A Lazy Bureaucrat Scheduling Game

Ling Gai
Tianjin University
Department of Mathematics
No. 92 Weijin Road, Nankai District, Tianjin
China
gailing@tju.edu.cn

Yuanchun Song
Tianjin University
Department of Mathematics
No. 92 Weijin Road, Nankai District, Tianjin
China
songyuanchunabc@sina.com

Abstract: In this paper we consider the game theoretical issue of the lazy bureaucrat scheduling problem. There are two players working on a pool of tasks, each of them can select a subset of the tasks to execute and spend the corresponding cost. The common choice would introduce the increasing of the task's cost. Each player has his own budget for these tasks and if the total cost of selected tasks are less than his budget, he can keep the difference part as his "additional" profit. The objective of the players is to make wise selection such that the cost that he spents on the tasks is minimized, while both of them have to obey an assumption called "busy requirement" that as long as there are tasks can be executed by some player (the left budget is more than the cost needed), he must select it to execute. The noncooperative nature and potential interactions between the two players make the problem dynamic and complicated. We prove that Nash equilibrium solutions exist under certain conditions where both players are satisfied with their selection and would not change their mind unilaterally. We also find the method by which we can obtain the Nash equilibrium no matter the player has a single machine or multiple machines to execute on the tasks. Furthermore, we adopt the concept of "price of anarchy" to compare the cost of the worst Nash equilibrium with the social optimum.

Key–Words: Game theory, Nash equilibrium, Price of anarchy, Lazy bureaucrat scheduling

1 Introduction

Lazy bureaucrat scheduling problem was first introduced by Arkin et. al [1] in 1999. Compare to the classical scheduling problem, the lazy bureaucrat scheduling problem take a new look at the scheduling problem from the point of view of the employers who perform the tasks that earn the company its profits. It is natural to expect that some employees may lack the motivation to perform at their peak levels of efficiency, either because they have no stake in the company's profits or because they are simply lazy, so they might try their best to minimize their real working time on scheduling. In this problem, the bureaucrat does not need to execute all of the tasks given, but he has to obey a "busy requirement" rule that as long as the time left before the deadline is enough for some task, he must execute it. This requirement is essential since otherwise the problem would become trivial and the bureaucrat just stay idle without doing any tasks. Lots of results have been presented on several objective functions ([min-makespan], [min-number-of-jobs], [min-time-spent], [min-weighted-sum]) since it was proposed [1, 2, 3, 4, 5, 6]. Arkin et al. proved that the general lazy bureaucrat scheduling problem is NP-hard in the strong sense under all the objective functions and is not approximable to within any fixed factor. Moreover they showed that the problem under all the objective functions becomes weakly NP-hard when the jobs have a common release time. Esfahbod et al. considered the "Common Deadline Lazy Bureaucrat Scheduling Problem" (CD-LBSP) where all tasks' deadlines are the same. They showed that CD-LBSP is weakly NP-hard under all the above four objective functions. It was also proved that the objective function [min-number-of-jobs] (and thus [min-weighted-sum]) is not approximable within any fixed factor unless $P = NP$. Gai et. al. proved that the lazy bureaucrat scheduling problem is still NP-hard even when the tasks are with the same release time and deadline.

Why the problem itself is so hard while still attracts so many attentions in literature? That is because it can be taken as one of the reverse objective combinatorial optimization problems, in which the objective functions are generally opposite to those of the classical one. Other broadly studied reverse objective combinatorial optimization problems include the lazy bin packing problem [3], the lazy bin covering problem [8] and the lazy interval coloring problem [8], the maximum TSP, the maximum cut, and the longest path etc.. It is believed that these inquiries can provide an interesting set of algorithmic questions, which may also lead to better understanding and discovery of the structure and algorithmic complexity of the original optimization problems.

In this paper, we study the lazy bureaucrat scheduling problem through a game-theoretic issue, here bureaucrats may interact with each other in reality when selecting their favorite tasks to execute. From the point of view of an employer, he certainly does not want most of his tasks to be left unexecuted, so he may introduce some punishment on the common choice of the task. Under such condition, a task selected by bureaucrat A to minimize his objective may be not perfect anymore after it is also selected by another bureaucrat. So bureaucrat A will change his mind and make a new selection, and that would cause other changes, too. The problem is so dynamic and complicated that one of our interests is whether or not the Nash equilibrium exist.

We use "player" to denote the lazy bureaucrat, the cost of the task corresponds to the processing time in the scheduling problem, and the player's budget corresponds to the deadline. As the bureaucrats are trying to minimize their working time by choosing "suitable" tasks, here the players are trying to make wise selection to minimize the cost spent on the tasks. The busy requirement still must be obeyed by players, which means as long as the left budget of some player is more than the cost needed of some task, this player must select the task to execute. If some task is selected by several players simultaneously, the cost would increase because of the non-cooperative essence. We introduce a parameter α ($0 < \alpha \leq 1$) to represent this incremental cost in percentage.

Some motivations for studying this problem are given below.

Suppose you get some fund to invest on the projects, you may make choices from a common pool, these projects are costly and independent to each other. Under the motivation of minimizing total cost, you have to make the decision wisely. You may do not want to choose the same projects as others, since these common choices may induce the shortage of material resources or the competition of human resources such that the costs increase; and you may concern to beat other persons by costing less. To make the problem nontrivial, we assume that if the fund left is enough for one of the projects in the pool, you will invest on it.

A real world situation maybe like the following: Government (or some social welfare groups) buy several fixed length time slots from two TV companies to make the public service announcements from a pool of well-designed announcements. Each TV company can get paid from the government if and only if they arrange the announcement as much as possible in every time slot. Meanwhile the TV companies can make tricks such that the left space of each time slot is maximized, which can be used to broadcast other commercial advertisements and get "extra" profits.

This scenario corresponds to the lazy bureaucrat scheduling problem. The length of time slots bought in each TV company corresponds to the machine deadlines in the multiple-machine scheduling case, the number of slots corresponds to the number of machines. Moreover, the length of each announcement corresponds to the processing time of the task. Hence for each TV company, the problem faced becomes the multiple-machine common deadline lazy bureaucrat scheduling problem if the interaction between the two companies are not taken into consideration.

The parameter α can be explained as a compensation to the information transmitting lost. That means if two companies choose too much common announcements, then they are asked by government to supply extra length of time to show other unselected ones. To make it simple, we just lengthen the broadcasting time of the common selected announcement by α percent. For example, a p minutes announcement is chosen by two TV companies simultaneously, then it costs two companies a time interval of $(1 + \alpha)p$ respectively. This parameter clearly introduces more mutual influence between two companies. Maybe one TV company choose an announcement as the last one because after broadcasting it, there is no else announcement can be added in, and the total broadcasting time is minimized. But another TV company may also prefer this announcement and make it as a choice, too. Then this simultaneous choice makes the length larger, which means it is not perfect anymore for one of the companies who will then drop it and make a new choice. This change may incur another moving action again. The choice and the cost of the two companies are dynamic and interactional.

Previous work. A pioneer work was given by Wang et. al. [9, 10] in 2010. They considered a two-person knapsack game, where each player has one knapsack (maybe with different capacity) and tries to maximize the profit of items packed in. The objective function of each player is assumed to be a linear combination of the two players' profits. They proved the existence of Nash equilibrium and studied the price of anarchy for several different objective models.

Our results. In this paper, we study the two player lazy bureaucrat scheduling game. Two players, each with an individual budget, work on a common pool of potential tasks. Both players act in a selfish manner with best-response to optimize their own objective functions by choosing portfolios under the budget restriction. We provide verifiable conditions that guarantee at least one pure Nash equilibrium exists in the lazy bureaucrat scheduling game, where no player can improve the objective by changing individual mind unilaterally. We also give a pseudo polynomial time algorithm to find an equilibrium. To show the sacrifice of system cost because of the selfish and noncooperative behavior of players, we adopt the concept of price of anarchy, which is the ratio of the cost of the worst Nash equilibrium to the system optimum, to quantify the quality of Nash equilibria.

The rest of the paper is organized as follows. The model and notations of the two player lazy bureaucrat scheduling game are given in Section 2. The existence of a Nash equilibrium solution is studied in Section 3. We then develop the pseudo-polynomial time algorithms for finding a Nash equilibrium in Section 4. The concept of price of anarchy is employed for the study of a simplified lazy bureaucrat scheduling game in Section 5. Concluding remarks and future research are included in Section 6.

2 Models and Notations

The lazy bureaucrat scheduling game with two players can be stated formally as follows. There are n tasks need to be executed, each task j has a cost c_j and a processing time p_j for $j = 1, \ldots, n$. Each player i is with a machine deadline d_i for $i = 1, 2$. If a task j is chosen by only one player, it costs the player c_j; if it is chosen by two players, the cost increases a factor of α_i for player i, i.e. the player i has to pay $(1 + \alpha_i)c_j$ for this task, where $0 < \alpha_i \leq 1$. Both players are best response to select a subset of tasks such that the sum of processing time in each machine is no more than his machine deadline, no other task can be added in (busy requirement), and simultaneously optimize his own objective. Without loss of generality, assume c_j, p_j, d_i are all nonnegative and preemption is not allowed.

A state of the lazy bureaucrat scheduling game can be described as an ordered pair of (S_1, S_2), where S_i is the set of tasks selected by player i ($i = 1, 2$).

Let $p(S_i)$ be the total processing time of tasks in set S_i. Then we say a state (S_1, S_2) is feasible if $p(S_i) \leq d_i$ and $d_i - p(S_i) < p_j$ ($\forall j \notin S_i$). For a given feasible state (S_1, S_2), let $C_1(S_1, S_2)$ and $C_2(S_1, S_2)$ denote the costs paid by two players respectively, and $c(S_i)$ denote the total original cost of tasks in set S_i. It is easy to verify that

$$\begin{aligned} C_1(S_1, S_2) = c(S_1) + \alpha_1 c(S_1 \cap S_2) \\ C_2(S_1, S_2) = c(S_2) + \alpha_2 c(S_1 \cap S_2) \end{aligned} \tag{1}$$

In order to present more clearly about the complicated incentive of two players, the objective considered in this paper is the minimization of a linear function of both player's costs. Specifically, let $O_i(S_1, S_2) = \beta_{ii} C_i(S_1, S_2) + \beta_{ij} C_j(S_1, S_2)$ ($j \neq i$) be the objective value of player i at state (S_1, S_2). We assume that $\beta_{ii} \geq 0$ for $i = 1, 2$. In this model, β_{ij} ($j \neq i$) can be negative that means player i's hope is not only to minimize his own cost, but also try to *maximize* the gap between their costs. However we assume that minimizing his own cost is always dominant in his objective, i.e. $|\beta_{ii}| > |\beta_{ij}|$ ($j \neq i$).

Substituting formula (1) into the objectives of players, we have

$$\begin{aligned} O_1(S_1, S_2) = \beta_{11} c(S_1) + \beta_{12} c(S_2) + \Delta_1 c(S_1 \cap S_2) \\ O_2(S_1, S_2) = \beta_{22} c(S_2) + \beta_{21} c(S_1) + \Delta_2 c(S_1 \cap S_2) \end{aligned} \tag{2}$$

where $\Delta_i = \alpha_i \beta_{ii} + \alpha_j \beta_{ij}$, $i \neq j$, for player i.

We say a state of the lazy bureaucrat scheduling game is a Nash equilibrium if it is a feasible state and no player can get better objective function value by resetting his choice unilaterally. This can be defined formally as following:

Definition 1 *A feasible state (S_1, S_2) is a Nash equilibrium if and only if $O_1(S_1, S_2) \leq O_1(S'_1, S_2)$ and $O_2(S_1, S_2) \leq O_2(S_1, S'_2)$ for any feasible states (S'_1, S_2) and (S_1, S'_2).*

3 The Existence of Nash Equilibrium

In this section, we would like to investigate the conditions that ensure the existence of a Nash equilibrium for the lazy bureaucrat scheduling game. The idea of "potential function" is employed.

Definition 2 *The potential function in the lazy bureaucrat scheduling game is a real valued function over the players' feasible states such that its value will decrease strictly if a player shifts to a new state to reduce his objective value.*

In the following, we will show a major existence theorem which completely characterizes the existence of Nash equilibrium of a two-player lazy bureaucrat scheduling game using the product value of Δ_1 and Δ_2. Some technical lemmas are presented first similar as in [9].

Lemma 3 *If $\Delta_1 = \Delta_2 = 0$, then $\Phi_1(S_1, S_2) = c(S_1) + c(S_2)$ is a potential function for the lazy bureaucrat scheduling game.*

Proof: According to $\Delta_1 = \Delta_2 = 0$, we have

$$\begin{aligned} O_1(S_1, S_2) = \beta_{11} c(S_1) + \beta_{12} c(S_2) \\ O_2(S_1, S_2) = \beta_{22} c(S_2) + \beta_{21} c(S_1) \end{aligned}$$

If player 1 can reduce his objective value by shifting to a new feasible state S_1', then we obtain

$$O_1(S_1', S_2) < O_1(S_1, S_2),$$

that is to say,

$$\beta_{11}c(S_1') + \beta_{12}c(S_2) < \beta_{11}c(S_1) + \beta_{12}c(S_2).$$

Consequently, we get $\beta_{11}c(S_1') < \beta_{11}c(S_1)$. As we have assumed that β_{11} is nonnegative, we have $c(S_1') < c(S_1)$, which implies that

$$\Phi_1(S_1', S_2) < \Phi_1(S_1, S_2).$$

Now, if player 2 can reduce his objective value via shifting to a new feasible state S_2', then we have $O_2(S_1, S_2') < O_2(S_1, S_2)$, in other word,

$$\beta_{22}c(S_2') + \beta_{21}c(S_1) < \beta_{22}c(S_2) + \beta_{21}c(S_1).$$

So we obtain $\beta_{22}c(S_2') < \beta_{22}c(S_2)$. Due to the general assumption on $\beta_{22} \geq 0$, we know $c(S_2') < c(S_2)$, which show that

$$\Phi_1(S_1, S_2') < \Phi_1(S_1, S_2).$$

In accordance with Definition 2, we can make a conclusion that $\Phi_1(S_1, S_2) = c(S_1) + c(S_2)$ is a potential function for $\Delta_1 = \Delta_2 = 0$ case. $\square$

Lemma 4 *If* $\Delta_1 = 0$ *and* $\Delta_2 \neq 0$*, define* $\Phi_2(S_1, S_2) = Mc(S_1) + \beta_{22}c(S_2) + \Delta_2 c(S_1 \cap S_2)$*; If* $\Delta_1 \neq 0$ *and* $\Delta_2 = 0$*, define* $\Phi_2'(S_1, S_2) = \beta_{11}c(S_1) + Mc(S_2) + \Delta_1 c(S_1 \cap S_2)$*. Then* Φ_2 *(*Φ_2'*) is a potential function for the lazy bureaucrat scheduling game, where* $M = 1 + 2max\{|\Delta_1|, |\Delta_2|\} \sum_{i=1}^{n} c_i$*.*

Proof: If $\Delta_1 = 0$ and $\Delta_2 \neq 0$, (2) can be written as

$$\begin{aligned} &O_1(S_1, S_2) = \beta_{11}c(S_1) + \beta_{12}c(S_2) \\ &O_2(S_1, S_2) = \beta_{22}c(S_2) + \beta_{21}c(S_1) + \Delta_2 c(S_1 \cap S_2). \end{aligned}$$

If player 1 can drop his objective value by changing to a new feasible state S_1', then we obtain $O_1(S_1', S_2) < O_1(S_1, S_2)$, that is to say,

$$\beta_{11}c(S_1') + \beta_{12}c(S_2) < \beta_{11}c(S_1) + \beta_{12}c(S_2),$$

Consequently, we get $\beta_{11}c(S_1') < \beta_{11}c(S_1)$. Because β_{11} is nonnegative, we have $c(S_1') < c(S_1)$. It might be assumed that $c(S_i)$ is a positive integer, so we have $c(S_1') \leq c(S_1) - 1$.

Notice that

$$\begin{aligned} &\Phi_2(S_1, S_2) = Mc(S_1) + \beta_{22}c(S_2) + \Delta_2 c(S_1 \cap S_2), \\ &\Phi_2(S_1', S_2) = Mc(S_1') + \beta_{22}c(S_2) + \Delta_2 c(S_1' \cap S_2), \end{aligned}$$

where $M = 1 + 2\max\{|\Delta_1|, |\Delta_2|\} \sum_{i=1}^{n} c_i$. We have

$$\begin{aligned} &\Phi_2(S_1, S_2) - \Phi_2(S_1', S_2) \\ &= Mc(S_1) + \beta_{22}c(S_2) + \Delta_2 c(S_1 \cap S_2) \\ &-(Mc(S_1') + \beta_{22}c(S_2) + \Delta_2 c(S_1' \cap S_2)) \\ &= M(c(S_1) - c(S_1')) + \Delta_2(c(S_1 \cap S_2) - c(S_1' \cap S_2)) \\ &\geq M + \Delta_2(c(S_1 \cap S_2) - c(S_1' \cap S_2)) \\ &\geq M - 2|\Delta_2| \sum_{i=1}^{n} c_i > 0, \end{aligned}$$

which implies that $\Phi_2(S_1', S_2) < \Phi_2(S_1, S_2)$.

Now, if player 2 can reduce his objective value via changing to a new feasible state S_2', then we have $O_2(S_1, S_2') < O_2(S_1, S_2)$, in other words,

$$\begin{aligned} &\beta_{22}c(S_2') + \beta_{21}c(S_1) + \Delta_2 c(S_1 \cap S_2') \\ &< \beta_{22}c(S_2) + \beta_{21}c(S_1) + \Delta_2 c(S_1 \cap S_2). \end{aligned}$$

Substract $\beta_{21}c(S_1)$ on the both side of inequality, then it becomes

$$\beta_{22}c(S_2') + \Delta_2 c(S_1 \cap S_2') < \beta_{22}c(S_2) + \Delta_2 c(S_1 \cap S_2).$$

In the meantime,

$$\begin{aligned} &\Phi_2(S_1, S_2) - \Phi_2(S_1, S_2') \\ &= Mc(S_1) + \beta_{22}c(S_2) + \Delta_2 c(S_1 \cap S_2) \\ &-(Mc(S_1) + \beta_{22}c(S_2') + \Delta_2 c(S_1 \cap S_2')) \\ &= \beta_{22}c(S_2) + \Delta_2 c(S_1 \cap S_2) \\ &-(\beta_{22}c(S_2') + \Delta_2 c(S_1 \cap S_2')). \end{aligned}$$

Therefore, we have $\Phi_2(S_1, S_2') < \Phi_2(S_1, S_2)$.

Through the analysis, we can know that $\Phi_2(S_1, S_2)$ is a potential function for this case by Definition 2.

If $\Delta_2 = 0$ and $\Delta_1 \neq 0$, we have

$$\begin{aligned} &O_1(S_1, S_2) = \beta_{11}c(S_1) + \beta_{12}c(S_2) + \Delta_1 c(S_1 \cap S_2), \\ &O_2(S_1, S_2) = \beta_{22}c(S_2) + \beta_{21}c(S_1). \end{aligned}$$

We can prove that $\Phi_2'(S_1, S_2)$ is a potential function by a similar proof. $\square$

Lemma 5 *If* $\Delta_1 > 0$ *and* $\Delta_2 > 0$*, then* $\Phi_3(S_1, S_2) = \Delta_2\beta_{11}c(S_1) + \Delta_1\beta_{22}c(S_2) + \Delta_1\Delta_2 c(S_1 \cap S_2)$ *is a potential function for the lazy bureaucrat scheduling game.*

Proof: Considering $\Delta_1 > 0$ and $\Delta_2 > 0$, we have

$$\begin{aligned} &O_1(S_1, S_2) = \beta_{11}c(S_1) + \beta_{12}c(S_2) + \Delta_1 c(S_1 \cap S_2) \\ &O_2(S_1, S_2) = \beta_{22}c(S_2) + \beta_{21}c(S_1) + \Delta_2 c(S_1 \cap S_2) \end{aligned}$$

If player 1 changes from S_1 to a feasible state S_1' with a reduced objective value, then we have $O_1(S_1', S_2) < O_1(S_1, S_2)$. This further implies that

$$\begin{aligned} &\beta_{11}c(S_1') + \beta_{12}c(S_2) + \Delta_1 c(S_1' \cap S_2) \\ &< \beta_{11}c(S_1) + \beta_{12}c(S_2) + \Delta_1 c(S_1 \cap S_2) \end{aligned}$$

Through setting and analyzing, we show

$$\beta_{11}c(S_1') + \Delta_1 c(S_1' \cap S_2) < \beta_{11}c(S_1) + \Delta_1 c(S_1 \cap S_2).$$

Since $\Delta_2 > 0$, we have

$$\begin{aligned}&\Phi_3(S_1, S_2) - \Phi_3(S_1', S_2)\\ &= \Delta_2(\beta_{11}c(S_1) + \Delta_1 c(S_1 \cap S_2)\\ &-\beta_{11}c(S_1') - \Delta_1 c(S_1' \cap S_2)) > 0.\end{aligned}$$

Therefore, we have $\Phi_3(S_1, S_2) > \Phi_3(S_1', S_2)$. Now, if player 2 changes from S_2 to a feasible state S_2' with a reduced objective value, then we have $O_2(S_1, S_2') < O_2(S_1, S_2)$. This further implies that

$$\begin{aligned}&\beta_{22}c(S_2') + \beta_{21}c(S_1) + \Delta_2 c(S_1 \cap S_2')\\ &< \beta_{22}c(S_2) + \beta_{21}c(S_1) + \Delta_2 c(S_1 \cap S_2)\end{aligned}$$

Minus $\beta_{21}c(S_1)$ on both sides simultaneously, then

$$\beta_{22}c(S_2') + \Delta_2 c(S_1 \cap S_2') < \beta_{22}c(S_2) + \Delta_2 c(S_1 \cap S_2)$$

Since $\Delta_1 > 0$, we have

$$\begin{aligned}&\Phi_3(S_1, S_2) - \Phi_3(S_1, S_2')\\ &= \Delta_1(\beta_{22}c(S_2) + \Delta_2 c(S_1 \cap S_2)\\ &-\beta_{22}c(S_2') - \Delta_2 c(S_1 \cap S_2'))\\ &> 0.\end{aligned}$$

That is, $\Phi_3(S_1, S_2) > \Phi_3(S_1, S_2')$. Hence, according to Definition 2, we know that $\Phi_3(S_1, S_2)$ is a potential function. $\square$

Lemma 6 *If $\Delta_1 < 0$ and $\Delta_2 < 0$, then $\Phi_4(S_1, S_2) = -\Phi_3(S_1, S_2)$ is a potential function for the lazy bureaucrat scheduling game.*

Proof: If player 1 changes from S_1 to a feasible state S_1' with a dropped objective value, then we have $O_1(S_1', S_2) < O_1(S_1, S_2)$. This further implies that

$$\begin{aligned}&\beta_{11}c(S_1') + \beta_{12}c(S_2) + \Delta_1 c(S_1' \cap S_2)\\ &< \beta_{11}c(S_1) + \beta_{12}c(S_2) + \Delta_1 c(S_1 \cap S_2).\end{aligned}$$

That is

$$\beta_{11}c(S_1') + \Delta_1 c(S_1' \cap S_2) < \beta_{11}c(S_1) + \Delta_1 c(S_1 \cap S_2).$$

Since $\Delta_2 < 0$, we have

$$\begin{aligned}&\Phi_4(S_1, S_2) - \Phi_4(S_1', S_2)\\ &= \Delta_2(\beta_{11}c(S_1') + \Delta_1 c(S_1' \cap S_2)\\ &-\beta_{11}c(S_1) - \Delta_1 c(S_1 \cap S_2))\\ &> 0.\end{aligned}$$

That is, $\Phi_4(S_1, S_2) > \Phi_4(S_1', S_2)$.

Now, if player 2 changes from S_2 to a feasible state S_2' with a reduced objective value, then we have $O_2(S_1, S_2') < O_2(S_1, S_2)$. Specifically,

$$\begin{aligned}&\beta_{22}c(S_2') + \beta_{21}c(S_1) + \Delta_2 c(S_1 \cap S_2')\\ &< \beta_{22}c(S_2) + \beta_{21}c(S_1) + \Delta_2 c(S_1 \cap S_2).\end{aligned}$$

Subtract $\beta_{21}c(S_1)$ on both sides simultaneously, then we have

$$\begin{aligned}&\beta_{22}c(S_2') + \Delta_2 c(S_1 \cap S_2')\\ &< \beta_{22}c(S_2) + \Delta_2 c(S_1 \cap S_2).\end{aligned}$$

Since $\Delta_1 < 0$, we have

$$\begin{aligned}&\Phi_4(S_1, S_2) - \Phi_4(S_1, S_2')\\ &= \Delta_1(\beta_{22}c(S_2') + \Delta_2 c(S_1 \cap S_2')\\ &-\beta_{22}c(S_2) - \Delta_2 c(S_1 \cap S_2)) > 0.\end{aligned}$$

After transposition of terms, we obtain $\Phi_4(S_1, S_2) > \Phi_4(S_1, S_2')$. It is easy to see that $\Phi_4(S_1, S_2)$ is a potential function from Definition 2. $\square$

Theorem 7 *In the lazy bureaucrat scheduling game, if $\Delta_1\Delta_2 \geq 0$, then the best-response behavior of players will lead an arbitrary feasible state to a Nash equilibrium; If $\Delta_1\Delta_2 < 0$, then at least one instance without Nash equilibrium exists.*

Proof: Note that the potential function at a Nash equilibrium of the lazy bureaucrat scheduling game may or may not achieve its minimum value. But a feasible state at which the potential function achieves its minimum value must be a Nash equilibrium. From above lemmas, we know that if current state is not a Nash equilibrium, it will switch to a new state with a strictly decreasing potential by the best-response behavior of players. There are only finite possibility of different states, thus the best-response behavior of players will lead an arbitrary feasible state to a Nash equilibrium if $\Delta_1\Delta_2 \geq 0$.

In case $\Delta_1\Delta_2 < 0$. Without loss of generality, we assume that $\Delta_1 > 0$ and $\Delta_2 < 0$. Consider a simple instance as follows. The machine deadlines of two players are d_1 and d_2, where $d_1 < d_2 < 2d_1$. There are two tasks both with the processing time $p_1 = p_2 = d_1$, and the costs are 1 and $1 + \varepsilon$ respectively ($\varepsilon > 0$ is an arbitrarily small number). We can see that each player can choose exactly one task. As $\Delta_2 < 0$, $\alpha_i > 0$, $\beta_{ii} \geq 0$, for $i = 1, 2$, we have $\beta_{21} < 0$. If two players select the same task. Say task 1 with cost 1, then the objectives are as follows:

$$\begin{aligned}O_1(task1, task1) &= \beta_{11} + \beta_{12} + \Delta_1,\\ O_2(task1, task1) &= \beta_{22} + \beta_{21} + \Delta_2.\end{aligned}$$

Player 1 can reduce his cost by changing to choose task 2 because $\Delta_1 > 0$:

$$O_1(task2, task1) = \beta_{11}(1 + \varepsilon) + \beta_{12},$$

while in this state, the cost of player 2 increase because $\Delta_2 < 0$ although ε is arbitrarily small:

$$O_2(task2, task1) = \beta_{22} + \beta_{21}(1 + \varepsilon).$$

He then changes to task 2 immediately to optimize his objective. This will motivate player 1 to choose task 1 again, and the procedure would continue like this without reaching any Nash equilibrium. □

Notice that there may be the case that a Nash equilibrium state is with $\Delta_1\Delta_2 < 0$. In another word, $\Delta_1\Delta_2 \geq 0$ is a sufficient condition for the existence of Nash equilibrium, but it is not a necessary condition. Here we take a look at the following simple example. There are two tasks, the corresponding parameters are as follows: $\beta_{11} = \beta_{22} = \beta_{12} = 1; \beta_{21} = -1; c_1 = 1, c_2 = 3; d_1 = d_2 = 1; p_1 = p_2 = 1; \alpha_1 = \frac{2}{3}, \alpha_2 = \frac{1}{3}$. It is easy to calculate that $\Delta_1 = 1$ and $\Delta_2 = -\frac{1}{3}$, so $\Delta_1\Delta_2 < 0$. While we can verify that the state $(task1, task1)$ is a Nash equilibrium.

Corollary 8 *In case $\Delta_1\Delta_2 \geq 0$, a feasible state with a minimum potential function value is a Nash equilibrium.*

Proof: We prove the corollary by contradiction.

Suppose that a feasible state with a minimum potential function value is not a Nash equilibrium. We assume the potential function can achieve its minimum value at state (S_1, S_2).

Since (S_1, S_2) is not a Nash equilibrium, there must be a feasible state (S_1', S_2) or (S_1, S_2'), such that $O_1(S_1', S_2) < O_1(S_1, S_2)$, or $O_2(S_1, S_2') < O_2(S_1, S_2)$.

Without loss of generality, we may assume that $O_1(S_1', S_2) < O_1(S_1, S_2)$. Then the potential function value at (S_1', S_2) is less than at (S_1, S_2) from the definition of potential function.

It is a contradiction with the minimum potential function value at state (S_1, S_2). □

4 Find Nash Equilibrium

From the key existence theorem, we know that at least one Nash equilibrium solution of a lazy bureaucrat scheduling game exists as long as $\Delta_1\Delta_2 \geq 0$. In this section, we present a dynamic programming based algorithm to find the Nash equilibrium in pseudo polynomial time, which can be extended easily even if each player holds more than one machine.

An important property of potential functions should be introduced in advance:

Definition 9 {**Additive Property**} *A potential function $\Phi(S_1, S_2)$ is additive if $\Phi(S_1, S_2) = \Phi(S_{11}, S_{21}) + \Phi(S_{12}, S_{22}) + \ldots + \Phi(S_{1n}, S_{2n})$ for any given states (S_{11}, S_{21}), (S_{12}, S_{22}), ..., (S_{1n}, S_{2n}) with $S_1 = S_{11} \cup \ldots \cup S_{1n}$, $S_2 = S_{21} \cup \ldots \cup S_{2n}$ and $S_{ik} \cap S_{il} = \emptyset$ for any $k \neq l$, $i = 1, 2$.*

It can be verified that the potential functions Φ_1, Φ_2 (Φ_2'), Φ_3, Φ_4 defined in above lemmas are all additive. This property is very important helping to find a feasible state that achieves the minimum value of its potential function.

4.1 Single-Machine Lazy Bureaucrat Scheduling Game

In this section, we first investigate the Nash equilibrium of the single machine lazy bureaucrat scheduling game, where there is exactly one machine hold by each player to execute on tasks.

For a feasible state (S_1, S_2), we can calculate its potential function value $\Phi(S_1, S_2)$, the sum of processing time $p_1(S_1)$ for player 1, and $p_2(S_2)$ for player 2. We use $F(S_1, S_2) = [S_1, S_2, p_1(S_1), p_2(S_2)]$ to record the information of each state. Following is the dynamic programming based algorithm which looks for a Nash equilibrium solution for the lazy bureaucrat scheduling game with $\Delta_1\Delta_2 \geq 0$:

DP Algorithm-1:

1. Start with $M_0 = \{F(\phi, \phi)\}$.
2. For $k = 1, 2, \ldots, n$, do
   (a) Set $M_k = M_{k-1}$.
   (b) For each $F(S_1, S_2) \in M_{k-1}$, (i) if $p_1(S_1) + p_{1k} \leq d_1$, add $F(S_1 \cup \{k\}, S_2)$ to M_k; (ii) if $p_2(S_2) + p_{2k} \leq d_2$, add $F(S_1, S_2 \cup \{k\})$ to M_k; (iii) if $p_1(S_1) + p_{1k} \leq d_1$ and $p_2(S_2) + p_{2k} \leq d_2$, add $F(S_1 \cup \{k\}, S_2 \cup \{k\})$ to M_k.
3. Check all states in M_n, such that for each state (S_1, S_2), $d_1 - p_1(S_1) < p_{1j}$, $\forall j \notin S_1$, and $d_2 - p_2(S_2) < p_{2j}$, $\forall j \notin S_2$. Delete all other infeasible states.
4. Check M_n to identify pairs of $F(S_1, S_2)$ and $F(S_1', S_2')$ with $p_1(S_1) = p_1(S_1')$ and $p_2(S_2) = p_2(S_2')$. For each such pair, delete the one with a larger potential value.
5. Find a $F(S_1, S_2)$ in M_n such that it has the smallest value in $\Phi(S_1, S_2)$. Output this state (S_1, S_2).

For this algorithm, it is not hard to observe that, with the help of "principal of optimality", the unnecessary states are eliminated though the enumeration of all feasible states is embedded in Steps 2. Since there are only finite possibilities of feasible states, we know that this dynamic programming algorithm will eventually find one with the smallest potential value in Step 5. It must be a Nash equilibrium solution of the game. Adopting the complexity analysis in [9], We can see that Step 1 is a trivial step using $O(1)$ computing time. The main computational effort comes from Step 2. Note that there are n stages. For each stage k, $k = 1, 2 \ldots, n$, since there are at most $(d_1 + 1)(d_2 + 1)$ elements in M_k, this step can be realized in $O(d_1 d_2)$ computing time. Consequently, a total of $O(nd_1 d_2)$ computing time is needed. Moreover, Step 3, 4, 5 only need $O(d_1 d_2)$ computing time for comparisons. Therefore, a Nash equilibrium can be found by the proposed dynamic programming algorithm in $O(nd_1 d_2)$ computing time.

Theorem 10 *For a single-machine lazy bureaucrat scheduling game with $\Delta_1\Delta_2 \geq 0$, a Nash equilibrium can be found by the DP Algorithm-1 in $O(nd_1 d_2)$ computing time.*

4.2 Multiple-Machine Lazy Bureaucrat Scheduling Game

In the multiple-machine lazy bureaucrat scheduling game, we assume that each player has several machines to execute on the tasks, and the cost is the sum of the cost on each machine. The definition of feasibility is same to the single-machine case. Precisely, a state $(S_1, S_2) = (\{S_{11}, \ldots, S_{1m_1}\}, \{S_{21}, \ldots, S_{2m_2}\})$ in multiple-machine case is feasible if for each S_{im_i} ($i = 1, 2$), $p_i(S_{im_i}) \leq d_i$ and $d_i - p_i(S_{im_i}) < p_{ij}$ ($\forall j \notin S_i$).

Since we list all the feasible subsets in DP Algorithm-1 for the single-machine lazy bureaucrat scheduling game and find a Nash equilibrium among them, we can extend the algorithm to the multiple-machine case, too. The key point here is that for each player, he can choose one task at most one time, and the processing time is fixed no matter which machine he arrange it to. So in addition to DP Algorithm-1, we further choose m_1 feasible subsets of S_1, and m_2 feasible subsets of S_2 from M_n, then find the smallest sum of their potential function value. That state corresponds to a Nash equilibrium. The running time is $O(n^m d_1 d_2)$, where $m = max\{m_1, m_2\}$.

DP Algorithm-2:

1. Start with $M_0 = \{F(\phi, \phi)\}$.
2. For $k = 1, 2, \ldots, n$, do
   (a) Set $M_k = M_{k-1}$.
   (b) For each $F(S_1, S_2) \in M_{k-1}$, (i) if $p_1(S_1) + p_{1k} \leq d_1$, add $F(S_1 \cup \{k\}, S_2)$ to M_k; (ii) if $p_2(S_2) + p_{2k} \leq d_2$, add $F(S_1, S_2 \cup \{k\})$ to M_k; (iii) if $p_1(S_1) + p_{1k} \leq d_1$ and $p_2(S_2) + p_{2k} \leq d_2$, add $F(S_1 \cup \{k\}, S_2 \cup \{k\})$ to M_k.
3. Check all states in M_n, such that for each state (S_1, S_2), $d_1 - p_1(S_1) < p_{1j}$, $\forall j \notin S_1$, and $d_2 - p_2(S_2) < p_{2j}$, $\forall j \notin S_2$. Delete all other infeasible states.
4. For the feasible subsets S_1, select all m_1 portfolio of them such that there is no common task inside; Similarly, find all m_2 portfolio of S_2 without any common task inside.
5. Find a $F(S_1, S_2)$ where $S_1 = \{S_{11}, \ldots, S_{1m_1}\}$, $S_2 = \{S_{21}, \ldots, S_{2m_2}\}$ in M_n such that it has the smallest value in $\Phi(S_1, S_2)$. Output this state (S_1, S_2).

Similar to the analysis of Theorem 10, we get the following conclusion.

Theorem 11 *For a multiple-machine lazy bureaucrat scheduling game with $\Delta_1\Delta_2 \geq 0$, a Nash equilibrium can be found by the DP Algorithm-2 in $O(n^m d_1 d_2)$ computing time, $m = max\{m_1, m_2\}$.*

5 Prices of Anarchy

In this section, we will prove the prices of anarchy for a simplified lazy bureaucrat scheduling game, in which the players just pay a fixed percent cost α for the common task instead of α_i, $i = 1, 2$. That is, for a feasible state (S_1, S_2),

$$\begin{aligned} C_1(S_1, S_2) = c(S_1) + \alpha c(S_1 \cap S_2) \\ C_2(S_1, S_2) = c(S_2) + \alpha c(S_1 \cap S_2) \end{aligned} \quad (3)$$

Definition 12 *The price of anarchy of the lazy bureaucrat scheduling game is defined as*

$$\sup_I \frac{\max_{(S_1,S_2)\in N_I} Z(S_1, S_2)}{\min_{(S_1,S_2)\in F_I} Z(S_1, S_2)}$$

where $Z(S_1, S_2)$ is the total cost of two players at state (S_1, S_2). And I is an instance of the game, N_I is the set of all Nash Equilibrium of instance I. F_I is the set of all feasible states of instance I.

In terms of objective, we consider two simple objectives for player i: One is to minimize his own cost $C_i(S_1, S_2)$; The other one is to minimize $C_i(S_1, S_2)-$

$C_j(S_1, S_2)$, $i \neq j$. Which means if $C_i(S_1, S_2) \geq C_j(S_1, S_2)$, player i is trying to minimize his cost comparing with player j; If $C_i(S_1, S_2) < C_j(S_1, S_2)$, player i is trying to maximize the cost gap they have to pay. The corresponding parameters are $\beta_{ii} = 1$, $\beta_{ij} = 0$ and $\beta_{ii} = 1$, $\beta_{ij} = -1$ for $j \neq i$ respectively. Each player decides his/her own objective individually. Considering the symmetry of players, the lazy bureaucrat scheduling game may has three kinds of objective: $(C_1(S_1, S_2), C_2(S_1, S_2))$, $(C_1(S_1, S_2) - C_2(S_1, S_2), C_2(S_1, S_2) - C_1(S_1, S_2))$ and $(C_1(S_1, S_2), C_2(S_1, S_2) - C_1(S_1, S_2))$. We name these three accordingly as: selfish lazy bureaucrat scheduling game, adversary lazy bureaucrat scheduling game and mixed lazy bureaucrat scheduling game. The adversary lazy bureaucrat scheduling game means the competitive between players are so fierce that each of them just wants to let the other one pay much more than himself. For each of these lazy bureaucrat scheduling games, it is easy to verify that $\Delta_i \geq 0$, $i = 1, 2$, and then at least one Nash equilibrium exists by Theorem 7.

We will quantify these Nash equilibria by the price of anarchy, which is the ratio of the worst Nash equilibrium to the social optimum. We assume that (S_1, S_2) is a Nash equilibrium and (S_1^*, S_2^*) is a social optimum state. Let z be the total cost of the two players at a state (S_1, S_2), then $z = C_1(S_1, S_2) + C_2(S_1, S_2) = c(S_1) + c(S_2) + 2\alpha c(S_1 \cap S_2)$. Similarly, let the minimum cost be z^*, thus we have $z^* = C_1(S_1^*, S_2^*) + C_2(S_1^*, S_2^*) = c(S_1^*) + c(S_2^*) + 2\alpha c(S_1^* \cap S_2^*)$. Note that here we can not make the conclusion that there exists a social optimal state where $S_1^* \cap S_2^* = \Phi$ for the definition of feasibility of the lazy bureaucrat scheduling game. For instance, the deadline of two players' machine are both $2 + 2\alpha$, and there are two tasks with the processing time of $p_{11} = p_{21} = p_{12} = p_{22} = 1$, then there is no choice for these two players other than choose both tasks simultaneously.

Theorem 13 *The price of anarchy is $1 + \alpha$ for the selfish lazy bureaucrat scheduling game.*

Proof: Since (S_1, S_2) and (S_1^*, S_2^*) are both feasible, so are (S_1, S_2^*) and (S_1^*, S_2). From the definition of Nash equilibrium and selfish lazy bureaucrat scheduling game, we know that

$$C_1(S_1, S_2) \leq C_1(S_1^*, S_2),$$
$$C_2(S_1, S_2) \leq C_2(S_1, S_2^*)$$

At the same time, noticing that

$$c(S_1^* \cap S_2) \leq c(S_1^*), \quad c(S_1 \cap S_2^*) \leq c(S_2^*)$$

Finally, we have

$$\begin{aligned} z &= C_1(S_1, S_2) + C_2(S_1, S_2) \\ &\leq C_1(S_1^*, S_2) + C_2(S_1, S_2^*) \\ &\quad = c(S_1^*) + c(S_2^*) + \alpha[c(S_1^* \cap S_2) + c(S_1 \cap S_2^*)] \\ &\leq c(S_1^*) + c(S_2^*) + \alpha[c(S_1^*) + c(S_2^*)] \\ &\quad \leq (1 + \alpha) z^* \end{aligned} \tag{4}$$

□

Lower bound 1. Given four tasks whose processing times are the same to two players, that means we can use p_j instead of p_{ij} $(i = 1, 2, j = 1, \ldots, 4)$ here. We know that $c_1 = p_1 = 1 + \varepsilon$; $c_2 = p_2 = 1$; $c_3 = p_3 = \alpha$; $c_4 = 1 + \alpha$, $p_4 = 1 - \alpha$. Let the machine deadline as $d_1 = 1 + \varepsilon$ and $d_2 = 1 + \alpha$ separately. So state $(task1, \{task2, task3\})$ is a Nash equilibrium with the cost of $2 + \alpha + \varepsilon$. While the minimum system cost is from the state of $(task2, task1)$, where the cost is $2 + \varepsilon$. So we get the bound of $1 + \frac{\alpha}{2}$ when $\varepsilon \to 0$.

Theorem 14 *The price of anarchy is $1 + \alpha$ for the adversary lazy bureaucrat scheduling game.*

Proof: It is easy to see that (S_1, S_2^*) and (S_1^*, S_2) are feasible, in the meantime, as state (S_1, S_2) is a Nash equilibrium in the competitive lazy bureaucrat scheduling game, we have

$$C_1(S_1, S_2) - C_2(S_1, S_2) \leq C_1(S_1^*, S_2) - C_2(S_1^*, S_2),$$
$$C_2(S_1, S_2) - C_1(S_1, S_2) \leq C_2(S_1, S_2^*) - C_1(S_1, S_2^*).$$

Then we have $c(S_1) \leq c(S_1^*)$ and $c(S_2) \leq c(S_2^*)$. And also because of

$$c(S_1 \cap S_2) \leq \frac{1}{2}(c(S_1) + c(S_2)),$$

we have

$$\begin{aligned} z &= c(S_1) + c(S_2) + 2\alpha c(S_1 \cap S_2) \\ &\leq c(S_1) + c(S_2) + \alpha(c(S_1) + c(S_2)) \\ &\leq (1 + \alpha)(c(S_1^*) + c(S_2^*)) \\ &\leq (1 + \alpha) z^* \end{aligned} \tag{5}$$

□

Lower bound 2. There are two tasks with the processing time of $p_1 = p_2 = 1$, and the cost as $c_1 = 1$, $c_2 = 1 + \varepsilon$. Let the machine deadline as $d_1 = d_2 = 1$. Then the state $(task1, task1)$ is a Nash equilibrium with $z = 2 + 2\alpha$. While the social optimum state is $(task1, task2)$ with $z^* = 2 + \varepsilon$. So the bound is

$$\frac{z}{z^*} = \frac{2 + 2\alpha}{2 + \varepsilon} \to 1 + \alpha$$

when $\varepsilon \to 0$.

Theorem 15 *The price of anarchy is $z \leq (1+\alpha)z^*$ for the mixed lazy bureaucrat scheduling game.*

Proof: Since (S_1, S_2) and (S_1^*, S_2^*) are both feasible, so it is obviously known that (S_1, S_2^*) and (S_1^*, S_2) are feasible. As state (S_1, S_2) is a Nash equilibrium in the mixed lazy bureaucrat scheduling game, we have

$$C_2(S_1, S_2) - C_1(S_1, S_2) \leq C_2(S_1, S_2^*) - C_1(S_1, S_2^*).$$

Then we get $c(S_2) \leq c(S_2^*)$ from formula (3). And we know that

$$C_2(S_1, S_2) = c(S_2) + \alpha c(S_1 \cap S_2) \leq (1+\alpha)c(S_2),$$

hence

$$C_2(S_1, S_2) \leq (1+\alpha)c(S_2^*).$$

Similarly, in this mixed lazy bureaucrat scheduling game we also have $C_1(S_1, S_2) \leq C_1(S_1^*, S_2)$. It is easy to see that $C_1(S_1^*, S_2) \leq (1+\alpha)c(S_1^*)$, then we have $C_1(S_1, S_2) \leq (1+\alpha)c(S_1^*)$. Therefore,

$$\begin{aligned} z &= C_1(S_1, S_2) + C_2(S_1, S_2) \\ &\leq (1+\alpha)(c(S_1^*) + c(S_2^*)) \\ &\leq (1+\alpha)z^* \end{aligned}$$

□

Lower bound 3. There are two tasks with the processing time as $p_1 = p_2 = 1$; cost as $c_1 = 1$, $c_2 = 1 + \alpha + \varepsilon$. Let the machine deadline as $d_1 = d_2 = 1$. Then the state $(task1, task1)$ is a Nash equilibrium with $z = 2 + 2\alpha$. While the social optimum state is $(task1, task2)$ with $z^* = 2 + \alpha + \varepsilon$. So the bound is

$$\frac{z}{z^*} = \frac{2+2\alpha}{2+\alpha+\varepsilon}$$

6 Remarks

In this paper we consider the lazy bureaucrat scheduling game with two players. We define the corresponding potential functions to prove the existence of a Nash equilibrium solution, and present the pseudo polynomial time algorithm to find such a solution for both of the single machine case and multiple machine case. As for the price of anarchy, we prove that in the adversary scheduling game, the price of anarchy is $1 + \alpha$ with a matched lower bound. And we conjecture that the price of anarchy should be lower in the mixed scheduling game. We may consider more general cases in the future in which both p_i and c_i may change if the tasks are selected simultaneously by both players.

Acknowledgements: The research was supported by the National Natural Science Foundation of China (Grant No. 11201333).

References:

[1] E. M. Arkin, M. A. Bender, J. S. B. Mitchell and S. S. Skiena, The lazy bureaucrat scheduling problem, *Algorithms and Data Structures Symposium,* LNCS 1663, 1999, pp. 122–133.

[2] E. M. Arkin, M. A. Bender, J. S. B. Mitchell and S. S. Skiena, The lazy bureaucrat scheduling problem, *Information and Computation,* Vol. 184, 2003, pp. 129–146.

[3] J. Boyar, L. Epstein, L. M. Favrholdt, J. S. Kohrt, K. S. Larsen, M. M. Pedersen and S. Wøhlk, The maximum resource bin packing problem, *Theoretical Computer Science,* Vol. 362, 2006, pp. 127–139.

[4] B. Esfahbod, M. Ghodsi and A. Sharifi, Common-deadline lazy bureaucrat scheduling problems, *In: Dehne FKHA, Sack J-R Smid MHM (eds). Algorithms and data structures,* LNCS 2748, 2003, pp. 59–66.

[5] L. Gai and G. Zhang, On lazy bureaucrat scheduling with common deadlines, *Journal of Combinatorial Optimization,* Vol. 15, 2008, pp. 191–199.

[6] C. Hepner and C. Stein, Minimizing makespan for the lazy bureaucrat problem, *In: Penttonen, M, Schmidt, E.M. (ed.s). Algorithm Theory.* LNCS 2368, 2002, pp. 40–50.

[7] E. Koutsoupias and C. H. Papadimitriou, Worst-case equilibria, *Proceedings of the 16th Symposium on Theoretical Aspects of Computer Science,* 1999, pp. 404–413.

[8] M. Lin, Y. Yang and J. Xu, On lazy bin covering and packing problems, *Proceedings of COCOON,* LNCS 4112, 2006, pp. 340–349.

[9] Z. Wang, W. Xing and S. Fang, Two-person knapsack game, *Journal of Industrial and Management Optimization,* Vol. 6, 2010, pp. 847–860.

[10] Z. Wang, W. Xing and S. Fang, Two-group knapsack game, *Theoretical Computer Science.* Vol. 411, 2010, pp. 1094–1103.