Randomized Truthful Mechanisms for Scheduling Unrelated Machines*

Pinyan Lu and Changyuan Yu

Institute for Theoretical Computer Science, Tsinghua University
Beijing, 100084, P. R. China
{lpy, yucy05}@mails.tsinghua.edu.cn

Abstract. In this paper, we consider randomized truthful mechanisms for scheduling tasks to unrelated machines, where each machine is controlled by a selfish agent. Some previous work on this topic focused on a special case, scheduling two machines, for which the best approximation ratio is 1.6737 [5] and the best lower bound is 1.5 [6]. For this case, we give a unified framework for designing universally truthful mechanisms, which includes all the known mechanisms, and also a tight analysis method of their approximation ratios. Based on this, we give an improved randomized truthful mechanism, whose approximation ratio is 1.5963. For the general case, when there are m machines, the only known technique is to obtain a $\frac{\gamma m}{2}$-approximation truthful mechanism by generalizing a γ-approximation truthful mechanism for two machines[6]. There is a barrier of $0.75m$ for this technique due to the lower bound of 1.5 for two machines. We break this $0.75m$ barrier by a new designing technique, rounding a fractional solution. We propose a randomized truthful-in-expectation mechanism that achieves approximation of $\frac{m+5}{2}$, for m machines.

For the lower bound side, we focus on an interesting family of mechanisms, namely *task-independent* truthful mechanisms. We prove a lower bound of 11/7 for two machines and a lower bound of $\frac{m+1}{2}$ for m machines with respect to this family. They almost match our upper bounds in both cases.

1 Introduction

Mechanism design, an important area both in Game Theory and Computer Science, has received extensive study in the past few years. It is usually used to design a protocol for achieving some global objective, however requiring the interaction of some selfish agents. To deal with this, the most common solution concept is "truthfulness", where the mechanism is designed so that for any participant agent, reporting his/her private data truthfully to the mechanism

* Supported by the National Natural Science Foundation of China Grant 60553001 and the National Basic Research Program of China Grant 2007CB807900, 2007CB807901.

will always maximize his/her own utility, no matter how other agents act. We also focus on truthful mechanisms in this paper.

The study of the algorithmic aspect of mechanism design was initiated by Nisan and Ronen in their seminal paper "Algorithmic Mechanism Design"[8]. Some computational properties such as good approximation ratios and polynomial running time are studied in mechanism design setting. Nisan and Ronen's work mainly focused on a fundamental problem in computer science, scheduling unrelated machines. In a scheduling problem, there are n tasks to be allocated to m machines, which are controlled by selfish agents. The objective is to allocate the tasks so that the maximum completion time of these machines (called *makespan*) is minimized. A mechanism for the scheduling problem consists of two algorithms, the allocation algorithm and the payment algorithm. Our main interest is on the approximation ratio of the allocation algorithm. Nisan and Ronen proposed a deterministic truthful mechanism with an approximation ratio of m. Moreover, they proved a lower bound of 2 for all the deterministic truthful mechanisms. Randomization is always more powerful, and this is also true for this scheduling problem. They provided a randomized truthful mechanism with approximation ratio of 1.75 for two machines. Recently Mu'alem and Schapira gave a lower bound of $2 - 1/m$ for randomized truthful mechanisms[6]. They also generalized the 1.75 approximation mechanism for two machines to a $0.875m$-approximation mechanism for m machines. In a previous work[5], we improved Nisan and Ronen's result by a 1.67-approximation randomized truthful mechanism for two machines, together with a $0.837m$-approximation mechanism for m machines using Mu'alem and Schapira's technique in [6].

A fractional variant of truthful scheduling unrelated machines was first considered by Christodoulou, Koutsoupias and Kovács in [2]. They gave a fractional truthful mechanism with approximation ratio of $(m+1)/2$, and a lower bound of $2 - 1/m$ for any fractional truthful mechanisms. They also defined a family of allocation algorithms named as *task-independent* algorithm, in which tasks are allocated independently. For the task-independent truthful fractional mechanisms, they proved a tight lower bound of $(m + 1)/2$.

1.1 Our Results

In this paper, we first propose a unified approach to design truthful mechanisms for two machines, which contains all the known truthful mechanisms. One main contribution is that we not only unify all the known mechanisms, but also give a unified and tight analysis method for their approximation ratios. Based on this, we are able to give a randomized mechanism for two machines, which is universally truthful and has an approximate ratio of 1.5963.

A natural question would be how far we can go with this unified approach. We answer this question by a lower bound of 1.5788 for this approach. Further more, we also prove a lower bound of 11/7 for all the task independent randomized mechanisms which are truthful even in a weaker version, i.e., truthful in expectation. So substantial new techniques are required to significantly improve our results.

For the general case, when there are m machines, the only known technique is to obtain a $\frac{\gamma m}{2}$-approximation truthful mechanism by generalizing a γ-approximation truthful mechanism for two machines[6]. However the lower bound of 1.5 for scheduling two machines gives a barrier of $0.75m$ for this technique. We break this $0.75m$ barrier by a new designing technique. First, we adopt a truthful fractional mechanism with ratio $(m+1)/2$ by Christodoulou, Koutsoupias and Kovács [2]. We add into this mechanism an important threshold so that it satisfies certain "bid condition", which is essential for us to bound the loss of approximation ratio during the rounding process. Then we use a rounding technique in [4] to get a randomized mechanism, which is still truthful in expectation, and only loses little in approximation ratio. We finally obtain a randomized mechanism which is truthful in expectation and achieves an approximation ratio of $(m+5)/2$.

We also give a lower bound of $(m+1)/2$ for all task independent randomized mechanisms. This result shows that we really need some new techniques to break this $0.5m$ barrier.

2 Preliminaries and Notations

In this section we review some definitions and results on mechanism design and scheduling problem. More details can be found in[8]. In the following, for a generic matrix $a = (a_{ij})$, we use a_i to denote the i-th row of the matrix, and a_{-i} to denote the matrix obtained from a deleting a_i. We also use (v, a_{-i}) to denote the matrix obtained from a by replacing a_i with vector v. We use R_+ to denote the set of nonnegative real numbers.

In a scheduling problem, there are n tasks and m machines, where each machine $i \in [m]$ needs t_{ij} units of time to perform task $j \in [n]$. We usually use the matrix $t = (t_{ij})$ to denote an instance of the scheduling problem. In this paper, we consider that each machine is controlled by a strategic player. We assume that player i privately knows t_i, and we call the vector t_i player i's type. After each player i declares his/her type, an allocation algorithm x will decide an allocation of all the tasks. We assume that all the players are selfish and want to perform as less tasks as possible, so players may misreport their types. We use $b_i \in R_+^n$ to denote player i's reported type, and call it player i's bid. Obviously b_i may not equal to t_i if that helps in player i's interest. To avoid this lying issue, we introduce the payment algorithm p into a mechanism. Formally, a mechanism $M = (x, p)$ consists of two parts:

- **An allocation algorithm:** the allocation algorithm x, given the input of players' bid matrix $b = (b_1, \cdots, b_m)$, outputs an allocation denoted by a matrix $x = (x_{ij})$. x_{ij} is 1 if task j is assigned to machine i, and 0 otherwise. In the fractional scheduling case, x_{ij} satisfies $0 \le x_{ij} \le 1$ and denotes the fraction of task j assigned to machine i. Every task must be completely assigned, hence $\sum_{j\in[n]} x_{ij} = 1, \forall i \in [m]$. Notice that each x_{ij} can be viewed as a function of b.

– **A payment algorithm:** the payment algorithm p, given the input of players' bid matrix b, outputs a vector $p = (p_1, \cdots, p_m)$, where p_i denotes the money that player i receives from the mechanism. Each p_i can also be viewed as a function of b.

Randomized mechanism is defined to be a distribution of several deterministic mechanisms. In randomized mechanism, x_{ij} is a random variable denoting whether task j is assigned to machine i. For simplicity, we also use x_{ij} to denote $Pr(x_{ij} = 1)$ when the context is clear.

Now we specify the utility of each player. We use the quasi linear utility, which means the utility u_i of player i with type t_i over an allocation x and money p_i is defined as:

$$u_i(x, p_i|t_i) = p_i - \sum_{j\in[n]} x_{ij}t_{ij}.$$

Since x and p_i are both functions of bid matrix b, we can also write the utility as

$$u_i(b|t_i) = p_i(b) - \sum_{j\in[n]} x_{ij}(b)t_{ij}.$$

Recall that we want to solve the issue of lying about types, we are interested in truthful mechanisms. A mechanism $M = (x, p)$ is truthful if for each player i, reporting his/her true type will maximize his/her own utility. Formally, for any i, any bids b_{-i} of all other players, we have

$$u_i((t_i, b_{-i})|t_i) \geq u_i((b_i, b_{-i})|t_i), \quad \forall b_i \in R_+^n$$

For randomized mechanism, there are two versions of truthfulness. The stronger version is universally truthful, which requires the mechanism to be truthful when fixing all the random bits. The weaker version is truthful in expectation, which only requires that for each player, reporting his/her true type will maximize his/her own expected utility.

For a truthful mechanism M, we may assume that all the players will report their true types, hence $b = t$. Now, how can we evaluate the performance of mechanism's allocation algorithm x? We consider the makespan, which is the maximum load of all the machines. Given input t, the makespan of mechanism M is denoted by $l_M(t)$, and $l_M(t) = \max_{i\in[m]} \sum_{j\in[n]} x_{ij}t_{ij}$. We use $l_{opt}(t)$ to denote the optimum, and $l_{opt}(t) = \min_x \max_{i\in[m]} \sum_{j\in[n]} x_{ij}t_{ij}$. A mechanism M is called c-approximation mechanism if for any instance t, we have $l_M(t) \leq c \cdot l_{opt}(t)$. For randomized mechanism M, we require $E[l_M(t)] \leq c \cdot l_{opt}(t)$, where the expectation is over the random bits used in the mechanism.

To sum up, we aim at designing (randomized) truthful mechanism with small approximation ratio. By the way, we also require the algorithms of the mechanism to be polynomial computable. When designing a mechanism, there are already several results about the characterization of truthfulness, which may help us to get rid of the payment issue. We mainly use Archer and Tardos' monotone theorem for one parameter mechanism in [1]. In the one parameter case, each player i only has a single value as his/her type (i.e. the speed of machine i). Similar result is obtained in [7] for the auction setting.

Theorem 1. *([7, 1]) In a one parameter scheduling mechanism, an allocation algorithm admits a payment scheme to make the mechanism truthful if and only if it is monotone decreasing. In this case, the mechanism is truthful if and only if the payments $p_i(b_i, b_{-i})$ are of the form*

$$h_i(b_{-i}) + b_i x_i(b_i, b_{-i}) - \int_0^{b_i} x_i(u, b_{-i})\, du$$

where the h_i are arbitrary functions, and x^i are the allocation functions (algorithm).

In this paper, we also consider the lower bound of approximation ratio for a special family of mechanisms, i.e. task independent truthful mechanisms. We first define task independent mechanisms.

Definition 1. *A deterministic mechanism M is task independent, if for any bid matrices b, b' such that $b_{ij} = b'_{ij}$ for any $i \in [m]$, then the allocation of task j does not change, i.e. $x_{ij}(b) = x_{ij}(b')$, $\forall i \in [m]$.*

For randomized mechanisms, there are also two versions of task independence. One is weak task independent randomized mechanism, which is a distribution over several task independent deterministic mechanisms. The other is (strong) task independent randomized mechanism, which satisfies that not only the allocation of task j does not change when j's column of b is not changed, but also all the random variables x_{ij} are independent between different tasks. In this paper, we consider the stronger version.

The following theorem is a main tool used in proving lower bound.

Theorem 2. *(Monotone theorem[8]) In any truthful mechanism, the allocation algorithm must satisfy the following monotone property: for any two bids b and b' which differ only on machine i, the corresponding allocation $x(b)$ and $x' = x(b')$ satisfy*

$$\sum_{j=1}^{m} (x_{ij} - x'_{ij})(b_{ij} - b'_{ij}) \leq 0.$$

We remark that for randomized mechanism, the monotone property of the allocation algorithm still holds, which is proved implicitly in [6]. In our paper, we only use the following corollary for task independent randomized truthful mechanisms.

Corollary 1. *For any task independent randomized truthful mechanism M, any two bid matrices b, b' where b' is obtained from b by only changing b_{ij} to b'_{ij}, then we have $\big(x_{ij}(b) - x_{ij}(b')\big)(b_{ij} - b'_{ij}) \leq 0$, where x_{ij} denotes the probability of assigning task j to machine i.*

3 Scheduling Two Machines

Most of the previous works on this topic are for scheduling two machines. In this section, we first propose a unified framework for all the known mechanisms. Based on this framework, we give an improved truthful mechanism. Then we also explore the limitation of this approach by showing an almost tight lower bound for all the task-independent truthful mechanisms.

3.1 Unified Randomized Truthful Mechanisms $\mathbf{M_f}$

Let $f : R_+ \rightharpoonup [0, 1]$ be a non-decreasing monotone function, satisfying $f(0) = 0$ and $\lim_{x\to\infty} f(x) = 1$. Then we have a randomized mechanism M_f for scheduling two machines based on f. Noticing that this kind of function f can be viewed as a cumulative distribution function for a random variable in R^+, we have the following formal description of the mechanism M_f:

Input: The reported bid matrix b.
Output: A randomized allocation x
and a payment $p = (p_1, p_2)$.
Allocation and Payment Algorithm:
$x_{1j} \leftarrow 0, x_{2j} \leftarrow 0, j = 1, 2 \cdots, n; p_1 \leftarrow 0; p_2 \leftarrow 0.$
For each task $j = 1, 2 \cdots, n$ do
  Choose $s_j \in R_+$ randomly according to function f
  such that $Pr(s_j \le u) = f(u)$.
  if $b_{1j} \le s_j^{-1} b_{2j}$,
    $x_{1j} \leftarrow 1, p_1 \leftarrow p_1 + s_j^{-1} b_{2j};$
  else
    $x_{2j} \leftarrow 1, p_2 \leftarrow p_2 + s_j b_{1j}.$

This unified mechanism M_f is actually a generalization of Nisan and Ronen's Biased MinWork Mechanism in a continuous setting. For the truthfulness, we have the following theorem.

Theorem 3. *For any non-decreasing monotone function $f : R_+ \rightharpoonup [0, 1]$, where $f(0) = 0$ and $\lim_{x\to\infty} f(x) = 1$, mechanism M_f is universally truthful.*

Proof. To prove that the mechanism M_f is universally truthful, we only need to prove that it is truthful when the random sequence $\{s_j\}$ is fixed. Since the utility of an agent equals the sum of the utilities obtained from each task and our mechanism is task-independent, we only need to consider the case of one task. In this case, say s_j is fixed and there is only one task j, the allocation algorithm is monotone decreasing and the payment makes the mechanism truthful, according to the theorem 1(with function $h_1(b_{2j}) = s_j^{-1} b_{2j}$ and $h_2(b_{1j}) = s_j b_{1j}$).

Now we demonstrate the power of our unified designing approach by showing that every known mechanism can be viewed as a mechanism M_f with respect to some function f.

$$f_1(x) = \begin{cases} 1, & x \geq 1, \\ 0, & 0 \leq x < 1; \end{cases} \quad f_2(x) = \begin{cases} 1, & x \geq \frac{4}{3}, \\ \frac{1}{2}, & \frac{3}{4} \leq x < \frac{4}{3}, \\ 0, & 0 \leq x < \frac{3}{4}; \end{cases} \quad f_3(x) = \begin{cases} 1, & x \geq \alpha, \\ r, & \beta \leq x < \alpha, \\ \frac{1}{2}, & \frac{1}{\beta} \leq x < \beta, \\ 1-r, & \frac{1}{\alpha} \leq x < \frac{1}{\beta}, \\ 0, & 0 \leq x < \frac{1}{\alpha}. \end{cases}$$

M_{f_1} is exactly the *Min Work Mechanism* proposed by Nisan and Ronen [8]. This is indeed a deterministic mechanism, whose approximation ratio is 2, and it is the best determinate mechanism. M_{f_2} is the *Biased Min Work Mechanism* also proposed by Nisan and Ronen [8], whose approximation ratio is 1.75. Then we improved their result by M_{f_3} in our previous work[5]. By taking $\alpha = 1.4844, \beta = 1.1854, r = 0.7932$ in f_3, we have a randomized truthful mechanism with approximation ratio of 1.6737.

We can see that all the previous functions f are distribution functions of some discrete random variables. One essential reason is that we can apply a "task reducing" technique [8, 5], then analyze the performance using a case by case method. However the number of subcases increased dramatically if we consider a more complicated function f. One of our main contribution in this paper is that we not only propose the unified framework M_f, but also provide a performance analysis method.

Theorem 4. *For any non-decreasing monotone function* $f : R_+ \rightharpoonup [0, 1]$*, the approximation ratio of the mechanism* M_f *is exactly* $\max_{\alpha_1,\alpha_2 \in R_+} F(\alpha_1, \alpha_2)$*, where* $F : R_+ \times R_+ \rightharpoonup R$ *is defined as following (Here* $r_1 = f(\alpha_1)$ *and* $r_2 = f(1/\alpha_2)$*)*

$$F(\alpha_1, \alpha_2) = (1+\alpha_2)r_1r_2 + r_1(1-r_2) + (1+\alpha_1)(1-r_1)(1-r_2) + max\{\alpha_1, \alpha_2\}r_2(1-r_1).$$

By this theorem, we can easily estimate the approximation ratio of a given mechanism M_f. In particular, by choosing $f(x) = 1 - \frac{1}{2^{x^{2.3}}}$, we can compute that its approximation ratio is 1.5963. The function we used here is only an illustration of our mechanism M_f. We also believe that there exists a better f, though very hard to find. It is also an interesting problem to explore the property of function f with which M_f can have smaller approximation ratio.

Theorem 5. *For* $f(x) = 1 - \frac{1}{2^{x^{2.3}}}$*, the mechanism* M_f *for two machines is universally truthful and can achieve an approximation ratio of* 1.5963*.*

Before we prove the theorem 4, we first give a lemma, which gives an alternative description of the allocation in our mechanism M_f. Its proof is direct from the definition of our mechanism. Since we already proved that our mechanism M_f is truthful, we can also denote the bid as t in the following.

Lemma 1. *For any type matrix t of the two machines, M_f allocates each task independently and for each task $j = 1, 2 \cdots, n$, if $t_{1j} = 0$, always allocate it to machine 1, otherwise allocates it to machine 1 with probability $f(t_{2j}/t_{1j})$ and to machine 2 with probability $1 - f(t_{2j}/t_{1j})$.*

Proof of Theorem 4: Fix any instance $t = (t_{ij})$, let l_{opt} be the optimal makespan. Let O_1, O_2 be the sets of tasks assigned to machine 1 and machine 2 respectively in an optimal solution. Then we have

$$l_{opt} = \max\{\sum_{j \in O_1} t_{1j}, \sum_{k \in O_2} t_{2k}\}.$$

Now we estimate the expected makespan of our mechanism M_f, denoted by l^f. We use l_i^f, $i = 1, 2$, to denote the completion time of machine i, then $l^f = \max\{l_1^f, l_2^f\}$. Let M be a random variable such that $M = 1$ if $l_1^f \geq l_2^f$, and $M = 2$ otherwise. We also denote $Pr(M = 1, x_{1j} = 1)$ as P_j^1 and $Pr(M = 2, x_{2j} = 1)$ as P_j^2 in the following calculation. Then we have:

$$\begin{aligned}
l^f &= \sum_{j \in [m]} t_{1j} P_j^1 + t_{2j} P_j^2 \\
&= \sum_{j \in O_1} t_{1j} \left(P_j^1 + \frac{t_{2j}}{t_{1j}} P_j^2 \right) + \sum_{k \in O_2} t_{2k} \left(\frac{t_{1k}}{t_{2k}} P_k^1 + P_k^2 \right) \\
&\leq \max_{j \in O_1} \left(P_j^1 + \frac{t_{2j}}{t_{1j}} P_j^2 \right) \cdot \sum_{j \in O_1} t_{1j} + \max_{k \in O_2} \left(\frac{t_{1k}}{t_{2k}} P_k^1 + P_k^2 \right) \cdot \sum_{k \in O_2} t_{2k} \\
&\leq l_{opt} \left(\max_{j \in O_1} \left(P_j^1 + \frac{t_{2j}}{t_{1j}} P_j^2 \right) + \max_{k \in O_2} \left(\frac{t_{1k}}{t_{2k}} P_k^1 + P_k^2 \right) \right) \\
&\leq l_{opt} \left(\max_{j \neq k} \left(P_j^1 + \frac{t_{2j}}{t_{1j}} P_j^2 + \frac{t_{1k}}{t_{2k}} P_k^1 + P_k^2 \right) \right)
\end{aligned}$$

So the approximate ratio is bounded by the term

$$\max_{j \neq k} \left(P_j^1 + \frac{t_{2j}}{t_{1j}} P_j^2 + \frac{t_{1k}}{t_{2k}} P_k^1 + P_k^2 \right).$$

Fix any j, k, let $\alpha_1 = \frac{t_{2j}}{t_{1j}}$, $\alpha_2 = \frac{t_{1k}}{t_{2k}}$ and $P_{abc} = Pr(M = a, x_{bj} = 1, x_{ck} = 1)$, $a, b, c \in \{0, 1\}$. Then we can expand P_j^1 as $P_{111} + P_{112}$, since

$$Pr(M = 1, x_{1j} = 1) = Pr(M = 1, x_{1j} = 1, x_{1k} = 1) + Pr(M = 1, x_{1j} = 1, x_{2k} = 1).$$

Let $r_1 = Pr(x_{1j} = 1)$, $r_2 = Pr(x_{1k} = 1)$, then we have:

$$
\begin{aligned}
& P_j^1 + \frac{t_{2j}}{t_{1j}} P_j^2 + \frac{t_{1k}}{t_{2k}} P_k^1 + P_k^2 \\
&= (P_{111} + P_{112}) + \alpha_1(P_{221} + P_{222}) + \alpha_2(P_{111} + P_{121}) + (P_{212} + P_{222}) \\
&= (P_{111} + P_{112} + P_{212}) + \alpha_2(P_{111} + P_{121} + P_{221}) + (\alpha_1 - \alpha_2)P_{221} + (1 + \alpha_1)P_{222} \\
&\le Pr(x_{1j} = 1) + \alpha_2 Pr(x_{1k} = 1) + (\alpha_1 - \alpha_2)Pr(M = 2, x_{2j=1}, x_{1k} = 1) \\
&\qquad + (1 + \alpha_1)Pr(M = 2, x_{2j} = 1, x_{2k} = 1) \\
&\le Pr(x_{1j} = 1) + \alpha_2 Pr(x_{1k} = 1) + \max\{\alpha_1 - \alpha_2, 0\}Pr(x_{2j} = 1, x_{1k} = 1) \\
&\qquad + (1 + \alpha_1)Pr(x_{2j} = 1, x_{2k} = 1) \\
&= (1 + \alpha_2)r_1 r_2 + r_1(1 - r_2) + (1 + \alpha_1)(1 - r_1)(1 - r_2) + max\{\alpha_1, \alpha_2\}r_2(1 - r_1) \\
&= F(\alpha_1, \alpha_2)
\end{aligned}
$$

The first inequality is because $Pr(x_{1j} = 1) = P_{111} + P_{112} + P_{211} + P_{212}$ and so on. The second inequality is because $Pr(M = 2, x_{2j=1}, x_{1k} = 1) \le Pr(x_{2j} = 1, x_{2k} = 1)$. By lemma 1, $r_1 = f(\alpha_1)$, $r_2 = f(1/\alpha_2)$, hence the approximation ratio is bounded by $\max_{\alpha_1, \alpha_2 \in R^+} F(\alpha_1, \alpha_2)$.

On the other direction, we use the following instance to show that our analysis of the approximation ratio is tight. We will use the following tables to illustrated tasks and their allocation throughout this paper. There are two tasks A and B. The left table shows the instance t, where $t_{1A} = 1$, $t_{1B} = \alpha_2$, $t_{2A} = \alpha_1$, $t_{2B} = 1$. The right table shows the allocation of this instance using our mechanism M_f: task A is assigned to machine 1 with probability r_1, to machine 2 with probability $1 - r_1$, etc. Here $r_1 = f(\alpha_1)$ and $r_2 = f(1/\alpha_2)$.

	machine 1	machine 2
task A	1	α_1
task B	α_2	1

$\rightarrow$

	machine 1	machine 2
task A	r_1	$1 - r_1$
task B	r_2	$1 - r_2$

For this instance, we have $l_{opt} \le 1$ and the expected makespan produced by M_f is exactly $F(\alpha_1, \alpha_2)$. So the approximation ratio is at least $F(\alpha_1, \alpha_2)$. □

3.2 Lower Bound for Task Independent Mechanisms

In this section, we show a lower bound for all task independent truthful mechanisms. This lower bound for task independent randomized truthful mechanisms is especially interesting, since a recent work in [3] shows that any truthful mechanism for two machines is task independent, however in the weaker version. So any lower bound better than 1.5 in the weaker version would imply an improvement of the lower bound 1.5 for randomized mechanisms for two machines case.

Theorem 6. *For any task independent truthful mechanism for two machines, its approximation ratio cannot be less than* $11/7 (\approx 1.5714)$.

Proof. Given any task independent truthful mechanism M, consider the following four instances(a is a constant to be specified later, and $a > 1$). We can assume that $r_1 \geq 1/2$, otherwise we relabel the machines in instance 1, and modify the other three instances respectively.

Instance 1:

	machine 1	machine 2
task 1	1	1
task 2	1	2

$\rightarrow$

	machine 1	machine 2
task 1	r_1	$1 - r_1$
task 2	r_2	$1 - r_2$

For this instance, we have $l_M/l_{opt} = 2r_1 + (1 - r_1)r_2 + 3(1 - r_1)(1 - r_2) \geq 1 + r_1 \triangleq A_1$.

Instance 2:

	machine 1	machine 2
task 1	1	1
task 2	1	a

$\rightarrow$

	machine 1	machine 2
task 1	r_1	$1 - r_1$
task 2	r_3	$1 - r_3$

For this instance, we have $l_M/l_{opt} = 2r_1r_3 - r_1 - ar_3 + a + 1 \triangleq A_2$.

Instance 3:

	machine 1	machine 2
task 1	a	a^2
task 2	1	a

$\rightarrow$

	machine 1	machine 2
task 1	r_4	$1 - r_4$
task 2	r_3	$1 - r_3$

For this instance, we have $l_M/l_{opt} = (1 + \frac{1}{a})r_3r_4 - r_3 - ar_4 + a + 1 \triangleq A_3$.

Instance 4:

	machine 1	machine 2
task 1	a	a
task 2	$2a$	a

$\rightarrow$

	machine 1	machine 2
task 1	r_5	$1 - r_5$
task 2	r_6	$1 - r_6$

For this instance, we have $l_M/l_{opt} = 2 - r_5 + 2r_5r_6 \geq 2 - r_5$.

Consider instance 3 and 4, we can change task 2's values in instance 3 to $2a$, a without affecting the allocation of task 1 since M is task independent. Then we decrease machine 2's value on task 1 from a^2 to a. By corollary 1, we know the probability that machine 2 gets task 1 should increase. That is to say, $1 - r_5 \geq 1 - r_4$. Then we have $l_M/l_{opt} \geq 2 - r_4 \triangleq A_4$.

To sum up, mechanism M's approximation ratio is at least $\max\{A_1, A_2, A_3, A_4\}$ with the condition $r_1 \geq 1/2, a > 1$, where $A_1 = 1+r_1$, $A_2 = 2r_1r_3-r_1-ar_3+a+1$, $A_3 = (1 + \frac{1}{a})r_3r_4 - r_3 - ar_4 + a + 1$, $A_4 = 2 - r_4$. Choosing $a = 3/2$ and using a case-by-case analysis, we can prove that $\max\{A_1, A_2, A_3, A_4\} \geq 11/7$ for any r_1, r_2, r_3, r_4 with the assumption $r_1 \geq 1/2$.

4 Scheduling m Machines

First we give the framework of our mechanism for scheduling m machines,BOUNDED-SQUARE mechanism. (Here we only give the allocation algorithm.)

Input: The reported bid matrix $b = (b_{ij})$.
Output: A randomized allocation $X = (X_{ij})$.

Allocation Algorithm:
(1) For each task $j = 1, 2 \cdots, n$ do
let $I_j \leftarrow \{i \in [m] : b_{ij} \leq 2 \min_{i \in [m]} b_{ij}\}$.
if $\min_{i \in [m]} b_{ij} = 0$, we assign task j among the machines in I_j with equal probabilities;
Otherwise we use the SQUARE allocation algorithm[2] in I_j:
For each machine $i = 1, 2 \cdots, m$ do:
if $i \in I_j$, $x_{ij} \leftarrow \frac{\frac{1}{(b_{ij})^2}}{\sum_{s \in I_j} \frac{1}{(b_{sj})^2}}$, otherwise $x_{ij} \leftarrow 0$.
(2) Round (x_{ij}) to a randomized integer solution (X_{ij}) such that $E[X_{ij}] = x_{ij}, \forall i, j$. We will specify the method of rounding later.

In our BOUNDED-SQUARE mechanism, $x = (x_{ij})$ can be viewed as a fractional solution of the scheduling problem. It is adapted from the fractional mechanism SQUARE in [2]. However, we need some "bid condition" in order to bound the loss of performance due to the rounding process. Here we give a threshold of $2 \min_{i \in [m]} b_{ij}$ in the allocation, so if $x_{ij} > 0$, then $b_{ij} \leq 2 \min_{i \in [m]} b_{ij} \leq 2 l_{opt}(b)$. This idea plays an essential role in our mechanism.

Regarding the truthfulness of our mechanism, the proof is based on the fact that the modified fractional mechanism is still truthful. The proof is similar as in [2] and omitted here.

Lemma 2. *For any rounding method satisfying $E[X_{ij}] = x_{ij}, \forall i, j$, there is a payment algorithm to make BOUNDED-SQUARE mechanism truthful in expectation.*

Now we begin to analysis the approximation ratio of our mechanism. Since our mechanism is already proved truthful, we can assume that the players will report their types truthfully, and use t instead of b. Given an instance t, we first show that this fractional solution approximates $l_{opt}(t)$ within a factor of $\frac{m+1}{2}$. The proof is also omitted.

Lemma 3. *Let $x = (x_{ij})$ be the fractional solution obtained in the BOUNDED-SQUARE mechanism, we have $\max_{i \in [m]} \sum_{j \in [n]} x_{ij} t_{ij} \leq \frac{m+1}{2} l_{opt}(t)$.*

For the rounding method, we use the algorithm proposed by Kumar et al. [4].

Lemma 4. *(Kumar et al. [4]) Given a fractional assignment x and a processing time matrix t, there exists a randomized rounding procedure that yields a random integer assignment X such that,*

1. *for any i, j, $E[X_{ij}] = x_{ij}$.*
2. *for any i, $\sum_j X_{ij} t_{ij} < \sum_j x_{ij} t_{ij} + \max_{j : x_{ij} \in (0,1)} t_{ij}$ with probability 1.*

In our mechanism, we already know that $\max_{j : x_{ij} \in (0,1)} t_{ij} \leq 2 l_{opt}(t)$ due to the bid condition. So putting everything together(lemma 2, lemma 3, lemma 4), we have the following theorem.

Theorem 7. *The BOUNDED-SQUARE mechanism is truthful in expectation and has an approximation ratio of $\frac{m+5}{2}$.*

References

1. A. Archer and É. Tardos. Truthful mechanisms for one-parameter agents. In *FOCS '01: Proceedings of the 42nd IEEE symposium on Foundations of Computer Science*, page 482, Washington, DC, USA, 2001. IEEE Computer Society.
2. G. Christodoulou, E. Koutsoupias, and A. Kovács. Mechanism design for fractional scheduling on unrelated machines. In *ICALP*, pages 40–52, 2007.
3. S. Dobzinski and M. Sundararajan. On characterizations of truthful mechanisms for combinatorial auctions and scheduling. In *EC '08: Proceedings of the 9th ACM conference on Electronic commerce*, pages 38–47, New York, NY, USA, 2008. ACM.
4. V. Kumar, M. Marathe, S. Parthasarathy, and A. Srinivasan. Approximation algorithms for scheduling on multiple machines. *Foundations of Computer Science, Annual IEEE Symposium on*, 0:254–263, 2005.
5. P. Lu and C. Yu. An improved randomized truthful mechanism for scheduling unrelated machines. In Susanne Albers and Pascal Weil, editors, *25th International Symposium on Theoretical Aspects of Computer Science (STACS 2008)*, pages 527–538, Dagstuhl, Germany, 2008. Internationales Begegnungs- und Forschungszentrum f"ur Informatik (IBFI), Schloss Dagstuhl, Germany.
6. A. Mu'alem and M. Schapira. Setting lower bounds on truthfulness: extended abstract. In *SODA '07: Proceedings of the eighteenth annual ACM-SIAM symposium on Discrete algorithms*, pages 1143–1152, Philadelphia, PA, USA, 2007. Society for Industrial and Applied Mathematics.
7. R. Myerson. Optimal auction design. *Mathematics of Operations Research*, 6(1):58–73, 1981.
8. N. Nisan and A. Ronen. Algorithmic mechanism design (extended abstract). In *STOC '99: Proceedings of the thirty-first annual ACM symposium on Theory of computing*, pages 129–140, New York, NY, USA, 1999. ACM.