A Selfish Allocation Heuristic in Scheduling: Equilibrium and Inefficiency Bound Analysis

Jac Braat* H

Herbert Hamers[†] Flip Klijn[‡]

Marco Slikker[§]

August 14, 2018

Abstract

For single decision maker optimization problems that lack time efficient algorithms to determine the optimum, there is a need for heuristics. In the context of coordinated production planning, the seminal paper of Graham (1969) provided a performance analysis of heuristics and obtained a bound in relation to the centralized optimum.

This paper introduces a framework that includes a performance analysis of a so-called equilibrium heuristic in the setting of multiple decision maker problems. The framework consists of three steps: a heuristic for each player that leads to a strategy profile, the verification that this strategy profile is a Nash equilibrium, and finally a worst case cost analysis to obtain a bound on the performance of the heuristic in terms of the aggregate cost in the obtained Nash equilibrium in relation to the centralized optimum.

We implement our general framework in a setting of sequencing situations with selfish agents, multiple identical machines, and the sum of completion times as cost criterion. We provide a tight upper bound for the performance of our equilibrium heuristic. Simulations show that the equilibrium heuristic generally performs much better than the derived tight upper bound. Finally, the relation with the price of anarchy is discussed.

Keywords: Game theory; sequencing situations; outsourcing; equilibrium; heuristic.

^{*}Department of Econometrics & OR and CentER, Tilburg University.

[†]Department of Econometrics & OR, CentER, and TIAS Business School for Society, Tilburg University.

[‡]Institute for Economic Analysis (CSIC), Barcelona GSE, and CentER. The first draft of the paper was written while F. Klijn was visiting CentER and the Department of Econometrics and Operations Research, Tilburg University. He gratefully acknowledges the hospitality of Tilburg University and an extramural fellowship from CentER. He also gratefully acknowledges financial support from AGAUR–Generalitat de Catalunya (2017-SGR-1359), Ministerio de Economía, Industria y Competitividad (ECO2017-88130-P AEI/FEDER, UE), and the Severo Ochoa Programme for Centres of Excellence in R&D (SEV-2015-0563).

[§]School of Industrial Engineering, Eindhoven University of Technology.

1 Introduction

Price-only contracts, which specify a constant selling price per unit, are a well-known cause for double marginalization. Its analysis exemplifies that local optimal decisions can lead to a joint outcome that can be improved upon (see, e.g., Perakis and Roels (2007)). This phenomenon is apparent in a wide variety of operations management settings. In selfish routing it is well-studied and best known for Braess's paradox in which the addition of an extra edge that is seemingly helpful, has a negative effect on all traffic, due to local optimization (see Braess (2005)). Fransoo and Lee (2013) name the coordination of container shipments across the container supply chains as a key industry problem, following up on, e.g., Lange and Chouly (2004) who state that the behavior of different actors involved affect the performance of hinterland supply chains. In competitive outsourcing environments this phenomenon can also be observed. Outsourcing is becoming increasingly important since companies that produce advanced products completely in-house are becoming more and more rare. Hence, apart from managing their own production facilities, companies have an increasing need to tightly control outsourced operations. Specialized suppliers may well be capable of providing high-quality parts that meet all product specifications. Several industries, e.g., electronics, automotive, aerospace, logistics, information technology, include the presence of multiple suitable suppliers, who all possibly serve other clients as well.

There are two important issues that arise in the competitive situations described above. First, an agent faces the problem of finding an appropriate strategy since his costs depend on the strategies of the other agents as well. This dependence on other agents' strategies is not only a complex problem, but also justifies the search for strategies that result in a Nash equilibrium. Second, equilibria are typically inefficient, i.e., the aggregate costs are higher than those in the centralized optimum (also known as the social optimum). This paper introduces a framework that takes both issues into account. The first step is the introduction of a heuristic that for each agent prescribes a strategy. The second step is the verification of whether the resulting strategy profile is a Nash equilibrium. If the heuristics induce a Nash equilibrium, then we call it an equilibrium heuristic. Finally, the third step is a performance analysis of the equilibrium heuristic in terms of the aggregate costs in relation to those in the centralized optimum.

Our approach parallels the performance analysis considered in the seminal paper of Graham (1969) for coordinated production planning. Graham (1969) introduces a heuristic for these single decision maker problems. A performance bound of the heuristic is obtained by worst case costs analysis. For an overview on bounds of performance analyses in scheduling we refer to Graham et al. (1979) and

Pinedo (2002).

In contrast to Graham (1969) we consider multiple decision maker problems. Therefore, in our framework we have a heuristic that is employed by each decision maker and thus leads to a strategy profile. The heuristics become more attractive if at the induced strategy profile no decision maker has an incentive to deviate unilaterally, i.e., cannot be profitably changed by an individual agent. There-fore, we incorporate the concept of Nash equilibrium to check whether the heuristic is an "equilibrium heuristic." Similarly to Graham (1969) a performance bound of the (equilibrium) heuristic is obtained by worst case cost analysis.

We illustrate our framework in a setting of $(M || \sum C_i)$ sequencing situations, i.e., multiple identical machine sequencing situations using as cost criterion the sum of completion times. These sequencing situations are not only interesting from a theoretical perspective to illustrate the implementation of the framework, but have also their own merits in practical environments. In fact, the practical situations that are described by our sequencing model are manyfold. This is due to the increasing trend of outsourcing and subcontracting of supply chain activities (both in manufacturing as well as logistics), in particular over the past two decades. Companies have come to the conclusion that it is better to have the execution of (part of) their supply chain carried out by specialized firms. Through outsourcing a company may obtain additional flexibility, lower its break-even point, or shorten its balance sheet. Typical examples can be found in the semi-conductor industry, in particular in customized chip design and manufacturing, in printed circuit board (sub-)assembly operations or in electronic products and systems assembly operations. Some world-class electronics companies act more as supply chain orchestrators rather than as operators. The level of control outsourcing companies want to have depends on the prime reason for outsourcing and the criticality of the outsourced operation and its resulting components or products. If the prime reason for outsourcing is cost and the resulting components are of a commodity type, the outsourcing company just wants to control the output level, and does not mind so much the sequencing and scheduling details. However, if the prime reason for outsourcing is the access to highly specialized and complex processes, and the resulting products are highly critical for the performance and quality of the end-product, then the outsourcing company probably aims to stay atop of the sequencing and scheduling of the outsourced operation.

More specifically, our model can be related to by the previously mentioned outsourcing activities in the semi-conductor industry. We consider a sequencing situation in which jobs have to be allocated to a number of identical machines. Indeed, it is not at all atypical that semi-conductor and electronics assembly situations have a number of identical machines in operation. Moreover, in order to mitigate supply chain risks, outsourcing companies tend to no longer have single-sourcing policies but apply dual or even multiple sourcing strategies. They outsource to multiple companies that are preferably located in different geographical, economic, and political regions, and that can all produce the required components or products at the required specification levels. In our setting, multiple outsourcing companies are assumed to draw upon a common capacity source of the identical machines. Indeed this is typical for the industry situations described above. Because of the highly specialized level of technology and the required economies of scale, the contract manufacturing companies almost always produce for multiple clients. The paper presents the supply chain situation as a typical area for application of the model developed. How this application works out specifically and in all detail, needs to be analyzed further as that depends also on the specific supply chain configuration (number of outsourcing companies, number of service providers used, characteristics of the machines and processes, etc.), as well as on the contractual and operational arrangements between outsourcing companies on the one hand and their supply chain service providers on the other.

We assume jobs are owned by selfish agents. Each job is characterized by its processing time, i.e., the time any machine takes to handle the job. The cost of a job is determined by its completion time and the cost criterion of an agent/player is the sum of completion times of his jobs. A set of jobs that is assigned to a machine is processed in such a way that the total costs are minimal. This implies that jobs are ordered in increasing processing time on each machine. A strategy of a player is an allocation of his jobs to the machines. There is no straightforward way for a player to determine an appropriate strategy due to the strategic interaction with the other players.

To implement our framework for $(M||\sum C_i)$ sequencing situations, we first introduce the so-called selfish allocation heuristic. For each player separately the heuristic probabilistically assigns the jobs to the machines in such a way that the resulting schedule would be optimal for this player in case there were no jobs of other players. So, if there is a unique player, the selfish allocation heuristic yields an optimal allocation. However, the selfish allocation heuristic is employed by all players simultaneously and independently. Hence, any machine typically receives jobs from different players. Second, we show that the selfish allocation heuristics lead to a strategy profile that is indeed a Nash equilibrium. Finally, we provide a performance bound of the selfish allocation heuristic through a worst case cost analysis. In particular, we obtain a tight bound that only depends on the number of machines.

Next, we discuss the literature that is closely related to our paper and that studies the efficiency loss due to competition in sequencing situations. The focus in these papers, however, is either on the price of anarchy or on the decentralization cost. The price of anarchy is the ratio between the cost of the worst Nash equilibrium (or, equivalently, the Nash equilibrium with highest cost) and the cost attained at the centralized optimum, and has been studied in parallel to the decentralized cost (which is determined by the best Nash equilibrium). Koutsoupias and Papadimitriou (1999) introduce the price of anarchy and show that the price of anarchy is at most $\frac{3}{2}$ if there are two identical machines and the objective criterion is the makespan. Immorlica et al. (2009) consider general multiple machine scheduling situations with objective criterion the makespan. They provide mechanisms that minimize the price of anarchy. Bounds for the price of anarchy in scheduling situations with the weighted completion time criterion are provided in Correa and Queyranne (2012) and Cole et al. (2015). For sequencing situations with the minsum objective, i.e., the sum of completion times, bounds for the price of anarchy are provided in Hoeksma and Uetz (2012). Lee et al. (2012) analyze the price of anarchy for various objective criterions as well as various ordering policies per machine. For a setting in which processing times increase as long as they are not processed, an analysis of the price of anarchy for various ordering policies per machine and several objective criterions can be found in Chen et al. (2017). Bukchin and Hanany (2007) study the decentralization cost of a dispatchingsequencing problem. Each decision maker has a set of jobs that can be processed either in-house. which is less costly, or can be sent to a subcontractor, which is more costly. They provide bounds for the decentralization cost for an arbitrary number of jobs and decision makers. Moreover, they introduce a scheduling-based coordinating mechanism such that the centralized optimum is obtained in a Nash equilibrium. Bukchin and Hanany (2011) consider a decentralized job shop scheduling system. They analyze the bounds of the decentralization cost and propose a mechanism to reduce this cost.

In contrast to our approach, the Nash equilibria in the papers mentioned above are not the result of a heuristic approach. Moreover, the Nash equilibria considered in these papers are the Nash equilibrium with lowest cost or the Nash equilibria with highest cost. This is not a requirement for the Nash equilibrium that results from the selfish allocation heuristic. Finally, the models studied in the abovementioned papers differ in at least one crucial aspect from our model. The difference is in the cost criterion (sum of completion times vs. make span as in Koutsoupias and Papadimitriou (1999) and Immorlica et al. (2009)), or the requirement that jobs have to be processed on all machines (cf. Correa and Queyranne (2012) and Cole et al. (2015)) whereas in our case each job has to be processed on one machine, or the assumption that each player owns exactly one job (cf. Hoeksma and Uetz (2012), Lee et al. (2012), and Chen et al. (2017)) whereas we assume that a player may own a set of jobs. Additionally, Lee et al. (2012), and Chen et al. (2017) consider pure Nash equilibria, whereas we focus on mixed Nash equilibria as pure Nash equilibria need not exist in our setting. Bukchin and Hanany (2007) consider a setting in which they allow the possibility of different processing times on different machines, but they consider two machines only, with the costs at the subcontractor for a job being fixed, implying it is independent of strategies of other players. Bukchin and Hanany (2011) study jobshops and flowshops and restrict attention to two machines as well.

The remainder of the paper is organized as follows. In Section 2, the sequencing model is formally described and explained by some illustrative examples. In Section 3, we present the framework that consists of an equilibrium heuristic and a performance analysis, which provides a tight bound for the performance of the equilibrium heuristic. In Section 4 we report on a series of simulations which show that the performance of the selfish equilibrium heuristic is typically much better than the bound. In Section 5, we show that in our particular setting the selfish equilibrium heuristic generates the worst equilibrium. Finally, Section 6 concludes and contains some suggestions for further research.

2 Sequencing model

Our sequencing model is inspired by the previously discussed outsourcing activities in the semiconductor industry. Let $M = \{1, \ldots, |M|\}$, $|M| \ge 2$, be the finite set of (identical) production entities, which we refer to as machines. Let N be the finite set of outsourcing companies, which we refer to as agents. Let J_i be the set of orders, or shortly jobs, owned by agent i. We assume that for all $i, j \in N$ with $i \ne j$, $J_i \cap J_j = \emptyset$. Let $J = \bigcup_i J_i$. We assume each job has non-preemptive processing requirements, i.e., once it is started it cannot be interrupted. Each job $j \in J$ has processing time $p_j > 0$. To avoid degenerate situations that require cumbersome notation we assume that for $j, j' \in J$ with $j \ne j'$, $p_j \ne p_{j'}$. A (scheduling) problem is a quadruple $\Lambda = (M, N, (J_i)_{i \in N}, (p_j)_{j \in J})$. Whenever there is no possible confusion we omit the set of machines M from the specification of the scheduling problem.

An allocation is an assignment of the jobs to the machines. Formally, a (deterministic) allocation is a function $a : J \to M$, where a(j) indicates the machine on which job j is allocated and hence to be processed. Let A be the set of allocations. A schedule is a sequenced assignment of the jobs to the machines. Formally, a (deterministic) schedule is a function $\sigma : J \to M \times \{1, \ldots, |J|\}$, where $\sigma(j) = (\sigma_1(j), \sigma_2(j)) = (m, k)$ indicates that job j is scheduled in position k of machine m. We assume that on each machine there is no idle time between jobs nor before the first job. Given a schedule σ , job j's predecessors are the jobs $P(\sigma, j) = \{j' \in J : \sigma_1(j') = \sigma_1(j) \text{ and } \sigma_2(j') < \sigma_2(j)\}$. Then, job j's completion time can be written as the sum of its processing time and the waiting time due to its predecessors, i.e.,

$$C_j(\sigma) = p_j + \left(\sum_{j' \in P(\sigma,j)} p_{j'}\right).$$

Each agent determines which of his jobs are processed on which machine. In other words, each

agent *i* chooses an allocation of its jobs $a_i : J_i \to M$, where $a_i(j)$ indicates the machine on which job j is to be processed. Then, the resulting allocation is $a : J \to M$ with $a(j) = a_i(j)$ for each agent $i \in N$ and each job $j \in J_i$.

The central objective is to minimize the sum of completion times respecting the chosen allocation. Let a be an allocation. A schedule σ respects allocation a if for each $j \in J$, $\sigma_1(j) = a(j)$, i.e., schedule σ assigns each job to the same machine as allocation a. A schedule σ' is a-optimal if it respects a and for any other schedule σ that respects a,

$$\sum_{j \in J} C_j(\sigma') \le \sum_{j \in J} C_j(\sigma),$$

i.e., among all schedules that respect a, the sum of all completion times is minimized by schedule σ' . Since allocation a determines the assignment of the jobs to the machines, a schedule that respects a is a-optimal if and only if the jobs on each machine are scheduled in optimal order. Then, given that the central objective is to minimize the sum of completion times, a schedule that respects a is a-optimal if and only if for each machine m its jobs $a^{-1}(m)$ are scheduled in order of shortest processing time (SPT), see e.g., Smith (1956).

A schedule σ^* is optimal if for any other schedule σ ,

$$\sum_{j \in J} C_j(\sigma^*) \le \sum_{j \in J} C_j(\sigma),$$

i.e., the sum of all completion times is minimized by schedule σ^* . The following algorithm can be used to find all optimal schedules.

Minimum Mean Flow Time¹ (MFT) algorithm. (Horowitz and Sahni, 1976)

Step 1. For each machine m, set $l_m \equiv 0$. Set $J^* \equiv J$. As long as $J^* \neq \emptyset$, do Procedure.

BEGIN PROCEDURE.

Let $j^* \in J^*$ be such that $p_{j^*} > p_j$ for all $j \in J^*$, $j \neq j^*$. Let $m \in M$ be a machine with lowest l_m . Set $a^*(j^*) \equiv m$ and update $l_m \equiv l_m + 1$ as well as $J^* \equiv J^* \setminus \{j^*\}$.

Step 2. Let σ^* be an a^* -optimal schedule.

 \diamond

We recall and state the following result for later reference.

Theorem 1. [Horowitz and Sahni, 1976] A schedule is optimal if and only if it can be obtained from the MFT algorithm.

¹Minimum mean flow time and minimum sum of completion times are equivalent objectives.

We associate with each scheduling problem $(N, (J_i)_{i \in N}, (p_j)_{j \in J})$ a (non-cooperative) scheduling game $\Gamma = (N, (A_i)_{i \in N}, (c_i)_{i \in N})$, which is explained next. The set of players is given by N. The set of (pure) strategies of player $i \in N$, denoted A_i , is the collection of functions $a_i : J_i \to M$. With a slight abuse of notation, we associate with each strategy profile $a = (a_i)_{i \in N}$ the allocation $a : J \to M$ with $a(j) = a_i(j)$ for any $i \in N$ and $j \in J_i$. Since all processing times are distinct, a strategy profile (or equivalently, allocation) a induces² a unique a-optimal schedule, which henceforth we will denote by σ^a . Player *i*'s resulting "costs" are given by the sum of completion times of his jobs in σ^a . In other words, player *i*'s cost function c_i is given by

$$c_i(a) = \sum_{j \in J_i} C_j(\sigma^a).$$

In the next example we illustrate the model and some of the previously introduced concepts.

Example 1.

Let $M = \{1, 2\}$ and $N = \{1, 2\}$. Let $J_1 = \{\alpha, \gamma\}$ and $J_2 = \{\beta, \delta\}$. Suppose $(p_\alpha, p_\beta, p_\gamma, p_\delta) = (1, 2, 3, 4)$. Each player has 4 pure strategies regarding his 2 jobs: he can send both jobs to machine 1, both jobs to machine 2, or different jobs to different machines (two ways). Table 1 concisely depicts the scheduling game. Player 1 is the row player and each row indicates which jobs are sent to machine 1 (the complement is sent to machine 2). For instance, $\{\alpha\}$ corresponds with player 1's strategy a_1 with $a_1(\alpha) = 1$ and $a_1(\gamma) = 2$. Similarly, player 2 is the column player and each column indicates which jobs are sent to machine 1. Next, we illustrate that each pair of numbers indicates the costs induced by the corresponding strategy-profile. Consider, for instance, the pair $(\{\alpha\}, \{\beta, \delta\})$, which corresponds with profile $a = (a_i)_{i=1,2}$ such that $a_1(\alpha) = a_2(\beta) = a_2(\delta) = 1$ and $a_1(\gamma) = 2$. Then, jobs α, β , and δ end up together on machine 1 and job γ on machine 2. The unique *a*-optimal schedule σ^a satisfies $\sigma^a(\alpha) = (1,1), \ \sigma^a(\beta) = (1,2), \ \sigma^a(\gamma) = (2,1), \ \text{and} \ \sigma^a(\delta) = (1,3).$ Recall that the first coordinate indicates the machine and the second coordinate indicates the position at that machine. So, machine 1 processes first α , then β , and finally δ ; machine 2 only processes job γ . Then, player 1's costs equal the sum of the completion times of his jobs α and γ : $C_{\alpha}(\sigma^a) + C_{\gamma}(\sigma^a) = 1 + 3 = 4$. Similarly, player 2's costs equal the sum of the completion times of his jobs β and δ : $C_{\beta}(\sigma^{a}) + C_{\delta}(\sigma^{a}) = (1+2) + (1+2+4) = 10.$ Hence, in this case the costs of the two players are given by (4, 10).

By applying Theorem 1 it follows that a schedule is optimal if and only if jobs δ and γ are processed in the second position of different machines and jobs α and β are processed in the first position of different machines. Hence, there are four optimal schedules and one easily verifies that their (minimal) associated cost is 13.

 $^{^2\}mathrm{By}$ SPT reordering on each machine.

$1\backslash 2$	Ø	$\{\beta\}$	$\{\delta\}$	$\{eta,\delta\}$
Ø	7,13	$5,\!10$	7, 7	5,8
$\{\alpha\}$	6,11	4 ,10	6, 7	4 ,10
$\{\gamma\}$	4,10	6,7	4 ,10	$6,\!11$
$\{\alpha,\gamma\}$	5,8	7, 7	5,10	7,13

Table 1: Table of Example 1

Finally, the boldfaced numbers in Table 1 are related to the concept of Nash equilibrium, which will be illustrated in Example 2. \diamond

Let $i \in N$. A mixed strategy \tilde{a}_i of player i is a probability distribution over all pure strategies $a_i \in A_i$. At mixed strategy \tilde{a}_i , let $Pr(a_i|\tilde{a}_i)$ be the probability assigned to pure strategy $a_i \in A_i$. Let $\tilde{a} = (\tilde{a}_i)_{i \in N}$ be a profile of mixed strategies. For any deterministic allocation $a \in A$, let $Pr(a|\tilde{a})$ be the probability of allocation a induced by \tilde{a} , i.e.,

$$Pr(a|\tilde{a}) = \prod_{i \in N} Pr(a_i|\tilde{a}_i).$$
(1)

Denoting the expected completion time of $j \in J$ by

$$\tilde{C}_j(\sigma^{\tilde{a}}) = \sum_{a \in A} Pr(a|\tilde{a}) C_j(\sigma^a),$$

we can write player i's expected "costs" as

$$\tilde{c}_i(\tilde{a}) = \sum_{j \in J_i} \tilde{C}_j(\sigma^{\tilde{a}}) = \sum_{j \in J_i} \left(\sum_{a \in A} \Pr(a|\tilde{a}) C_j(\sigma^a) \right).$$
(2)

A profile of mixed strategies is a Nash equilibrium if no player has a profitable deviation, i.e., any different strategy would not reduce his costs. Formally, a profile of mixed strategies \tilde{a} is a (Nash) equilibrium if there is no player $i' \in N$ with a strategy $\tilde{a}'_{i'}$ such that

 $\tilde{c}_{i'}(\tilde{a}') < \tilde{c}_{i'}(\tilde{a}),$

where $\tilde{a}' = (\tilde{a}'_{i'}, (\tilde{a}_i)_{i \neq i'})$. Let $\mathcal{E}(\Gamma)$ be the set of Nash equilibria of game Γ . Set $\mathcal{E}(\Gamma)$ is nonempty as all players have a finite number of pure strategies in Γ (cf. Nash (1951)).

The following example shows why we consider mixed strategies: not all games have a Nash equilibrium in pure strategies.

Example 2. (No Nash equilibrium in pure strategies.)

Consider again the scheduling game discussed in Example 1. The boldfaced numbers in Table 1

indicate each player's best strategy (best response) given any of the other player's strategies. For instance, player 1's strategy $\{\gamma\}$ is the unique best response to player 2's strategy $\{\delta\}$ since any other strategy of player 1 yields higher costs for player 1: \emptyset gives costs 7, $\{\alpha\}$ gives costs 6, and $\{\alpha, \gamma\}$ gives costs 5, but $\{\gamma\}$ gives costs 4. For this reason, 4 is the unique boldfaced number in column $\{\delta\}$.

Although $\{\gamma\}$ is a best response to $\{\delta\}$, strategy-profile $(\{\gamma\}, \{\delta\})$ is not a Nash equilibrium. The reason is that player 2 has a profitable deviation: player 2's cost is 10 but by switching to strategy $\{\beta\}$ he would have costs of only 7. (Note in fact that $\{\beta\}$ is a best response to $\{\gamma\}$ and hence its associated payoff of 7 is boldfaced in row $\{\gamma\}$.) It is easy to verify that also in any other strategy-profile some player has a profitable deviation. Hence, there is no Nash equilibrium in pure strategies.

There is a unique Nash equilibrium in mixed strategies. To see this, note that for player 1 it is always strictly better to play { α } than \emptyset , and it is always strictly better to play { γ } than { α, γ }. Similarly, for player 2 it is always strictly better to play { β } than \emptyset , and it is always strictly better to play { δ } than { β, δ }. Therefore, in any Nash equilibrium, strategies \emptyset (for both players), { α, γ } (for player 1), and { β, δ } (for player 2) receive probability 0. Applying standard game-theoretic tools (see, e.g., Osborne (2004)), one can easily show that the strategy-profile in which each of the remaining strategies receives probability 0.5 constitutes the unique Nash equilibrium \tilde{a} in mixed strategies. Formally, $\tilde{a} = (\tilde{a}_i)_{i=1,2}$ consists of a probability distribution \tilde{a}_1 over player 1's pure strategies and a probability distribution \tilde{a}_2 over player 2's pure strategies. Here, player 1 assigns probability 0.5 to his strategy a_1^1 defined by $a_1^1(\alpha) = 1$ and $a_1^1(\gamma) = 2$, and also probability 0.5 to his strategy a_1^2 defined by $a_1^2(\alpha) = 2$ and $a_1^2(\gamma) = 1$. Similarly, player 2 assigns probability 0.5 to his strategy a_2^2 defined by $a_2^1(\beta) = 1$ and $a_2^1(\delta) = 2$, and also probability 0.5 to his strategy $a_2^2(\beta) = 2$ and $a_2^2(\delta) = 1$.³

To calculate the expected costs of the unique Nash equilibrium \tilde{a} we first compute the probability of each deterministic allocation $a \in A$. From (1), it follows that for k, l = 1, 2,

$$Pr((a_1^k, a_2^l) | \tilde{a}) = Pr(a_1^k | \tilde{a}_1) \times Pr(a_2^l | \tilde{a}_2) = 0.5 \times 0.5 = 0.25.$$
(3)

Obviously, if for all $k, l = 1, 2, a \neq (a_1^k, a_2^l)$, then $Pr(a|\tilde{a}) = 0$.

³To see this, note that player 2 is indifferent between $\{\beta\}$ and $\{\delta\}$ if and only if player 1 plays each of strategies $\{\alpha\}$ and $\{\gamma\}$ with probability 0.5 while player 1 is indifferent between $\{\alpha\}$ and $\{\gamma\}$ if and only if player 2 plays each of strategies $\{\beta\}$ and $\{\delta\}$ with probability 0.5.

Now, from (2) and (3) the expected costs of player 1 equal

$$\tilde{c}_{1}(\tilde{a}) = \sum_{j=\alpha,\gamma} \tilde{C}_{j}(\sigma^{\tilde{a}}) = \sum_{j=\alpha,\gamma} \left(\sum_{a \in A} Pr(a|\tilde{a}) C_{j}(\sigma^{a}) \right)$$
$$= \sum_{j=\alpha,\gamma} \left(\sum_{k,l=1,2} 0.25 \times C_{j}(\sigma^{(a_{1}^{k},a_{2}^{l})}) \right)$$
$$= 0.25 \sum_{k,l=1,2} \left(\sum_{j=\alpha,\gamma} C_{j}(\sigma^{(a_{1}^{k},a_{2}^{l})}) \right)$$
$$= 0.25 \times (4+6+6+4) = 5,$$

where the penultimate equality follows from Table 1. Similar calculations give player 2's expected costs $\tilde{c}_2(\tilde{a}) = 8.5$. Hence, the cost induced by the unique Nash equilibrium is $\tilde{c}_1(\tilde{a}) + \tilde{c}_2(\tilde{a}) = 13.5$, while the optimal (centralized) cost is 13 (see Example 1).

Example 2 shows that Nash equilibria in pure strategies need not exist and that in a mixed Nash equilibrium there can be a performance loss with respect to the situation in which there would be a central authority.

3 Equilibrium heuristic and its performance

In this section we will formally describe the framework that consists of an equilibrium heuristic and a performance analysis for the sequencing situations introduced in the previous section.

The general structure of the framework, that can be applied to any situation with selfish players, can be described as follows. The first part of the equilibrium heuristic consists of a heuristic for each player which leads to a strategy profile. In the second part it has to be verified that this strategy profile is a Nash equilibrium. The quality of the equilibrium heuristic is evaluated by executing a worst case cost analysis.

Now, we will describe formally the framework for the sequencing situation. The first step is the introduction of the selfish allocation heuristic.

Selfish Allocation (SA) Heuristic (for player $i \in N$)

Step 1. For each machine m, set $l_m \equiv 0$. Set $J^* \equiv J_i$. As long as $J^* \neq \emptyset$, do Procedure.

BEGIN PROCEDURE.

Let $j^* \in J^*$ be such that $p_{j^*} > p_j$ for all $j \in J^*$, $j \neq j^*$. Let $m \in M$ be a machine with lowest l_m . Set $a_i^{SA}(j^*) \equiv m$ and update $l_m \equiv l_m + 1$ as well as $J^* \equiv J^* \setminus \{j^*\}$. END PROCEDURE. **Step 2.** Let \tilde{a}_i^{SA} be the probability distribution over A_i that assigns equal probability to any $a_i \in A_i$ that for all $j, j' \in J_i$ satisfies

$$a_i(j) = a_i(j') \iff a_i^{SA}(j) = a_i^{SA}(j'),$$

and 0 otherwise.

Note that the probability distribution obtained in Step 2 of the SA heuristic only gives (the same) strictly positive probability to all allocations that can be obtained from a_i^{SA} by permutation of the machines. Example 3 illustrates the two steps of the SA heuristic.

Remark 1. Since the procedure in the SA heuristic coincides with the procedure in the MFT algorithm, it follows from Theorem 1 that schedule $\sigma^{a_i^{SA}}$ is optimal for $(\{1\}, J_1, (p_j)_{j \in J_1})$.

Example 3. (Illustration of the SA heuristic.)

Assume |M| = 3. Suppose player 1 has 7 jobs and $J_1 = \{1, 2, ..., 7\}$. Suppose that $(p_1, p_2, ..., p_7) = (1, 2, ..., 7)$. It is easy to check that $a_1^{SA} : J_1 \to M$ defined by

$$a_1^{SA}(3) = a_1^{SA}(7) = 1$$
$$a_1^{SA}(4) = a_1^{SA}(5) = 2$$
$$a_1^{SA}(1) = a_1^{SA}(2) = a_1^{SA}(6) = 3$$

can be obtained in the procedure in the SA heuristic. The corresponding probability distribution \tilde{a}_i^{SA} from the SA heuristic assigns probability $\frac{1}{6}$ to the allocation a_1^{SA} and the 5 other allocations of its jobs that can be obtained by permuting the three batches of jobs on the machines. For instance, \tilde{a}_i^{SA} assigns probability $\frac{1}{6}$ to the allocation $a'_1: J_1 \to M$ defined by

$$a'_1(1) = a'_1(2) = a'_1(6) = 1$$

 $a'_1(3) = a'_1(7) = 2$
 $a'_1(4) = a'_1(5) = 3.$

 \diamond

We now show that the profile $\tilde{a}^{SA} = (\tilde{a}_i^{SA})_{i \in N}$ is a Nash equilibrium. Since \tilde{a}^{SA} is obtained from a heuristic, we will call it an equilibrium heuristic.

Theorem 2. Let Λ be a scheduling problem. Let Γ be its associated scheduling game. Then, the profile obtained from the selfish allocation heuristic constitutes a Nash equilibrium of Γ . That is, $\tilde{a}^{SA} \in \mathcal{E}(\Gamma)$.

The proof of Theorem 2 and all other proofs are relegated to the Appendix.

~	
· · ·	
~	

Remark 2. Any allocation a_i^{SA} that can be generated in the SA heuristic leads to a possibly different mixed strategy \tilde{a}_i^{SA} . However, any combination of such allocations for all players $(\tilde{a}_i^{SA})_{i\in N}$ is, by Theorem 2, a Nash equilibrium of the scheduling game. Even though different combinations give possibly different Nash equilibria, for each player all these Nash equilibria yield the same associated expected costs. The proof is relegated to the end of the Appendix.

Next, we study the performance of the equilibrium heuristic. More precisely, we determine the loss of efficiency of the equilibrium heuristic with respect to the optimal schedules which could be implemented by a central authority. It is convenient to introduce the ratio between the total cost obtained in the equilibrium heuristic and the minimal total cost (obtained in case there would be a central authority). For any scheduling game Γ , define

$$Price(\Gamma, SA) = \frac{\sum_{j \in J} \tilde{C}_j(\sigma^{\tilde{a}^{SA}})}{\sum_{j \in J} C_j(\sigma^*)},$$

where σ^* is an optimal schedule. For any scheduling game Γ , $Price(\Gamma, SA) \geq 1$. Small values of $Price(\Gamma, SA)$ indicate that the equilibrium heuristic performs relatively well in comparison to the minimal total cost that a central authority could achieve. Our aim is to provide an upper bound for *Price*.

The next theorem gives a tight bound for the performance of the equilibrium heuristic.

Theorem 3. Let Λ be a scheduling problem. Let Γ be its associated scheduling game. Then,

$$Price(\Gamma, SA) \le \frac{3|M| - 1}{2|M|}.$$
(4)

Moreover, the bound in (4) is tight. That is, for any $\rho < \frac{3|M|-1}{2|M|}$ there is a scheduling problem with |M| machines such that for its associated game Γ' , $Price(\Gamma', SA) \ge \rho$.

Theorem 3 shows that there is no sharper bound of the price of the equilibrium heuristic than (4). Note that the bound is strictly increasing in the number of machines and that it converges to $\frac{3}{2}$.

4 Simulations

In this section we investigate the behavior of the price of selfish allocation in relation to the tight bound of Theorem 3. For this purpose we simulate four different classes of scheduling problems that are classified by the number of players and jobs. Examples of settings that have a good fit with essential elements of our model when restricted to the various parameter domains, i.e., to the four classes, are discussed at the end. First, we report the settings of the simulations. The inputs for each simulation are the number of players |N|, the number of jobs per player $|J_i|$ ($i \in N$), the processing time of each job p_j ($j \in J$), and the number of machines |M|. We distinguish among four different classes. The first class randomly selects the number of players from the set $\{2, 3, 4, 5\}$. The number of jobs of each player is randomly drawn from the set $\{1, 2, 3, 4, 5\}$ and the processing times of each job is randomly drawn from the set $\{1, 2, 3, 4, 5\}$ and the processing times of each job is randomly drawn from the interval (0, 1) of real numbers. Finally, the number of machines is randomly selected from the set $\{1, 2, ..., 10\}$. The second class differs from the first one only by the set of numbers of jobs which is replaced by $\{10, 11, 12, 13, 14, 15\}$. The third class differs from the first one only by the set of numbers of players which is replaced by $\{5, 6, 7, 8, 9, 10\}$. Finally, the fourth class differs from the first one by replacing the set of numbers of players by $\{5, 6, 7, 8, 9, 10\}$ and the set of numbers of machines by $\{2, 3, ..., 50\}$.

For each class 10,000 simulations are executed which results in the price of the selfish allocation heuristic for each simulated scheduling problem. The resulting numerical data is depicted by means of box plots in Figure 1. In each box plot the central line is the median, the central circle is the average, the edges of the box are the 25th and 75th percentiles, and the whiskers extend to the 2.5th (or lower) and 97.5th (or upper) percentiles. We have also included the graph of the function $x \mapsto \frac{3x-1}{2x}$, which by Theorem 3 gives the tight bound of the price of the equilibrium heuristic for any integer $x \ge 2$ when there are x machines. Finally, in each box plot, the medians are connected by the graph of a piecewise linear function.

Figure 1 a (b,c,d) represents the first (second, third, fourth) class of simulated scheduling problems and allows us to make the following observations.

First, we consider the percentage loss of performance with respect to optimal schedules. We observe that the average loss in performance is at most 23%. This value is attained in the fourth class with |M| = 16, i.e., with relatively high numbers of players and machines. Here, the average price is approximately 1.23. This is considerably lower than the tight upper bound in this situation: 1.468, which reflects a loss of performance of 47%. Moreover, the relative distance as a percentage of (related to the tight upper bound) the price averages and the corresponding tight upper bounds is at least 15%. This value is closely approximated in the third class with |M| = 10. If we consider the upper percentile of the price of selfish allocation, then we observe that this relative distance is at least 4%. This value is closely approximated in the first case with |M| = 2. However, in most situations it is more than 10%. So, in most situations, it seems reality will work out a lot better than the worst case possible outcome.

Second, taking the first class as starting point, we see that augmenting the number of jobs per



Figure 1: Box plots of price of selfish allocation for uniformly distributed processing times

players (i.e., shifting from the first class to the second class), the average price is reduced drastically. So, with a relatively more dispersed job profile, the extra costs are less. Moreover, the distance between the lower and upper percentiles is very small in comparison with this distance in the first class. This seems counter intuitive, but in these two classes we have only a few players. The difference between the two classes are the number of jobs. In the second class, where we have more jobs, the probability to obtain a bad schedule is smaller than in the first class. Therefore, the payoff of the equilibrium heuristic in the second class is considerably lower than the payoff of the equilibrium heuristic in the first class. If the number of players is augmented (i.e., shifting from the first class to the third class) we observe that the average price increases for sufficiently large |M|. This is caused by the fact that with a smaller number of players each of them optimizes already to quite an extent, and there is less room for a centralized decision maker to optimize this any further.

Third, if we augment the number of machines, the average price increases relatively quickly. This implies that the optimal costs decrease more rapidly (i.e., a centralized decision maker would have more room to optimize) than do the costs associated with the equilibrium heuristic. After the price reaches a maximum it decreases relatively slowly. In fact, we can argue that when the number of machines tends to infinite (and all other parameters do not change), the price of anarchy tends to 1. This is an immediate consequence of the fact that when there is a very large number of machines any optimal schedule processes at most one job on each machine and the probability that the equilibrium heuristic assigns two jobs to the same machine tends to 0.

All four classes can be envisaged. In many situations of outsourcing manufacturing jobs, one observes (i) relatively few players, (ii) relatively few outsourcing companies (machines) available and used in each specific industry, and (iii) relatively many jobs (particularly when it comes to manufacturing electronic components or producing electronics sub-assemblies (PCB boards) of assemblies). This suggests that the simulations of the second class are particularly relevant. Note that of all four classes it is precisely in the second class where the average price of selfish allocation is lowest, i.e., where our equilibrium heuristic is likely to perform well.

In industries like aerospace, the number of players (e.g., Boeing, Airbus) is small, and the number of outsourcing companies for, e.g., engines (GE, Rolls Royce) is small. However, a difference with electronics is that in these industries the number of jobs is relatively small as well. Therefore the latter industries seem to be best represented by the first class of our simulations.

The third class (which has more players in comparison to the first class) can also be encountered in the semi-conductor industry, particularly in the area of application specific or custom designs. In this section of the industry there are typically many specialized equipment producing players drawing upon a limited number of chip making companies (foundries). In comparison to the second class, these players (often smaller companies) submit less jobs per player. The price of selfish allocation in the third class is higher than in the first class, and much higher than in the second class.

In comparison to the third class, the fourth class can have many more machines (up to 50). This is typical for an industry situation where the outsourced operation is less capital intensive, allowing more and smaller service providers to be viable. Examples can be found in an assembly type of manufacturing, as well as packing or logistics operations. In this class we see that the price of selfish allocation increases with the number of machines up to a point, but then decreases after that point (i.e., when only one or very few jobs need to be allocated to a single machine).

We consider our simulations as a first necessary step before a decision about a possible cost-benefit analysis is to be made.

5 Price of anarchy and decentralization cost

So far, we have focused on the performance of equilibrium heuristics. For a specific instance of a multiple decision maker problem (sequencing situation in our setting) the equilibrium heuristic results

in an equilibrium. Naturally, its performance is in between the performance of the equilibrium with lowest total cost and the equilibrium with highest total cost, usually referred to as decentralization cost and price of anarchy, respectively. Exploiting this relation per instance, performance bound guarantees on a class of situations, naturally respect the same order between decentralization cost, price of equilibrium heuristic, and price of anarchy. In this section we focus on these relations between the three performance measures.

Let Λ be a scheduling problem. Let Γ be its associated scheduling game. The price of anarchy is the ratio between the highest cost across all Nash equilibria and the optimal cost. Formally, the price of anarchy is defined as

$$PoA(\Gamma) = \frac{\max_{\tilde{a}\in\mathcal{E}(\Gamma)}\sum_{j\in J}\tilde{C}_j(\sigma^{\tilde{a}})}{\sum_{j\in J}C_j(\sigma^*)},$$

where σ^* is an optimal schedule.

Similarly, the decentralization cost is the ratio between the lowest cost across all Nash equilibria and the optimal cost. Formally, the decentralization cost is defined as

$$DC(\Gamma) = \frac{\min_{\tilde{a} \in \mathcal{E}(\Gamma)} \sum_{j \in J} \tilde{C}_j(\sigma^{\tilde{a}})}{\sum_{j \in J} C_j(\sigma^*)}$$

where σ^* is an optimal schedule.

Theorem 2 states the profile obtained from the selfish allocation heuristic is a Nash equilibrium, implying that for any game Γ the performance of the equilibrium heuristic is in between the decentralization cost and the price of anarchy, i.e.,

$$1 \le DC(\Gamma) \le Price(\Gamma, SA) \le PoA(\Gamma).$$
(5)

Theorem 3 provides a performance guarantee on the whole class of sequencing situations for the price of selfish allocation. This performance guarantee is tight. In case tight performance guarantees can be found for decentralization cost and price of anarchy as well, these guarantees should respect an order in line with (5) as well. Figure 2 illustrates this relation in a stylized class containing exactly two instances. At the same time it illustrates that the three (tight) performance guarantees need not coincide.

In Figure 2 a class of multiple decision maker situations containing exactly two instances is considered. Each instance allows for several Nash equilibria. Equilibrium heuristic A selects equilibria A1 and A2 in instance 1 and 2, respectively. The left and middle parts in the figure illustrate the relation between price of anarchy (PoA; highest NE), decentralization cost (DC; lowest NE), and equilibrium heuristic performance per instance; the right part in the figure does the same for the associated tight performance guarantees for the class.



Figure 2: Relation between equilibrium heuristic, price of anarchy, and decentralization cost

Figure 2 illustrates that all three tight bounds could in principle be different. Assuming that there are |M| = 2 machines, in Example 4 we show that for any $\rho < \frac{5}{4} = \frac{3|M|-1}{2|M|}$ there is a scheduling game $\Gamma^{\epsilon(\rho)}$ with $DC(\Gamma^{\epsilon(\rho)}) = Price(\Gamma^{\epsilon(\rho)}, SA) = PoA(\Gamma^{\epsilon(\rho)}) \ge \rho$. Therefore, since the bound in Theorem 3 is tight, by (5), the same bound is tight for the decentralized cost as well.

Example 4. (|M| = 2. Price of anarchy, price of selfish allocation, and decentralization cost.) Let $M = \{1, 2\}$ and $N = \{1, 2\}$. Let $J_1 = \{\alpha, \gamma\}$ and $J_2 = \{\beta, \delta\}$. Let $\epsilon \in (0, \frac{1}{4})$. Suppose $(p_\alpha, p_\beta, p_\gamma, p_\delta) = (\epsilon, 2\epsilon, 1 - 2\epsilon, 1 - \epsilon)$. Note $p_\alpha < p_\beta < p_\gamma < p_\delta$. Consider the associated game Γ^{ϵ} . Similarly to the scheduling game discussed in Examples 1 and 2, each strategy can be fully described by indicating which jobs are sent to machine 1 (the complement is sent to machine 2). And, as before, there is a unique Nash equilibrium in mixed strategies. For player 1 it is always strictly better to play $\{\alpha\}$ than \emptyset , and it is always strictly better to play $\{\gamma\}$ than $\{\alpha, \gamma\}$. Similarly, for player 2 it is always strictly better to play $\{\beta\}$ than \emptyset , and it is always strictly better to play $\{\delta\}$ than $\{\beta, \delta\}$. So, in any Nash equilibrium, strategies \emptyset (for both players), $\{\alpha, \gamma\}$ (for player 1), and $\{\beta, \delta\}$ (player 2) receive probability 0. Hence, it suffices to restrict attention to the reduced game described in Table 2, where boldfaced numbers indicate best responses.

$1\backslash 2$	$\{\beta\}$	$\{\delta\}$
$\{\alpha\}$	$1-\epsilon$, 2	$1+\epsilon, 1+2\epsilon$
$\{\gamma\}$	$1+\epsilon, 1+2\epsilon$	$1-\epsilon$, 2

Table 2: Table of Example 4

Applying standard game-theoretic tools (see, e.g., Osborne, 2009) one can readily show that the strategy-profile in which each of the strategies $\{\alpha\}, \{\gamma\}, \{\beta\}, \text{ and } \{\delta\}$ receives probability $\frac{1}{2}$ constitutes the unique Nash equilibrium \tilde{a}^{SA} in mixed strategies.⁴ Hence, the cost induced by the unique Nash equilibrium are

$$\tilde{c}_1(\tilde{a}^{SA}) + \tilde{c}_2(\tilde{a}^{SA}) = \left[\frac{1}{2}(1-\epsilon) + \frac{1}{2}(1+\epsilon)\right] + \left[\frac{1}{2}(2) + \frac{1}{2}(1+2\epsilon)\right] = \frac{5}{2} + \epsilon.$$

From Theorem 1 it follows that the cost associated with any optimal schedule σ^* equals

$$c_1(\sigma^*) + c_2(\sigma^*) = \epsilon + 2\epsilon + (\epsilon + 1 - 2\epsilon) + (2\epsilon + 1 - \epsilon) = 2 + 3\epsilon.$$

Hence, from the unicity of the Nash equilibrium \tilde{a}^{SA} it follows that

$$DC(\Gamma^{\epsilon}) = Price(\Gamma^{\epsilon}, SA) = PoA(\Gamma^{\epsilon}) = \frac{\frac{5}{2} + \epsilon}{2 + 3\epsilon}$$

Note that $\lim_{\epsilon \to 0} DC(\Gamma^{\epsilon}) = \frac{5}{4}$, which for |M| = 2 coincides with the tight bound established for the price of selfish allocation in Theorem 3.

In particular, in view of (5), Example 4 shows that for |M| = 2, the tight bound of the price of selfish allocation is also a tight bound of the decentralization cost.

Corollary 1. Let Λ be a scheduling problem with |M| = 2. Let Γ be its associated scheduling game. Then,

$$DC(\Gamma) \le \frac{5}{4}.\tag{6}$$

Moreover, the bound in (6) is tight. That is, for any $\rho < \frac{5}{4}$ there is a scheduling problem with |M| = 2 such that for its associated game Γ' , $DC(\Gamma') \ge \rho$.

For the scheduling games in Example 4, the price of anarchy coincides with the price of selfish allocation. This follows from the following theorem, which shows this holds in fact for any scheduling game.

Theorem 4. Let Λ be a scheduling problem. Let Γ be its associated scheduling game. Then, the price of anarchy of Γ coincides with the price of selfish allocation of Γ , i.e., $PoA(\Gamma) = Price(\Gamma, SA)$.

Theorems 3 and 4 immediately give the following tight upper bound of the price of anarchy.⁵

⁴See also footnote 3.

⁵For the specific case of 1 job per player, Rahn and Schäfer (2013) independently derived a similar result as in our Corollary 2. Compared to our results we stress that they do not provide a proof, do not discuss the price of selfish allocation, nor do they look at multiple jobs per player.

Corollary 2. Let Λ be a scheduling problem. Let Γ be its associated scheduling game. Then,

$$PoA(\Gamma) \le \frac{3|M| - 1}{2|M|}.\tag{7}$$

Moreover, the bound in (7) is tight. That is, for any $\rho < \frac{3|M|-1}{2|M|}$ there is a scheduling problem with |M| machines such that for its associated game Γ' , $PoA(\Gamma') \ge \rho$.

Moreover, when there are more than two machines the upper bound of the price of selfish allocation is, by (5), an upper bound of the decentralization cost as well. Note that, opposed to Corollary 1 in which we could identify a tight bound for the case with exactly two machines, there is no guarantee that this bound is tight for more than two machines as well. Hence, identifying a *tight* upper bound of the decentralization cost in case of more than two machines is an interesting open problem.

6 Concluding remarks

Our analysis showed that optimal local decisions may lead to a joint outcome that can be improved upon. This phenomenon is apparent in a wide variety of operations management settings, for example in competitive outsourcing environments which are becoming increasingly important in business. In this paper, we therefore focused on the costs of outsourcing decisions being made individually rather than cooperatively.

This paper introduced a framework that takes into account two important issues that arise in these competitive situations. First, the dependence of an appropriate strategy on the strategies of other players, which justifies looking for equilibria. Second, the performance loss due to playing an equilibrium. In this framework, we first introduce a heuristic that for each agent prescribes the strategy in a specific setting. Second, we verify whether the strategy profile that results from the heuristics is a Nash equilibrium. If a Nash equilibrium is established by the heuristics we call it an equilibrium heuristic. Finally, we evaluate the performance of the heuristic equilibrium, i.e., we establish a bound on the performance of the corresponding equilibrium payoffs in relation to the centralized optimum.

We illustrated our framework in the specific setting of sequencing situations. Sequencing situations are not only interesting from a theoretical perspective, but also have their own merits in practical environments. Many examples can be found in the outsourcing and subcontracting of supply chain activities (both in manufacturing as well as logistics). For these situations, the results obtained in this paper help players to select and set their strategies in case they act as selfish agents. Our analysis quantified the difference between the individual and uncoordinated approach on the one hand, and the common and coordinated approach on the other. It did so for different situations that vary in the number of players (outsourcing companies), number of jobs, and the number of machines (outsourcing service providers). Simulation results showed that the performance gap in general is much smaller than the worst case theoretical outcome. It was also observed that this gap increases with the number of players (as with more selfish players there is more chance they are in conflict with each other), but decreases drastically with the number of jobs per player (as the negative effects of individualistic behavior tend to smooth out). Also the performance gap increases with the number of machines (with more machines a coordinated or centralized approach has more options to optimize), but only up to a point, after which it starts to decrease as the number of machines gets so large that only one or very few jobs need to be allocated per machine.

This all gives support to business management in deciding on the right outsourcing approach (which in a wider context includes decisions on the level of control on schedules, single vs. multiple sourcing, whether to include sourcing from service providers that also serve the direct competition, etc.). Knowing the performance gap, the management of outsourcing companies involved can decide how and to what extent they want to participate in coordinating strategies. Such a coordination on the one hand would introduce control effort, complexity and costs, but would allow to reach solutions that are more efficient overall, and thus provide benefits for all involved. The approach of this paper is a first step in a solid cost-benefit analysis of the desired levels of control and coordination in outsourcing situations. The ultimate goal is to further enrich the analysis in order to provide support to effective business management decisions.

Though focus of this paper has been on sequencing situations, we stress that our 3-step equilibrium heuristic framework can be applied in a much wider context. In fact it can be applied to any Operations Management setting in which locally optimal decisions can lead to an outcome that could centrally be improved upon. Moreover, it is interesting to consider this approach if the setting of the sequencing situation changes. For example, what will be the effect if there is a change in the cost criterion (e.g., makespan), job specific characteristics (e.g., release dates), job related characteristics (e.g., precedence relations), etcetera. Hence, this paper has the potential to initiate a new line of research that is related to sequencing situations.

7 Appendix

Lemma 1 is used in the proof of Theorems 2 and 4.

Lemma 1. For any pure strategy a_1° of player 1, any $j \in J_1$, and any $j' \in J \setminus J_1$,

$$\sum_{a \in A: a(j') = a(j)} \Pr(a \mid (a_1^{\circ}, (\tilde{a}_i^{SA})_{i \in N \setminus \{1\}})) = \frac{1}{|M|}.$$
(8)

Similarly, for any mixed strategies $(\tilde{a}_i^\circ)_{i \in N \setminus \{1\}}$ of players $N \setminus \{1\}$, any $j \in J_1$, and any $j' \in J \setminus J_1$,

$$\sum_{a \in A: a(j')=a(j)} \Pr(a \mid (\tilde{a}_1^{SA}, (\tilde{a}_i^{\circ})_{i \in N \setminus \{1\}})) = \frac{1}{|M|}.$$
(9)

Proof. Let $j \in J_1$. Let $j' \in J \setminus J_1$ and i = o(j'). Note $i \neq 1$. We first prove the first statement. By construction of \tilde{a}_i^{SA} , player i assigns job j' to any machine with probability $\frac{1}{|M|}$. Therefore, independently of player 1's strategy a_1° , job $j \in J_1$ will be on the same machine as j' with probability $\frac{1}{|M|}$. This proves the first statement.

We now prove the second statement. By construction of \tilde{a}_1^{SA} , player 1 assigns job $j \in J_1$ to any machine with probability $\frac{1}{|M|}$. Therefore, independently of player *i*'s strategy \tilde{a}_i° , job $j' \in J_i$ will be on the same machine as j with probability $\frac{1}{|M|}$. This proves the second statement.

Proof of Theorem 2:

Without loss of generality we show that player 1 has no profitable deviation. It suffices to show that there is no *pure* strategy deviation that yields strictly lower costs for player 1 (see, e.g., Osborne, 2009). Let a'_1 be a pure strategy of player 1. Let $\tilde{a}' = (a'_1, (\tilde{a}_i^{SA})_{i \in N \setminus \{1\}})$. Let $a_1 \in A_1^{AS}$, i.e., a_1 is a pure strategy of player 1 such that $Pr(a_1 | \tilde{a}_1^{SA}) > 0$. Then,

$$\begin{split} \tilde{c}_{1}(\tilde{a}') &= \sum_{j \in J_{1}} \tilde{C}_{j}(\sigma^{\tilde{a}'}) \\ &= \sum_{j \in J_{1}} \left(\sum_{a \in A} Pr(a|\tilde{a}')C_{j}(\sigma^{a}) \right) \\ &= \sum_{j \in J_{1}} \left(\sum_{a \in A} Pr(a|\tilde{a}') \left(p_{j} + \sum_{j' \in P(\sigma^{a},j)} p_{j'} \right) \right) \\ &= \sum_{j \in J_{1}} \left(p_{j} + \sum_{a \in A} Pr(a|\tilde{a}') \sum_{j' \in P(\sigma^{a},j)} p_{j'} \right) \\ &= \sum_{j \in J_{1}} \left(p_{j} + \sum_{a \in A} Pr(a|\tilde{a}') \left(\sum_{j' \in J_{1} \cap P(\sigma^{a},j)} p_{j'} + \sum_{j' \in (J \setminus J_{1}) \cap P(\sigma^{a},j)} p_{j'} \right) \right) \\ &= \sum_{j \in J_{1}} \left(p_{j} + \left(\sum_{a \in A} Pr(a|\tilde{a}') \sum_{j' \in J_{1} \cap P(\sigma^{a},j)} p_{j'} \right) + \left(\sum_{a \in A} Pr(a|\tilde{a}') \sum_{j' \in (J \setminus J_{1}) \cap P(\sigma^{a},j)} p_{j'} \right) \right) \\ &= \sum_{j \in J_{1}} \left(p_{j} + \sum_{j' \in J_{1} \cap P(\sigma^{a'_{1},j)}} p_{j'} \right) + \sum_{j \in J_{1}} \left(\sum_{a \in A} Pr(a|\tilde{a}') \sum_{j' \in (J \setminus J_{1}) \cap P(\sigma^{a},j)} p_{j'} \right) \\ &= \sum_{j \in J_{1}} \left(p_{j} + \sum_{j' \in J_{1} \cap P(\sigma^{a'_{1},j)}} p_{j'} \right) + \sum_{j \in J_{1}} \sum_{i \in N \setminus \{1\}} \sum_{i \in N \setminus \{1\}} \sum_{j' \in J_{i} \cap P(\sigma^{a},j)} p_{j'} \right) \\ &= \sum_{j \in J_{1}} \left(p_{j} + \sum_{j' \in J_{1} \cap P(\sigma^{a'_{1},j)}} p_{j'} \right) + \sum_{j \in J_{1}} \sum_{i \in N \setminus \{1\}} \sum_{i \in N \setminus \{0\}} \sum_{j' \in J_{i} \cap P(\sigma^{a},j)} p_{j'} \right) \\ &= \sum_{j \in J_{1}} \left(p_{j} + \sum_{j' \in J_{1} \cap P(\sigma^{a'_{1},j)}} p_{j'} \right) + \sum_{j \in J_{1}} \sum_{i \in N \setminus \{1\}} \sum_{j' \in J_{i} \cap P(\sigma^{a},j)} p_{j'} \right) \\ &= \sum_{j \in J_{1}} \left(p_{j} + \sum_{j' \in J_{1} \cap P(\sigma^{a'_{1},j)}} p_{j'} \right) + \sum_{j \in J_{1}} \sum_{i \in N \setminus \{1\}} \sum_{j' \in J_{i} \cap P(\sigma^{a},j)} p_{j'} \right) \\ &= \sum_{j \in J_{1}} \left(p_{j} + \sum_{j' \in J_{1} \cap P(\sigma^{a'_{1},j)}} p_{j'} \right) + \sum_{j \in J_{1}} \sum_{i \in N \setminus \{1\}} \sum_{j' \in J_{i} \cap P(\sigma^{a'},j)} p_{j'} \right) \\ &= \sum_{j \in J_{1}} \left(p_{j} + \sum_{j' \in J_{i} \cap P(\sigma^{a'_{i},j)}} p_{j'} \right) + \sum_{j \in J_{1}} \sum_{i \in N \setminus \{1\}} \sum_{j' \in J_{i} \cap P(\sigma^{a'_{i},j)}} p_{j'} \right) + \sum_{j \in J_{i}} \sum_{i \in N \setminus \{1\}} \sum_{j' \in J_{i} \cap P(\sigma^{a'_{i},j)}} p_{j'} \right) \\ &= \sum_{j \in J_{i}} \left(p_{j'} + \sum_{j' \in J_{i} \cap P(\sigma^{a'_{i},j)}} p_{j'} \right) + \sum_{j \in J_{i}} \sum_{i \in N \setminus \{1\}} \sum_{j' \in J_{i} \cap P(\sigma^{a'_{i},j)}} p_{j'} \right)$$

$$\begin{split} &= \sum_{j \in J_{1}} \left(p_{j} + \sum_{j' \in J_{1} \cap P(\sigma^{a'_{1}}, j)} p_{j'} \right) + \sum_{j \in J_{1}} \sum_{i \in N \setminus \{1\}} \left(\sum_{a \in A} Pr(a | \tilde{a}') \sum_{j' \in J_{i}: p_{j'} < p_{j}, a(j') = a(j)} p_{j'} \right) \\ &= \sum_{j \in J_{1}} \left(p_{j} + \sum_{j' \in J_{1} \cap P(\sigma^{a'_{1}}, j)} p_{j'} \right) + \sum_{j \in J_{1}} \sum_{i \in N \setminus \{1\}} \left(\sum_{j' \in J_{i}: p_{j'} < p_{j}} \left(\sum_{a \in A: a(j') = a(j)} Pr(a | \tilde{a}') \right) p_{j'} \right) \\ &\stackrel{(a)}{=} \sum_{j \in J_{1}} \left(p_{j} + \sum_{j' \in J_{1} \cap P(\sigma^{a'_{1}}, j)} p_{j'} \right) + \sum_{j \in J_{1}} \left(\sum_{i \in N \setminus \{1\}} \sum_{j' \in J_{i}: p_{j'} < p_{j}} \frac{p_{j'}}{|M|} \right) \\ &\stackrel{(b)}{\geq} \sum_{j \in J_{1}} \left(p_{j} + \sum_{j' \in J_{1} \cap P(\sigma^{a_{1}}, j)} p_{j'} \right) + \sum_{j \in J_{1}} \left(\sum_{i \in N \setminus \{1\}} \sum_{j' \in J_{i}: p_{j'} < p_{j}} \frac{p_{j'}}{|M|} \right) \\ &\stackrel{(c)}{=} \tilde{c}_{1}(a_{1}, (\tilde{a}_{i}^{SA})_{i \in N \setminus \{1\}}). \end{split}$$

Equality (a) follows from (8) with $a_1^{\circ} = a_1'$. Inequality (b) follows from the fact that by Remark 1 σ^{a_1} is optimal for $(\{1\}, J_1, (p_j)_{j \in J_1})$. Equality (c) follows from arguments similar to those applied to establish all previous equalities and (8) with $a_1^{\circ} = a_1$.

So far, we have shown that

$$\tilde{c}_1(a'_1, (\tilde{a}^{SA}_i)_{i \in N \setminus \{1\}}) \ge \tilde{c}_1(a^+_1, (\tilde{a}^{SA}_i)_{i \in N \setminus \{1\}})$$
 for all $a^+_1 \in A^{AS}_1$.

But then, since \tilde{c}_1 is linear, it immediately follows that

$$\begin{split} \tilde{c}_{1}(a'_{1}, (\tilde{a}^{SA}_{i})_{i \in N \setminus \{1\}}) &= \sum_{a_{1}^{+} \in A^{SA}_{1}} Pr(a_{1}^{+} | \tilde{a}^{SA}_{1}) \, \tilde{c}_{1}(a'_{1}, (\tilde{a}^{SA}_{i})_{i \in N \setminus \{1\}}) \\ &\geq \sum_{a_{1}^{+} \in A^{SA}_{1}} Pr(a_{1}^{+} | \tilde{a}^{SA}_{1}) \, \tilde{c}_{1}(a_{1}^{+}, (\tilde{a}^{SA}_{i})_{i \in N \setminus \{1\}}) \\ &= \tilde{c}_{1}(\tilde{a}^{SA}_{1}, (\tilde{a}^{SA}_{i})_{i \in N \setminus \{1\}}) \\ &= \tilde{c}_{1}(\tilde{a}^{SA}). \end{split}$$

This shows that deviation a'_1 is not profitable. Hence, \tilde{a}^{SA} is a Nash equilibrium. This completes the proof.

Preparations Proof of Theorem 3:

Let $i \in N$. Denote $J_i = \{j_{i1}, \ldots, j_{in_i}\}$. We denote the processing time of job j_{il} by p_{il} and assume without loss of generality that $p_{i1} < \cdots < p_{in_i}$, i.e., the processing time of the jobs is increasing in the second index. By Remark 2, we can conveniently fix, for any $i \in N$, any particular allocation a_i^{SA} generated by the selfish allocation heuristic. It is easy to verify that for any $i \in N$, the allocation a_i^{SA} such that

$$a_i^{SA}(j_{il}) = 1 + (l-1) \mod |M| \text{ for each } m \in M = \{1, \dots, |M|\} \text{ and each } l \in \{1, \dots, n_i\},$$
(10)

assigns the |M| largest jobs in J_i to different machines, the next |M| largest jobs to different machines, etc. Hence, this allocation can be generated by the selfish allocation heuristic. For this reason, we henceforth use exclusively the notation a_i^{SA} to denote the allocation specified in (10), unless explicitly noticed otherwise. Similarly, \tilde{a}_i^{SA} will denote the probability distribution obtained from applying the second step of the SA heuristic to the allocation specified in (10). Finally, $A_i^{SA} \subseteq A_i$ will denote the pure strategies that receive (the same) strictly positive probability at \tilde{a}_i^{SA} .

For $j \in J$, let o(j) denote the owner of job j, i.e., o(j) = i where $i \in N$ is such that $j \in J_i$. For $j \in J$, define

$$\lambda_j = |\{j' \in J : p_{j'} > p_j\}|$$
 and
 $\kappa_j = |\{j' \in J : p_{j'} > p_j \text{ and } o(j') = o(j)\}|.$

The next lemma provides a convenient expression for the sum of expected costs induced by the Nash equilibrium \tilde{a}^{SA} in terms of $(\kappa_j, \lambda_j)_{j \in J}$ and the processing times $(p_j)_{j \in J}$.

Lemma 2. Let Λ be a scheduling problem. Let Γ be its associated scheduling game. The sum of expected costs induced by Nash equilibrium \tilde{a}^{SA} is given by⁶

$$\sum_{j \in J} \tilde{C}_j(\tilde{a}^{SA}) = \sum_{j \in J} p_j \left(1 + \frac{\lambda_j - \kappa_j}{|M|} + \left\lfloor \frac{\kappa_j}{|M|} \right\rfloor \right).$$

Proof. For $i \in N$ and $m \in \{1, ..., |M|\}$, let J_{im} be the set of jobs of player *i* that allocation a_i^{SA} assigns to machine *m*, i.e.,

$$J_{im} = \{j \in J_i : a_i^{SA}(j) = m\}$$

Note that

$$\tilde{c}_{1}(\tilde{a}^{SA}) \stackrel{(d)}{=} \sum_{a_{1} \in A_{1}^{SA}} Pr(a_{1}|\tilde{a}_{1}^{SA}) \left(\tilde{c}_{1}(a_{1}, (\tilde{a}_{i}^{SA})_{i \in N \setminus \{1\}}) \right)$$

$$\stackrel{(e)}{=} \sum_{a_{1} \in A_{1}^{SA}} Pr(a_{1}|\tilde{a}_{1}^{SA}) \left(\sum_{j \in J_{1}} \left(p_{j} + \sum_{j' \in J_{1} \cap P(\sigma^{a_{1}}, j)} p_{j'} \right) + \sum_{j \in J_{1}} \left(\sum_{i \in N \setminus \{1\}} \sum_{j' \in J_{i}: p_{j'} < p_{j}} \frac{p_{j'}}{|M|} \right) \right),$$

where equality (d) follows from the linearity of \tilde{c}_1 and equality (e) follows from (c) in the proof of Theorem 2.

Since player 1 was an arbitrarily chosen player in the proof of Theorem 2, it holds that in fact for each player $i \in N$,

$$\tilde{c}_i(\tilde{a}^{SA}) = \sum_{a_i \in A_i^{SA}} Pr(a_i | \tilde{a}_i^{SA}) \left(\sum_{j \in J_i} \left(p_j + \sum_{j' \in J_i \cap P(\sigma^{a_i}, j)} p_{j'} \right) + \sum_{j \in J_i} \left(\sum_{i' \in N \setminus \{i\}} \sum_{j' \in J_{i'}: p_{j'} < p_j} \frac{p_{j'}}{|M|} \right) \right)$$

⁶For $x \in \mathbb{R}$, $\lfloor x \rfloor$ denotes the largest integer n with $n \leq x$.

Hence,

$$\begin{split} \sum_{j \in J} \tilde{C}_{j}(\sigma^{\tilde{a}^{SA}}) &= \sum_{i \in N} \tilde{c}_{i}(\tilde{a}^{SA}) \\ &= \sum_{i \in N} \sum_{a_{i} \in A_{i}^{SA}} Pr(a_{i}|\tilde{a}_{i}^{SA}) \left(\sum_{j \in J_{i}} \left(p_{j} + \sum_{j' \in J_{i} \cap P(\sigma^{a_{i}}, j)} p_{j'} \right) + \sum_{j \in J_{i}} \left(\sum_{i' \in N \setminus \{i\}} \sum_{j' \in J_{i'}, p_{j'} < p_{j}} \frac{p_{j'}}{|M|} \right) \right) \\ &= \sum_{i \in N} \sum_{a_{i} \in A_{i}^{SA}} Pr(a_{i}|\tilde{a}_{i}^{SA}) \left(\sum_{m \in M} \sum_{j \in J_{im}} \left(p_{j} + \sum_{j' \in J_{i} \cap P(\sigma^{a_{i}}, j)} p_{j'} \right) \right) + \sum_{i \in N} \sum_{a_{i} \in A_{i}^{SA}} Pr(a_{i}|\tilde{a}_{i}^{SA}) \left(\sum_{j \in J_{i}} \sum_{i' \in N \setminus \{i\}} \sum_{j' \in J_{i'}, p_{j'} < p_{j}} \frac{p_{j'}}{|M|} \right) \\ &= \sum_{i \in N} \sum_{m \in M} \sum_{j \in J_{im}} \left(p_{j} + \sum_{j' \in J_{im}; p_{j'} < p_{j}} p_{j'} \right) + \sum_{i \in N} \sum_{j \in J_{i}} \sum_{i' \in N \setminus \{i\}} \sum_{j' \in J_{i'}; p_{j'} < p_{j}} \frac{p_{j'}}{|M|} \\ &= \sum_{i \in N} \sum_{m \in M} \sum_{j \in J_{im}} \left(p_{j} + \sum_{j' \in J_{im}; p_{j'} < p_{j}} p_{j} \right) + \sum_{i \in N} \sum_{j \in J_{i}} \sum_{i' \in N \setminus \{i\}} \sum_{j' \in J_{i'}; p_{j'} < p_{j}} \frac{p_{j'}}{|M|} \\ &= \sum_{i \in N} \sum_{m \in M} \sum_{j \in J_{im}} \left(p_{j}(1 + \sum_{j' \in J_{im}; p_{j'} > p_{j}} p_{j}) + \sum_{i \in N} \sum_{j \in J_{i}} \sum_{i' \in N \setminus \{i\}} \sum_{j' \in J_{i'}; p_{j'} < p_{j}} \frac{p_{j}}{|M|} \\ &= \sum_{j \in J} p_{j} \left(1 + \left\lfloor \frac{K_{j}}{|M|} \right\rfloor \right) + \sum_{j \in J} \frac{p_{j}}{|M|} (\lambda_{j} - \kappa_{j}) \\ &= \sum_{j \in J} p_{j} \left(1 + \frac{\lambda_{j} - \kappa_{j}}{|M|} + \left\lfloor \frac{\kappa_{j}}{|M|} \right\rfloor \right), \end{split}$$
which proves the desired equality.

which proves the desired equality.

The next lemma shows that for a given set of jobs the price of the allocation heuristic is maximal when all jobs are owned by different players. Let $\Lambda = (N, (J_i)_{i \in N}, (p_j)_{j \in J})$ be a scheduling problem. We define its associated simple scheduling problem by $\Lambda' = (N', (J_i)_{i \in N'}, (p_j)_{j \in J})$ where N' is such that |N'| = |J| and each player in N' owns exactly one job in J, i.e., for each $i \in N'$, $|J_{i'}| = 1$.

Lemma 3. Let Γ be the game associated with a scheduling problem Λ . Let Γ' be the game associated with the corresponding simple schedule problem Λ' . Then, $Price(\Gamma, SA) \leq Price(\Gamma', SA)$.

Proof. Let $\tilde{a}^{SA}(\Lambda)$ be the equilibrium heuristic for Λ and let $\tilde{a}^{SA}(\Lambda')$ be the equilibrium heuristic for Λ' . Then, by Lemma 2,

$$\sum_{j \in J} \tilde{C}_j(\sigma^{\tilde{a}^{SA}(\Lambda)}) = \sum_{j \in J} p_j \left(1 + \frac{\lambda_j - \kappa_j}{|M|} + \left\lfloor \frac{\kappa_j}{|M|} \right\rfloor \right)$$
$$\leq \sum_{j \in J} p_j \left(1 + \frac{\lambda_j}{|M|} \right)$$
$$= \sum_{j \in J} \tilde{C}_j(\sigma^{\tilde{a}^{SA}(\Lambda')}).$$

Let σ^* be an optimal schedule (for both Λ and Λ'). Then,

$$Price(\Gamma, SA) = \frac{\sum_{j \in J} \tilde{C}_j(\sigma^{\tilde{a}^{SA}}(\Lambda))}{\sum_{j \in J} C_j(\sigma^*)} \le \frac{\sum_{j \in J} \tilde{C}_j(\sigma^{\tilde{a}^{SA}}(\Lambda'))}{\sum_{j \in J} C_j(\sigma^*)} = Price(\Gamma', SA),$$

which completes the proof.

Lemma 3 shows that the extra cost of decentralized decision-making is maximal when the set of all jobs is maximally spread over all players; and thus there are less costs when the set of jobs is concentrated in less players.

The following lemma provides a convenient expression for the optimal sum of costs (as if there were a central authority).

Lemma 4. Let Λ be a scheduling problem. For any optimal schedule σ^* , the associated (optimal) sum of costs equals

$$\sum_{j \in J} C_j(\sigma^*) = \sum_{j \in J} p_j \left(1 + \left\lfloor \frac{\lambda_j}{|M|} \right\rfloor \right).$$

Remark 3. Obviously, the optimal cost is independent of the owners of the jobs. Also, in case there is a unique player, the cost associated with any Nash equilibrium is optimal. Therefore, Lemma 4 can be obtained from Lemma 2 by assuming that there is a unique player (and hence, $\kappa_j = \lambda_j$ for all $j \in J$).

Proof of Theorem 3:

Let $\Lambda = (N, (J_i)_{i \in N}, (p_j)_{j \in J})$ be a scheduling problem. Let Γ be the game associated with Λ . By Lemma 3, we may assume that for each $i \in N$, $|J_i| = 1$. With a slight abuse of notation, let $J = N = \{1, \ldots, n\}$. Without loss of generality we assume that $p_1 > \cdots > p_n$. Let $K = \lfloor \frac{|J|}{|M|} \rfloor$. We assume $K \neq \frac{|J|}{|M|}$ since the case $K = \frac{|J|}{|M|}$ follows from similar (but easier) arguments.

The sum of expected costs induced by the equilibrium heuristic \tilde{a}^{SA} can now be written as

$$\begin{split} \sum_{j \in J} \tilde{C}_{j}(\sigma^{\tilde{a}^{SA}}) &= \sum_{j \in J} p_{j} \left(1 + \frac{\lambda_{j} - \kappa_{j}}{|M|} + \left\lfloor \frac{\kappa_{j}}{|M|} \right\rfloor \right) \\ &= \sum_{j \in J} p_{j} (1 + \frac{\lambda_{j}}{|M|}) \\ &= \sum_{k=0,\dots,K-1} \sum_{l=1,\dots,|M|} p_{(k|M|+l)} \left(1 + k + \frac{l-1}{|M|} \right) + \left(p_{(K|M|+1)} (1 + K + \frac{0}{|M|}) + \dots + p_{|J|} (1 + K + \frac{|J| - K|M| - 1}{|M|}) \right) \end{split}$$

$$(11)$$

$$\leq \sum_{k=0,\dots,K-1} \sum_{l=1,\dots,|M|} p_{(k|M|+l)} \left(1+k+\frac{|M|-1}{2|M|}\right) + \left(p_{(K|M|+1)}\left(1+K+\frac{|M|-1}{2|M|}\right)+\dots+p_{|J|}\left(1+K+\frac{|M|-1}{2|M|}\right)\right)$$
$$= \sum_{k=0,\dots,K-1} \left(1+k+\frac{|M|-1}{2|M|}\right) \left(\sum_{l=1,\dots,|M|} p_{(k|M|+l)}\right) + \left(12\right) \left(1+K+\frac{|M|-1}{2|M|}\right) \left(p_{(K|M|+1)}+\dots+p_{|J|}\right).$$

Here, the first equality follows from Lemma 2. The second equality follows from the fact that for all $j \in J$, $\kappa_j = 0$ (since for all $i \in N$, $|J_i| = 1$). The inequality follows from the identity

$$\sum_{l=1,\dots,|M|} \frac{l-1}{|M|} = \frac{|M|-1}{2}$$

and the redistribution of this sum in such a way that the jobs with longer (shorter) processing times have larger (smaller) coefficients on the right hand side than on the left hand side of the inequality. (For the case k = K, the sum of coefficients is even augmented.)

The minimal sum of costs induced by an optimal schedule σ^* can be written as

$$\sum_{j \in J} C_{j}(\sigma^{*}) = \sum_{j \in J} p_{j} \left(1 + \left\lfloor \frac{\lambda_{j}}{|M|} \right\rfloor \right)$$

$$= \sum_{\substack{k=0,...,K-1 \\ (1+K) \left(\sum_{l=1,...,|M|} p_{(k|M|+l)} \right) + \dots + p_{|J|} \right),$$
(13)

where the first equality follows from Lemma 4.

For any integer $k \ge 0$, we have

$$\frac{1+k+\frac{|M|-1}{2|M|}}{1+k} = \frac{1+k}{1+k} + \frac{\frac{|M|-1}{2|M|}}{1+k} \le 1 + \frac{|M|-1}{2|M|}.$$
(14)

Let $k = 0, \dots, K - 1$. From (14) it follows that

$$\frac{\left(1+k+\frac{|M|-1}{2|M|}\right)\left(\sum_{l=1,\dots,|M|}p_{(k|M|+l)}\right)}{(1+k)\left(\sum_{l=1,\dots,|M|}p_{(k|M|+l)}\right)} \le \left(1+\frac{|M|-1}{2|M|}\right).$$
(15)

Similarly,

$$\frac{\left(1+K+\frac{|M|-1}{2|M|}\right)\left(p_{(K|M|+1)}+\dots+p_{|J|}\right)}{(1+K)\left(p_{(K|M|+1)}+\dots+p_{|J|}\right)} \le \left(1+\frac{|M|-1}{2|M|}\right).$$
(16)

From (12), (13), (15), (16) and the fact that for any $\beta, \delta > 0$ and any $\alpha, \gamma, \epsilon \in \mathbb{R}$, $\left[\frac{\alpha}{\beta}, \frac{\gamma}{\delta} \le \epsilon \implies \frac{\alpha + \gamma}{\beta + \delta} \le \epsilon\right]$ it follows that

$$\frac{\sum_{j\in J} \tilde{C}_j(\sigma^{\tilde{a}^{SA}})}{\sum_{j\in J} C_j(\sigma^*)} \le \left(1 + \frac{|M| - 1}{2|M|}\right) = \frac{3|M| - 1}{2|M|},$$

which completes the first part of the proof.

Next, we prove that the bound is tight. Let $\rho < \frac{3|M|-1}{2|M|}$. Let n = |M|. Let $0 < p_1 < p_2 < \cdots < p_{n-1} < p_n$ be such that

$$\frac{p_1}{p_n} \left(\frac{3n-1}{2n}\right) \ge \rho. \tag{17}$$

Let $\Lambda = (N, (J_i)_{i \in N}, (p_j)_{j \in J})$ be the scheduling problem for which

- |J| = |N| = |M|;
- $J = \{1, ..., n\};$
- for each $i \in N$, $|J_i| = 1$; and
- for each $j \in J$, the processing time of job j equals p_j .

From Lemma 2, for the sum of expected costs induced by the equilibrium heuristic \tilde{a}^{SA} we have

$$\sum_{j \in J} \tilde{C}_j(\sigma^{\tilde{a}^{SA}}) = \sum_{j=1,\dots,n} p_j\left(1 + \frac{n-j}{n}\right).$$
(18)

Obviously, the minimal sum of costs induced by an optimal schedule σ^* equals

$$\sum_{j \in J} C_j(\sigma^*) = \sum_{j=1,\dots,n} p_j.$$
⁽¹⁹⁾

From (18) and (19) it follows that for the game Γ associated with Λ we have

$$Price(\Gamma, SA) = \frac{\sum_{j \in J} \tilde{C}_j(\sigma^{\tilde{a}^{SA}})}{\sum_{j \in J} C_j(\sigma^*)}$$

$$= \frac{\sum_{j=1,...,n} p_j \left(1 + \frac{n-j}{n}\right)}{\sum_{j=1,...,n} p_j}$$

$$\geq \frac{\sum_{j=1,...,n} p_j + \sum_{j=1,...,n} p_j \left(\frac{n-j}{n}\right)}{np_n}$$

$$\geq \frac{np_1 + p_1 \sum_{j=1,...,n} \left(\frac{n-j}{n}\right)}{np_n}$$

$$= \frac{np_1 + p_1 \left(\frac{n-1}{2}\right)}{np_n}$$

$$= \frac{p_1 \left(n + \frac{n-1}{2}\right)}{p_n n}$$

$$= \frac{p_1}{p_n} \left(\frac{3n-1}{2n}\right)$$

$$\geq \rho,$$

where the last inequality follows from (17). This completes the second part of the proof.

Proof of Theorem 4:

Let \tilde{a} be a Nash equilibrium. It is sufficient to show that each player has higher costs at \tilde{a}^{SA} than at \tilde{a} . Without loss of generality we show that player 1 has higher costs at \tilde{a}^{SA} than at \tilde{a} . We have

$$\begin{split} \tilde{c}_{1}(\tilde{a}) &\leq \tilde{c}_{1}(\tilde{a}_{1}^{SA}, (\tilde{a}_{i})_{i\in N\setminus\{1\}}) \\ \stackrel{(f)}{=} \sum_{a_{1}\in A_{1}^{SA}} Pr(a_{1}|\tilde{a}_{1}^{SA}) \left(\tilde{c}_{1}(a_{1}, (\tilde{a}_{i})_{i\in N\setminus\{1\}}) \right) \\ &= \sum_{a_{1}\in A_{1}^{SA}} Pr(a_{1}|\tilde{a}_{1}^{SA}) \left(\sum_{j\in J_{1}} \left(p_{j} + \sum_{j'\in J_{1}\cap P(\sigma^{a_{1}}, j)} p_{j'} \right) \right) + \\ \sum_{a_{1}\in A_{1}^{SA}} Pr(a_{1}|\tilde{a}_{1}^{SA}) \left(\sum_{j\in J_{1}} \left(\sum_{i\in N\setminus\{1\}} \sum_{j'\in J_{1}\cap P(\sigma^{a_{1}}, j)} p_{j'} \right) \right) + \\ \sum_{a_{1}\in A_{1}^{SA}} Pr(a_{1}|\tilde{a}_{1}^{SA}) \sum_{j\in J_{1}} \left(p_{j} + \sum_{j'\in J_{1}\cap P(\sigma^{a_{1}}, j)} p_{j'} \right) \right) + \\ \sum_{j\in J_{1}} \left(\sum_{i\in N\setminus\{1\}} \sum_{j'\in J_{i}: p_{j'} < p_{j}} \sum_{a\in A: a(j')=a(j)} Pr(a|(\tilde{a}_{1}^{SA}, (\tilde{a}_{i})_{i\in N\setminus\{1\}}))p_{j'} \right) \right) \\ \stackrel{(g)}{=} \left(\sum_{a_{1}\in A_{1}^{SA}} Pr(a_{1}|\tilde{a}_{1}^{SA}) \sum_{j\in J_{1}} \left(p_{j} + \sum_{j'\in J_{1}\cap P(\sigma^{a_{1}}, j)} p_{j'} \right) \right) + \\ \sum_{j\in J_{1}} \left(\sum_{i\in N\setminus\{1\}} \sum_{j'\in J_{i}: p_{j'} < p_{j}} \frac{p_{j'}}{|M|} \right) \\ = \sum_{a_{1}\in A_{1}^{SA}} Pr(a_{1}|\tilde{a}_{1}^{SA}) \left(\sum_{j\in J_{1}} \left(p_{j} + \sum_{j'\in J_{1}\cap P(\sigma^{a_{1}}, j)} p_{j'} \right) + \sum_{j\in J_{1}} \left(\sum_{i\in N\setminus\{1\}} \sum_{j'\in J_{i: p_{j'}} < p_{j}} \frac{p_{j'}}{|M|} \right) \\ \stackrel{(h)}{=} \sum_{a_{1}\in A_{1}^{SA}} Pr(a_{1}|\tilde{a}_{1}^{SA}) \left(\tilde{c}_{1}(a_{1}, (\tilde{a}_{i}^{SA})_{i\in N\setminus\{1\}}) \right) \\ \stackrel{(i)}{=} \tilde{c}_{1}(\tilde{a}^{SA}). \end{split}$$

Here, the inequality follows from the fact that \tilde{a} is a Nash equilibrium. Equalities (f) and (i) follows from the linearity of \tilde{c}_1 . Equality (g) follows from (9) with $\tilde{a}_i^\circ = \tilde{a}_i$, $i \in N \setminus \{1\}$. Equality (h) follows from (c) in the proof of Theorem 2. Therefore, for any player the expected costs at \tilde{a}^{SA} are higher than those at any other Nash equilibrium \tilde{a} .

Proof of Remark 2:

In the proof of Theorem 4 we show that for each player the equilibrium heuristic \tilde{a}^{SA} induced by (10) yields weakly higher costs than any other Nash equilibrium. But one easily verifies that the arguments

in Theorem 4 do not rely on our specific choice of \tilde{a}^{SA} . Therefore, any other Nash equilibrium in the statement of Theorem 2 similarly yields for each player weakly higher costs than any other Nash equilibrium. Then, for each player, all Nash equilibria in Theorem 2 yield precisely the same costs. \Box

References

- Braess, D. 2005. On a paradox of traffic planning. Transportation Science **39** 446–450.
- Bukchin, Y., E. Hanany. 2007. Decentralization cost in scheduling: a game-theoretic approach. Manufacturing & Service Operations Management 9(3) 263–275.
- Bukchin, Y., E. Hanany. 2011. Decentralization cost in supply chain jobshop scheduling with minimum flowtime objective. Research memo, Tel Aviv University, Israel.
- Chen, Q., L. Lin, Z. Tan, Y. Yan. 2017. Coordination mechanisms for scheduling games with proportional deterioration. *European Journal of Operational Research* 263 380–389.
- Cole, R., J. Correa, V. Gkatzelis, V. Mirrokni, N. Olver. 2015. Decentralized utilitarian mechanisms for scheduling games. *Games and Economic Behavior* **92** 306–326.
- Correa, J., M. Queyranne. 2012. Efficiency of equilibria in restricted uniform machine scheduling with total weighted completion time as social cost. Naval Research Logistics 95(5) 384–395.
- Fransoo, J., C. Lee. 2013. The critical role of ocean container transport in global supply chain performance. Production and Operations Management 22 253–268.
- Graham, R. 1969. Bounds on multiprocessing timing anomalies. SIAM Journal of Applied Mathematics 17 416–429.
- Graham, R., E. Lawler, J.K. Lenstra, A. Rinnooy Kan. 1979. Optimization and approximation in deterministic sequencing and scheduling: a survey. Annals of Discrete Mathematics 5 287–326.
- Hoeksma, R., M. Uetz. 2012. The price of anarchy for minsum related machine scheduling. Approximation and Online Algorithms (WAOA 2011), R. Solis-Oba and G. Persiano (eds.). Lecture Notes in Computer Science 7164, 404–413.
- Horowitz, E., S. Sahni. 1976. Exact and approximate algorithms for scheduling non-identical processors. Journal of the ACM 23 317–327.

- Immorlica, N., L. Li, V. Mirrokni, A. Schulz. 2009. Coordination mechanisms for selfish scheduling. *Theoretical Computer Science* 410 1589–1598.
- Koutsoupias, E., C. Papadimitriou. 1999. Worst-case equilibria. Lecture notes in Computer Science, 16th annual symposium on theoretical aspects of computer science. Springer, Germany, 404–413.
- Lange, P. De, A. Chouly. 2004. Hinterland access regimes in seaports. European Journal of Transport and Infrastructure Research 4 361–380.
- Lee, K., J. Leung, M. Pinedo. 2012. Coordination mechanisms for parallel machine scheduling. European Journal of Operational Research 220 305–313.
- Nash, J. 1951. Non-cooperative games. Annals of Mathematics 54 286–295.
- Osborne, M. 2004. An introduction in game theory. Oxford University Press, Oxford.
- Perakis, G., G. Roels. 2007. The price of anarchy in supply chains: Quantifying the efficiency of price-only contracts. *Management Science* 53 1249–1268.
- Pinedo, M. 2002. Scheduling: theory, algorithms and systems. Prentice Hall, Englewood Cliffs, NJ.
- Rahn, M., G. Schäfer. 2013. Bounding the inefficiency of altruism through social contribution games.Y. Chen, N. Immorlica, eds., Web and internet economics. Springer, London, 391–404.
- Smith, W. 1956. Various optimizers for single-stage production. Naval Research Logistics Quarterly 3 59–66.