every i: $t_i p_i(t)$ is non-decreasing in t_i,

- for every i: $p_i(t)$ is non-increasing in t_i,

- $\sum_i p_i(t) = 1$

and which minimize $\max_t \sum_i t_i p_i(t)/\min_i t_i$. The first two conditions capture truthfulness and the third condition the natural property that the probabilities add to 1.

We will show the following theorem:

Theorem 1. *There is a truthful in expectation mechanism without payments with approximation ratio $(n+1)/2$. Conversely, no truthful in expectation mechanism without payments can have approximation ratio better than $(n+1)/2$.*

Before proceeding with the proof of the theorem, it is instructive to consider first the case of $n = 2$ players. We will consider a *symmetric* mechanism, so it suffices to give the probabilities $p_i(t)$ of assigning the task to player i when $t_1 \leq t_2$. We claim that the mechanism with probabilities

$$p_1(t) = 1 - \frac{t_1}{2t_2} \qquad\qquad p_2(t) = \frac{t_1}{2t_2}$$

is truthful and has approximation ratio $3/2$. To clarify: these are the probabilities when $t_1 \leq t_2$; by symmetry, we can compute the probabilities when the declared value of the second player is smaller than the declared value of the first player.

Let us verify that this mechanism is truthful. Specifically we want to show that the expected cost of player i is minimized when he declares his true value; this must hold for every value t_{-i} of the other player. Consider first player 1 (the one with true value less than the declared value of the other player).

- He has no reason to declare something less than t_1, because $p_1(t)$ is non-increasing in t_1, consequently $t_i p_i(t) \leq t_i p_i(\tilde{t}_i, t_{-i})$.

- He has no reason to declare something in (t_1, t_2), because $t_1 p_1(t) = t_1 - \frac{t_1^2}{2t_2} = \frac{t_2^2 - (t_2 - t_1)^2}{2t_2}$ is increasing in t_1 for $t_1 < t_2$.

- Finally, he has no reason to declare something in $[t_2, \infty)$. In this case, his lie changes the order of the values, and, by the definition of the mechanism, the probability of getting the task will be $p_2(t_2, \tilde{t}_1)$. Nevertheless, we still have

$$t_1 p_1(t) = \frac{t_2^2 - (t_2 - t_1)^2}{2t_2} \le \frac{t_2}{2} = \tilde{t}_1 \frac{t_2}{2\tilde{t}_1} = \tilde{t}_1 p_2(t_2, \tilde{t}_1).$$

We work similarly for the second player (the one with true value greater than the declared value of the other player). If he declares his true value t_2, his expected cost is $t_2 p_2(t) = t_1/2$.

- He has no reason to declare something less than t_1, because in this case the probability $p_1(t_1, \tilde{t}_2)$ of getting the task is at least $1/2$ and his cost will be at least $t_2/2 \ge t_1/2$.

- He has no reason to declare any other value greater than t_1 because his cost is going to be $\tilde{t}_2 p_2(t_1, \tilde{t}_2) = t_1/2$, anyway.

The above mechanism has approximation ratio $3/2$, because the cost is

$$t_1 p_1(t) + t_2 p_2(t) = t_1 - \frac{t_1^2}{2t_2} + \frac{t_1}{2} = \frac{3}{2} t_1 - \frac{t_1^2}{2t_2} \le \frac{3}{2} t_1.$$

Trivially, the approximation ratio tends to $3/2$ as t_2 tends to ∞.

We proceed to generalize the above to more than two players. Again, we define a symmetric mechanism, so it suffices to describe it when $t_1 \le \cdots \le t_n$:

$$p_1 = \frac{1}{t_1} \int_0^{t_1} \prod_{i=2}^{n} \left(1 - \frac{y}{t_i}\right) dy \tag{1}$$

$$p_k = \frac{1}{t_k t_1} \int_0^{t_1} \int_0^{y} \prod_{i=2\ldots n, i \ne k} \left(1 - \frac{x}{t_i}\right) dx\, dy \qquad \text{for } k \ne 1$$

For example, for $n = 2$ we get the mechanism discussed above, and for $n = 3$ the probabilities are

$$p_1 = 1 - \frac{t_1(t_2 + t_3)}{2t_2 t_3} + \frac{t_1^2}{3t_2 t_3} \qquad p_2 = \frac{t_1}{2t_2} - \frac{t_1^2}{6t_2 t_3} \qquad p_3 = \frac{t_1}{2t_3} - \frac{t_1^2}{6t_2 t_3}.$$

This definition is not arbitrary, but it is the natural solution to the requirements at the beginning of this subsection. This will become apparent as we proceed to show that this is an optimal algorithm for our problem.

First we verify that the mechanism is well-defined: We need to show that these probabilities are nonnegative and add up to 1. Indeed, consider the quantities q_i inside the integrals

$$q_1(y) = \prod_{i=2}^{n} \left(1 - \frac{y}{t_i}\right)$$

$$q_k(y) = \frac{1}{t_k} \int_0^y \prod_{i=2...n, i \neq k} \left(1 - \frac{x}{t_i}\right) dx$$

for which

$$p_i = \frac{1}{t_1} \int_0^{t_1} q_i(y) \, dy.$$

Since the integral is for $y \leq t_1$ and t_1 is the minimum among the t_k's, all factors in these expressions are nonnegative. This shows that $q_i(y) \geq 0$ for every $i = 1, \ldots, n$. We also observe that $q_1'(y) = -\sum_{i=2}^{n} q_k'(y)$ which shows that $\sum_{i=1}^{n} q_i(y)$ is constant; taking $y = 0$, we see that this constant is 1. In summary the p_i's are nonnegative and their sum is $\frac{1}{t_1} \int_0^{t_1} \sum_{i=1}^{n} q_i(y) \, dy = \frac{1}{t_1} \int_0^{t_1} 1 \, dy = 1$.

We also need to verify that (1) defines the probabilities consistently. More precisely, the definition of the probabilities is based on the order of the values; when these values are distinct, there is a unique order of the values $t_1 < t_2 < \ldots < t_n$, and there is no inconsistency. However, when some values are equal, there are many orders of the values and we need to guarantee that every such order gives the same probabilities. For all values except the minimum one, this follows directly from the symmetry of the algebraic expressions in (1). Thus we need only to verify consistency when the first two values are equal and specifically, we need to show that if $t_1 = t_2$, then $p_1 = p_2$. Indeed, it is straightforward to verify it by employing the following easy identity for every function g:

$$\int_0^a (a - y) g(y) \, dy = \int_0^a \int_0^y g(x) \, dx \, dy^2$$

In this case, $g(y) = \prod_{i=3}^{n} \left(1 - \frac{y}{t_i}\right)$ and $a = t_1$.

We now proceed to establish that the mechanism is truthful.

Lemma 2. *The symmetric mechanism defined by the probabilities in* (1) *is truthful.*

Proof. To show that the algorithm is truthful, we observe that

- The probabilities p_i are non-increasing in t_i. This is trivially true for $i \neq 1$ and it can be easily verified for $i = 1$. The fact that the probabilities are non-increasing in t_i shows that no player i has a reason to lie and declare a value $\tilde{t}_i < t_i$. To see

---

[2]Proof: Let $G(y) = \int_0^y g(x) \, dx$. Then

$$\int_0^a (a - y) g(y) \, dy - \int_0^a \int_0^y g(x) \, dx \, dy = \int_0^a (a - y) g(y) - G(y) \, dy$$

$$= \int_0^a a g(y) - (y G(y))' \, dy$$

$$= a G(a) - a G(a) = 0.$$

this, fix the values of the remaining players and assume that player i changes his value to $\tilde{t}_i < t_i$. We will argue that the probability of getting the task with the new value is greater than or equal to the original probability; this suffices because the cost of the player is this probability times t_i.

If, after the change, the order of the players remains the same, the probability does not decrease by this change because p_i is non-increasing in t_i. We need to show the same when the change affects the order of the players. This turns out to be easy but we note that we need to be careful, because the expressions that define the probabilities of the mechanism depend on the order of the values. However, we don't need the expressions of the probabilities but a simpler argument: imagine that we lower the value from t_i to $\tilde{t}_i$ in stages: from t_i to t_{i-1} to t_{i-2}, and so on, to t_{i-k}, and finally to $\tilde{t}_i$. In each stage, the order of the values remains the same and therefore the probability can only increase. It follows that it does not decrease from the total change. The fact that the mechanism is symmetric is crucial in this argument because the algebraic expressions of the probability of the player may change from stage to stage, but the values at the boundaries $t_{i-1}, \ldots, t_{i-k}$ of successive stages are the same.

- every player k for $k \neq 1$ is truthful since $t_k p_k$ is independent of his value t_k; this holds even when the player reports a higher value which may change the order of the players

- player 1 has no reason to lie and report a value in $(t_1, t_2]$ because $t_1 p_1$ is increasing in t_1; furthermore, player 1 has no reason to report a higher value than t_2 which will change the order of the players because the cost will change from $t_1 p_1$ to $t'_k p'_k = \frac{1}{t_2} \int_0^{t_2} \int_0^y \prod_{i=3}^n \left(1 - \frac{x}{t_i}\right) dx\,dy$ (because now the minimum value is t_2). It suffices therefore to show

$$t_1 p_1 \leq t'_k p'_k \qquad\qquad \Leftrightarrow$$

$$\int_0^{t_1} \prod_{i=2}^n \left(1 - \frac{y}{t_i}\right) dy \leq \frac{1}{t_2} \int_0^{t_2} \int_0^y \prod_{i=3}^n \left(1 - \frac{x}{t_i}\right) dx\,dy \qquad \Leftarrow$$

$$\int_0^{t_2} \prod_{i=2}^n \left(1 - \frac{y}{t_i}\right) dy \leq \frac{1}{t_2} \int_0^{t_2} \int_0^y \prod_{i=3}^n \left(1 - \frac{x}{t_i}\right) dx\,dy \qquad \Leftrightarrow$$

$$\int_0^{t_2} (t_2 - y) \prod_{i=3}^n \left(1 - \frac{y}{t_i}\right) dy \leq \int_0^{t_2} \int_0^y \prod_{i=3}^n \left(1 - \frac{x}{t_i}\right) dx\,dy$$

The last holds because of Equation (2).

Putting everything together, we see that the mechanism is indeed truthful. $\qquad\square$

Lemma 3. *The symmetric mechanism defined by the probabilities in* (1) *has approximation ratio* $(n+1)/2$.

Proof. Now that we have established that the algorithm is truthful, we proceed to bound its approximation ratio. The approximation ratio is $\sum_{i=1}^n t_i p_i / t_1$. Clearly, $t_1 p_1 \leq$

t_1 and for $k > 1$

$$t_k p_k = \frac{1}{t_1} \int_0^{t_1} \int_0^y \prod_{i=2..n, i \neq k} \left(1 - \frac{x}{t_i}\right) dx \, dy$$

$$\leq \frac{1}{t_1} \int_0^{t_1} \int_0^y 1 \, dx \, dy$$

$$= t_1/2.$$

Therefore, $\sum_{i=1}^n t_i p_i \leq t_1 + (n-1) t_1/2 = t_1(n+1)/2$, which shows that the approximation ratio is at most $(n+1)/2$. It is trivial that if we fix the other values and let t_1 tend to 0, the above inequalities are almost tight, and therefore the approximation ratio can be arbitrarily close to $(n+1)/2$. $\square$

We will show that no other truthful algorithm has a better approximation ratio. To do this, we first show that the optimal approximation ratio is achieved by a symmetric mechanism. By *symmetric* or *anonymous*, we mean a mechanism with the property that if we permute the indexes of the players, the allocation probabilities are permuted similarly. More formally, a mechanism is symmetric if for every t and every permutation π: $p(\pi \cdot t) = \pi \cdot p(t)$, where $\pi \cdot t$ denotes $(t_{\pi_1}, \ldots, t_{\pi_n})$. This in particular implies that if two players have equal values they have the same allocation probabilities, a property that we will use to establish the lower bound below.

Lemma 4. *The optimal approximation ratio of the symmetric truthful mechanisms is the same with the approximation ratio of all truthful mechanisms.*

Proof. Let p be the allocation probabilities of an arbitrary truthful mechanism. Consider the mechanism which selects a (uniformly) random permutation π of the values, computes the allocation probabilities using p, and then assigns the probabilities to the players by taking the inverse permutation π^{-1}. More formally, the new mechanism has allocation probabilities $p'(t) = E_\pi[\pi^{-1} \cdot p(\pi \cdot t)]$ and it is clearly symmetric.

We now establish that if p is truthful, then p' is also truthful. The reason is simple: when we permute players and values by permutation π, player i plays the role of player π_i; since player π_i is truthful in the original mechanism, player i has no reason to lie. More formally, $p'_i(t) = E_\pi[p_{\pi_i^{-1}}(\pi \cdot t)]$. To establish the truthfulness of the new mechanism, we need to establish the monotonicity of $p'_i(t)$ and $t_i p'_i(t)$. Let $t' = \pi \cdot t$. Since the original mechanism is truthful, $p_{\pi_i^{-1}}(t')$ is monotone in $t'_{\pi_i^{-1}} = t_i$ and therefore $p'_i(t)$, being the expectation of monotone functions in t_i, is also monotone. An almost identical argument establishes that $t_i p'_i(t)$ is also monotone. In summary, p' is a truthful mechanism.

We will now argue that the approximation ratio of the new symmetric mechanism is not higher than the approximation ratio of the original mechanism. Indeed, suppose that the original mechanism has approximation ratio ρ. This implies that for every t and every permutation π, the cost of the mechanism when the values are $\pi \cdot t$ is at most $\rho \min_i t_{\pi_i} = \rho \min_i t_i$. It follows that if π is selected randomly, the expected cost of the original mechanism, which is equal to the cost of the new mechanism, is at most $\rho \min_i t_i$; this shows that the approximation ratio of the new mechanism is at most ρ. $\square$

Lemma 5. *No truthful in expectation mechanism without payments has approximation ratio smaller than $(n+1)/2$.*

Proof. We will employ instances with values of the form $(1,1,m,\dots,m)$ and $(1,m,\dots,m)$ where m is some large value (which we will allow to tend to infinity to obtain the lower bound). By the previous lemma, we can assume without loss of generality that the mechanism is symmetric. Let p and p' be the probabilities assigned by an algorithm to the above instances. For the instance $(1,1,m,\dots,m)$, we have $2p_2+(n-2)p_3=1$ and the approximation ratio is at least $r \geq 2p_2+m(n-2)p_3 = m-2p_2(m-1)$. Similarly for the other instance $(1,m,\dots,m)$ we have $p'_1+(n-1)p'_2=1$ and $r \geq p'_1+(n-1)mp'_2 = 1+(m-1)(n-1)p'_2$.

The crucial step is to use the truthfulness of player 2 to connect the two instances. Specifically, the second player is truthful only when $1 \cdot p_2 \leq m \cdot p'_2$. Substituting the value of p'_2 in the bound of the second instance, we get that the approximation is at least $1+p_2(m-1)(n-1)/m$.

In summary, the approximation ratio is at least $\max\{m-2p_2(m-1), 1+p_2(m-1)(n-1)/m\}$. The first expression is decreasing in p_2 while the second one is increasing in p_2; the minimum approximation ratio is achieved when the expressions are equal, that is, when $p_2 = m/(2m+n-1)$ which gives ratio at least $(n+1)/2 - (n^2-1)/(4m+2n-2)$. As m tends to infinity, the approximation ratio tends to $(n+1)/2$. □

4   Extension to many tasks and discussion

In the previous section, we gave an optimal mechanism for one task. We can use it to get a mechanism for many tasks by running it independently for every task. Since the tasks are allocated independently, the resulting mechanism remains truthful. If the objective is the social cost (i.e., to minimize the sum of the cost of all players), the mechanism clearly retains its approximation ratio. If the objective however is the makespan, then the approximation ratio is at most $n(n+1)/2$, for the simple reason that $\max_i c_i \leq \sum_i c_i \leq n\max_i c_i$. So we get

Theorem 2. *There is a truthful in expectation mechanism without payments for the problem of scheduling unrelated machines with approximation ratio $(n+1)/2$ when the objective is the social cost and $n(n+1)/2$ when the objective is the makespan.*

It follows from the case of one task that the approximation ratio $(n+1)/2$ is tight for the social cost. It is not clear that the pessimistic way we used to bound the approximation ratio of the makespan is tight. It remains open to estimate the approximation ratio of the given mechanism for many tasks. It is also open whether this is the best mechanism without payments for many tasks, that is, whether there are other (task-independent or not) mechanisms without payments for many tasks with better approximation ratio.

4.1   Mechanisms and the price of anarchy

So far, we have considered truthful mechanisms. It is however natural to ask whether there are non-truthful mechanisms with better performance. To formulate rigorously

this question, we need first to agree on how to quantify the performance of non-truthful mechanisms, an issue that did not arise for truthful mechanisms because they have unique dominant equilibria. A natural way to evaluate non-truthful mechanism is to consider their price of anarchy (the performance of the worst Nash equilibrium) or their price of stability (the performance of the best Nash equilibrium) [12, 3].

To explain the issues of this approach, we analyze here the non-truthful natural greedy algorithm which gives the task to the machine with minimum reported time. Assume also that the ties are broken randomly (or in any deterministic way). We argue below that

Proposition 2. *The price of stability of the greedy algorithm for one task is 1, while its price of anarchy is unbounded.*

Proof. We will consider only the case of 2 players since it can be directly generalized to the case of more players. We assume that the actual values of the two players are $t_1 \leq t_2$.

For the price of stability, consider the Nash equilibrium in which the first player reports his true value t_1 while the second player declares any value $x \geq t_2$ with cumulative distribution $F_2(x) = 1 - t_2/x$. The cost of the players is t_1 and 0 respectively, and the social cost is t_1, optimal. To see that this is indeed a Nash equilibrium, notice that the first player has no reason to switch: if he reports a value x in $[t_2, \infty)$ his expected cost would be $xPr[t_2 > x] = x(1 - F_2(x)) = t_2$. Trivially the second player has no reason to switch since his cost is already 0.

For the price of anarchy, consider a similar Nash equilibrium in which the second player reports his true value t_2 and the first player reports a random value in $[t_2, \infty)$ with cumulative distribution $F_1(x) = 1 - t_2/x$. It is easy to check that this is indeed Nash equilibrium and that the social cost is t_2. The price of anarchy is t_2/t_1 which is unbounded. □

The optimal truthful mechanism of Subsection 3.1 for one task has price of anarchy (and stability) $(n+1)/2$. Because it is optimal, no other truthful mechanism has a better price of anarchy. Is there a non-truthful mechanism with better price of anarchy? If the answer is positive, it will perhaps be a challenging open problem to find an algorithm which achieves the minimum price of anarchy for the case of a single task, as well as the case of many tasks.

References

[1] Alon, N., Feldman, M., Procaccia, A., Tennenholtz, M.: Strategyproof approximation of the minimax on networks. Mathematics of Operations Research **35**(3), 513–526 (2010)

[2] Angel, E., Bampis, E., Pascual, F., Tchetgnia, A.: On truthfulness and approximation for scheduling selfish tasks. Journal of Scheduling **12**(5), 437–445 (2009)

[3] Anshelevich, E., Dasgupta, A., Kleinberg, J., Tardos, E., Wexler, T., Roughgarden, T.: The price of stability for network design with fair cost allocation. In: Founda-

tions of Computer Science, 2004. Proceedings. 45th Annual IEEE Symposium on, pp. 295–304. IEEE (2004)

[4] Auletta, V., De Prisco, R., Penna, P., Persiano, G.: The power of verification for one-parameter agents. Automata, Languages and Programming pp. 63–82 (2004)

[5] Christodoulou, G., Gourves, L., Pascual, F.: Scheduling selfish tasks: about the performance of truthful algorithms. Computing and Combinatorics pp. 187–197 (2007)

[6] Christodoulou, G., Koutsoupias, E., Kovács, A.: Mechanism design for fractional scheduling on unrelated machines. Automata, Languages and Programming pp. 40–52 (2007)

[7] Christodoulou, G., Koutsoupias, E., Nanavati, A.: Coordination mechanisms. Automata, Languages and Programming pp. 45–56 (2004)

[8] Dughmi, S., Ghosh, A.: Truthful assignment without money. In: Proceedings of the 11th ACM conference on Electronic commerce, pp. 325–334. ACM (2010)

[9] Fotakis, D., Tzamos, C.: Winner-imposing strategyproof mechanisms for multiple facility location games. WINE pp. 234–245 (2010)

[10] Gibbard, A.: Manipulation of voting schemes: a general result. Econometrica: Journal of the Econometric Society pp. 587–601 (1973)

[11] Guo, M., Conitzer, V.: Strategy-proof allocation of multiple items between two agents without payments or priors. In: Proceedings of the 9th International Conference on Autonomous Agents and Multiagent Systems: volume 1-Volume 1, pp. 881–888. International Foundation for Autonomous Agents and Multiagent Systems (2010)

[12] Koutsoupias, E., Papadimitriou, C.H.: Worst-case equilibria. Computer Science Review **3**(2), 65–69 (2009). An earlier version appeared in STACS 1999

[13] Koutsoupias, E., Vidali, A.: A lower bound of $1+\varphi$ for truthful scheduling mechanisms. Mathematical Foundations of Computer Science 2007 pp. 454–464 (2007)

[14] Lu, P., Sun, X., Wang, Y., Zhu, Z.: Asymptotically optimal strategy-proof mechanisms for two-facility games. In: Proceedings of the 11th ACM conference on Electronic commerce, pp. 315–324. ACM (2010)

[15] Lu, P., Wang, Y., Zhou, Y.: Tighter bounds for facility games. WINE pp. 137–148 (2009)

[16] Mu'alem, A., Schapira, M.: Setting lower bounds on truthfulness: extended abstract. In: Proceedings of the eighteenth annual ACM-SIAM symposium on Discrete algorithms, pp. 1143–1152. Society for Industrial and Applied Mathematics (2007)

[17] Nisan, N., Ronen, A.: Algorithmic mechanism design (extended abstract). In: Proceedings of the thirty-first annual ACM symposium on Theory of computing, pp. 129–140. ACM (1999)

[18] Nisan, N., Roughgarden, T., Tardos, E., Vazirani, V.: Algorithmic game theory. Cambridge Univ Pr (2007)

[19] Nissim, K., Smorodinsky, R., Tennenholtz, M.: Approximately optimal mechanism design via differential privacy. Arxiv preprint arXiv:1004.2888 (2010)

[20] Penna, P., Ventre, C.: Collusion-resistant mechanisms with verification yielding optimal solutions. Algorithms-ESA 2008 pp. 708–719 (2008)

[21] Penna, P., Ventre, C.: Optimal collusion-resistant mechanisms with verification. Games and Economic Behavior (2012)

[22] Procaccia, A., Tennenholtz, M.: Approximate mechanism design without money. In: Proceedings of the tenth ACM conference on Electronic commerce, pp. 177–186. ACM (2009)

[23] Roberts, K.: The characterization of implementable choice rules. Aggregation and Revelation of Preferences pp. 321–348 (1979)

[24] Satterthwaite, M.: Strategy-proofness and Arrow's conditions: Existence and correspondence theorems for voting procedures and social welfare functions. Journal of Economic Theory **10**(2), 187–217 (1975)