# Sequencing situations with just-in-time arrival, and related games

Math Meth Oper Res (2014) 80:285–305
DOI 10.1007/s00186-014-0481-x

ORIGINAL ARTICLE

# Sequencing situations with Just-in-Time arrival, and related games

**Edwin Lohmann · Peter Borm · Marco Slikker**

Received: 4 November 2012 / Accepted: 29 September 2014 / Published online: 22 October 2014


**Abstract** In this paper sequencing situations with Just-in-Time (JiT) arrival are introduced. This new type of one-machine sequencing situations assumes that a job is available to be handled by the machine as soon as its predecessor is finished. A basic predecessor dependent set-up time is incorporated in the model. Sequencing situations with JiT arrival are first analyzed from an operations research perspective: for a subclass an algorithm is provided to obtain an optimal order. Secondly, we analyze the allocation problem of the minimal joint cost from a game theoretic perspective. A corresponding sequencing game is defined followed by an analysis of a context-specific rule that leads to core elements of this game.

**Keywords** Just-in-Time arrival · Set-up time · Sequencing situations · Cooperative game theory · Core · Nucleolus

## 1 Introduction

Sequencing theory deals with a variety of problems sharing several characteristics: a number of jobs have to be processed on one or more machines, in such a way

Edwin Lohmann would like to thank the Department of Econometrics and Operations Research at Tilburg University. This research was conducted as part of the PhD-program at this university.

E. Lohmann · P. Borm
CentER and Department of Econometrics and OR,
Tilburg University, P.O. Box 90153, 5000, LE, Tilburg, The Netherlands

M. Slikker (✉)
School of Industrial Engineering, Eindhoven University of Technology, P.O. Box 513,
5600 MB Eindhoven, The Netherlands
e-mail: M.Slikker@tue.nl

that a cost criterion is minimized. From one sequencing problem to another the way these characteristics are defined can differ and additional constraints can be added: the machines can be parallel or serial, there can be conditions on the order in which the jobs should be processed and different cost criteria can be used. Applications of the theory of sequencing situations are numerous and diverse: from manufacturing and maintenance to scheduling patients in an operating room.

The starting point of the game theoretic analysis of sequencing situations is the paper by Curiel et al. (1989). In their one-machine model, only one job can be processed at a time. The processing time is deterministic for every job, and every job has a certain constant cost per time unit it spends in the system. A job is in the system from the moment the machine starts processing the first job until the job itself is processed by the machine. An order that minimizes total cost, processes the jobs in a decreasing order with respect to their urgency [cost per time unit divided by the length of the job, cf. Smith (1956)]. A procedure is introduced that, given an initial order, uses neighbor switches to obtain the optimal order and constructs a stable cost allocation in the process. Since Curiel et al. (1989) several related classes of sequencing problems are discussed, including ready times, due dates, multiple machines and numerous cost criteria (see e.g., Curiel et al. 2002; Borm et al. 2002; Calleja et al. 2002; Slikker 2005 for game-theoretic discussions).

From an operations research point of view, classes of sequencing problems where between jobs one needs time to set-up the machine are discussed extensively in the literature. Different types of set-up time are considered to match the application under consideration, such as sequencing aircraft landings (Psaraftis 1980) and steel pipe manufacturing (Ahn and Hyun 1990). In Gupta (1988) the mean flow time is minimized in sequencing situations with switching times between jobs depending on the class of both jobs. The change-over model by van der Veen et al. (1998) on the other hand minimizes the makespan in a setting which uses set-up times as well as after-processing times to define the switching time. Çiftçi (2009) discusses a sequencing model where a predecessor-independent switching time occurs if two subsequent jobs belong to different classes of jobs. Here, the focus is on the game-theoretic aspect of cost-sharing, based on the presence of an initial queue.

Similar to Gupta (1988) and van der Veen et al. (1998), we incorporate a set-up time in the model that is to be executed by the job to be processed next. We consider a basic setting of this set-up time - one can think of cleaning up a machine, adjusting a machine to the new jobs or something simple as erasing the blackboard before one can start the lecture—such that it only depends on the state in which the system is left behind by the predecessor.

A key feature of the sequencing situations discussed in this paper is the Just-in-Time (JiT) arrival of the jobs: a job is available to be handled by the machine as soon as its predecessor is finished. Hence, we leave the setting of the sequencing literature described above, where every job is waiting in a queue from the moment the first job starts. The owner of the job has to execute the set-up himself. So, given the job that is processed before, the time it takes before a next job can start processing is fixed. The time a job spends being processed is formed by this predecessor dependent set-up time and the length of the job itself. This way, the cost incurred by a job depends on the set-up time, its own processing time and his individual cost per time unit. As the costs incurred during the processing of the job itself is constant across all possible orders of jobs, we do not incorporate these costs into our analysis. For sequencing without set-ups, JiT arrival of the jobs would imply that the time a job spends in the system equals the processing time of the job itself. Of course, this problem is trivial as in this case every order of jobs results in the same costs.

Verdaasdonk (2007) initiated the study on this subject. The model discussed in this paper can also be modeled in terms of the traveling salesman problem (see e.g. Lawler et al. 1985). Our specific costs structure makes that the matrix underlying the traveling salesman problem corresponding with a sequencing situation with JiT arrival in general is not contained in any well-studied matrix class for the traveling salesman problem such as the class of Monge matrices or Van der Veen matrices (see, e.g., Burkard et al. 1998). This means that, to the best of our knowledge, for this specific subclass of traveling salesman problems no efficient algorithm is known in the literature. Although the corresponding optimization problem incorporates features from matching problems, the basic idea of combining jobs with high set-up times with jobs with low costs per time unit, and vice versa, is flawed by the sequencing nature of JiT arrival. Typically pairs of jobs cannot be viewed as separate entities, but interact with jobs before and behind them. Moreover, the fact that one has to choose first and last jobs in the queue creates further complications. We focus in the current paper on those sequencing situations with JiT arrival and predecessor dependent set-up times, where there are two different values for the set-up time and two different values for the costs per time unit.

The topic will be treated from two perspectives. The first part concerns the operations research perspective. For each sequencing situation with JiT arrival (or, for short, JiT sequencing situation) we provide sufficient conditions to check if an order minimizes joint costs, and provide an algorithm to obtain such an optimal order. Some remarks on a possible generalization to a larger class of JiT sequencing situations are provided. The second part, concerning the game theoretic perspective, involves the allocation of the minimal joint costs. For this, we define an objective and consistent way to determine cost savings for each coalition of jobs, thus defining a cooperative JiT sequencing game with transferable utility. In particular, we assume that the players are part of a larger system with players in the system before and after the grand coalition, so that for the grand coalition consistent with the worst-case approach for subcoalitions, the system is left behind with high set-up time due.

We show that every JiT sequencing game has a non-empty core, which implies that there exist stable reallocations of the total joint cost savings such that no coalition has an incentive to stop cooperating with the other players. In particular we provide a context specific allocation rule, the *large instance based allocation rule*, that for any JiT sequencing situation leads to a core element of the corresponding JiT sequencing game. It turns out that in general the core of a JiT sequencing game can be quite large. Focussing on *complete* JiT sequencing situations in which there is at least one player of each set-up/cost per time unit type (high–high, high/low, low/high, low/low), we formulate conditions that guarantee the core either to be just one point or a line segment. Under the same conditions we show that the outcome of the large instance based allocation rule coincides with the nucleolus of the corresponding JiT sequencing game. Our results when the core is a line segment are in line with results on assignment games (cf. Shapley and Shubik 1972; Böhm-Bawerk 1891; Núñez and Rafels 2005).

This paper is organized as follows: in the subsequent section, we formally introduce the JiT sequencing model. Also, we provide optimality conditions regarding the processing order, and give an algorithm to find an optimal order. Section 3 analyzes general JiT sequencing games while Sect. 4 focusses on complete sequencing games.

## 2 JiT sequencing situations

A *JiT sequencing situation* is defined by a tuple $\Psi = (N, \alpha, s, s_0)$. Here, $N$ denotes the nonempty finite player set. It is assumed that every player owns exactly one job. As there is a one-to-one correspondence between players and jobs, we will use the words player and job interchangeably throughout this paper. The vector $\alpha \in \mathbb{R}^N_+$ is such that for player $i \in N$, the costs of spending $t$ time units in the system is given by $\alpha_i t$. The *set-up times* are denoted by the vector $s \in \mathbb{R}^N_+$, where for $i \in N$, $s_i$ is the set-up time needed after the job of player $i$ is processed and before the machine can process another job. The time needed before the machine can process the first job is denoted by $s_0$.

An *order of processing* $\sigma$ of the jobs in $N$ is a bijective function $\sigma : \{1, \ldots, |N|\} \to N$. Here, $\sigma(k)$ denotes the player at position $k \in \{1, \ldots, |N|\}$ in the order $\sigma$. The set of all orders of $N$ is denoted with $\Pi(N)$. For notational convenience, we set $\sigma(0) = 0$ and therefore $s_{\sigma(0)} = s_0$ for all $\sigma \in \Pi(N)$.

In JiT sequencing situations, it is assumed that a player enters the system at the moment the player starts to prepare the machine for his job and leaves the system as soon as his job is finished. This situation is shown in Fig. 1. This differs from standard sequencing problems as depicted in Fig. 2, where a player enters the system already as the first job in the order starts processing and leaves after his own job is finished. Also, we include set-up times.

The time a job spends in the system consists of a set-up time depending on the job that is processed before him and his own processing time. The costs arising from this last part is constant over all orders. Hence, we just focus on the costs arising from set-up. So, for an order $\sigma \in \Pi(N)$ the corresponding costs $\gamma_i(\sigma)$ for player $i \in N$ are given by

$$\gamma_i(\sigma) = \alpha_i s_{\sigma(\sigma^{-1}(i)-1)}.$$

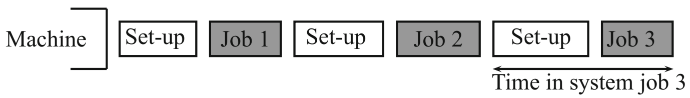


**Fig. 1** Time in system for JiT sequencing

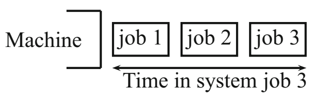


**Fig. 2** Time in system for standard sequencing

**Table 1** Partition of the player set

| | Costs | Set-up time |
|---|---|---|
| $N_h^H$ | $\alpha^H$ | $s^h$ |
| $N_l^H$ | $\alpha^H$ | $s^l$ |
| $N_h^L$ | $\alpha^L$ | $s^h$ |
| $N_l^L$ | $\alpha^L$ | $s^l$ |

For a coalition $S \in 2^N$, we set $\gamma_S(\sigma) = \sum_{i \in S} \gamma_i(\sigma)$. We call an order $\sigma^* \in \Pi(N)$ optimal for $N$ if $\gamma_N(\sigma^*) = \min\{\gamma_N(\sigma) \mid \sigma \in \Pi(N)\}$.

In this paper we restrict ourselves to the analysis of JiT sequencing situations with two different values for the set-up times and two different values for the cost per time unit. We denote by $JiT^{2,2}$ the class of all JiT sequencing situations satisfying this restriction. So, for every $(N, \alpha, s, s_0) \in JiT^{2,2}$, there exist $\alpha^H, \alpha^L \in \mathbb{R}_+, \alpha^H > \alpha^L$ such that for all $i \in N$ it holds that either $\alpha_i = \alpha^H$ or $\alpha_i = \alpha^L$. With respect to the set-up times, we assume there exist $s^h, s^l \in \mathbb{R}_+, s^h > s^l$, such that for all $i \in N \cup \{0\}$ it holds that either $s_i = s^h$ or $s_i = s^l$. We partition the set of players according to their characteristics as provided in Table 1, into sets $N_h^H$, $N_l^H$, $N_h^L$ and $N_l^L$. Note that the superscript refers to the cost per time unit, and the subscript refers to the set-up time. Also, throughout the paper uppercase $H$ and $L$ refer to cost per time unit and lowercase $h$ and $l$ to set-up time. We denote $N^H = N_h^H \cup N_l^H$, and define $N^L$, $N_h$ and $N_l$ in a similar way. For a subset $S \in 2^N$ we use a similar notation: $S_h^H = S \cap N_h^H$, $S^H = S \cap N^H$, etc.

Naturally, an interesting question is how we can identify whether an order is optimal or not. Also, if we can find sufficient conditions for this, could we use these conditions to construct an optimal order? As it turns out, we can indeed find such conditions and use these to obtain an algorithm that constructs an optimal order for every sequencing situation in $JiT^{2,2}$.

First we focus on the sufficient conditions. For this, we introduce the following additional notation. Given a JiT sequencing situation $(N, \alpha, s, s_0) \in JiT^{2,2}$ and an order $\sigma \in \Pi(N)$, define the following classes of neighboring pairs:

$$M^{hH}(\sigma) = \{(i, j) \in (N \cup \{0\}) \times N \mid s_i = s^h,\ \alpha_j = \alpha^H,\ \sigma^{-1}(i) = \sigma^{-1}(j) - 1\},$$
$$M^{lL}(\sigma) = \{(i, j) \in (N \cup \{0\}) \times N \mid s_i = s^l,\ \alpha_j = \alpha^L,\ \sigma^{-1}(i) = \sigma^{-1}(j) - 1\},$$
$$M^{hL}(\sigma) = \{(i, j) \in (N \cup \{0\}) \times N \mid s_i = s^h,\ \alpha_j = \alpha^L,\ \sigma^{-1}(i) = \sigma^{-1}(j) - 1\},$$

and

$$M^{lH}(\sigma) = \{(i, j) \in (N \cup \{0\}) \times N \mid s_i = s^l,\ \alpha_j = \alpha^H,\ \sigma^{-1}(i) = \sigma^{-1}(j) - 1\}.$$

Note that the first superscript indicates the set-up time and the second superscript indicates the cost level. For every $\sigma \in \Pi(N)$, we have the following equalities:

$$|M^{hH}(\sigma)| + |M^{lH}(\sigma)| = |N^H|, \tag{1}$$

and

$$|M^{lL}(\sigma)| + |M^{hL}(\sigma)| = |N^L|. \tag{2}$$

If $s_0 = s^h$, we have for every order $\sigma \in \Pi(N)$ that

$$|M^{lH}(\sigma)| + |M^{lL}(\sigma)| = |N_l| - \mathbb{1}_{[s_{\sigma(|N|)}=s^l]}, \tag{3}$$

and

$$|M^{hH}(\sigma)| + |M^{hL}(\sigma)| = |N_h| + \mathbb{1}_{[s_{\sigma(|N|)}=s^l]}, \tag{4}$$

since $\mathbb{1}_{[s_{\sigma(|N|)}=s^l]} = 1 - \mathbb{1}_{[s_{\sigma(|N|)}=s^h]}$, $s_0 = s^h$ and the set-up time of the last player in the order does not incur costs for a player in $N$. If $s_0 = s^l$, we have

$$|M^{lH}(\sigma)| + |M^{lL}(\sigma)| = |N_l| + \mathbb{1}_{[s_{\sigma(|N|)}=s^h]}, \tag{5}$$

and

$$|M^{hH}(\sigma)| + |M^{hL}(\sigma)| = |N_h| - \mathbb{1}_{[s_{\sigma(|N|)}=s^h]}, \tag{6}$$

for every order $\sigma \in \Pi(N)$. Again, note that $\mathbb{1}_{[s_{\sigma(|N|)}=s^h]} = 1 - \mathbb{1}_{[s_{\sigma(|N|)}=s^l]}$. The following theorem states sufficient conditions for an order to be optimal. In general, an order satisfying these sufficient conditions need not exist. However, Proposition 2.1 shows that only in the specific case where $N_h^H \cup N_l^L = \emptyset$ such an order does not exist.[1]

**Theorem 2.1** *Let $(N, \alpha, s, s_0) \in JiT^{2,2}$ and let $\sigma \in \Pi(N)$. If $s_{\sigma(|N|)} = \max_{i \in N} s_i$ and either $|M^{hH}(\sigma)| = 0$ or $|M^{lL}(\sigma)| = 0$, then $\sigma$ is optimal.*

The optimality conditions in Theorem 2.1 consist of two parts: the first condition states that it is optimal to place a player with highest set-up time possible at the last position. The second condition means that it is optimal to place players with low costs behind players with high set-up time and players with high costs behind players with low set-up time. These conditions are used in the following algorithm. The first condition is explicitly taken care of in step 2, the second condition is dealt with in step 3. Step 4 deals with these optimality conditions more implicitly, which is demonstrated in Example 2.1.

**Algorithm 1**
*Input: a sequencing situation $(N, \alpha, s, s_0) \in JiT^{2,2}$.*
*Output: an order $\tilde{\sigma} \in \Pi(N)$.*

*Step*1. Initialize $p = 1$ and $C_1^1 = N$.

[1] The proof of the results in this section can be found in the appendix.

*Step*2. Define

$$C_p^2 = \begin{cases} C_p^1 \backslash N_h & \text{if } |C_p^1 \cap N_h| = 1 \text{ and } p \neq |N|; \\ C_p^1 & \text{else.} \end{cases}$$

*Step*3. Define

$$C_p^3 = \begin{cases} C_p^2 \cap N^L & \text{if } s_{\tilde{\sigma}(p-1)} = s^h \text{ and } C_p^2 \cap N^L \neq \emptyset; \\ C_p^2 \cap N^H & \text{if } s_{\tilde{\sigma}(p-1)} = s^l \text{ and } C_p^2 \cap N^H \neq \emptyset; \\ C_p^2 & \text{else.} \end{cases}$$

*Step*4. Define

$$C_p^4 = \begin{cases} C_p^3 \cap N_l & \text{if } C_p^3 \cap N_l \neq \emptyset; \\ C_p^3 & \text{else.} \end{cases}$$

*Step*5. Choose a job $i \in C_p^4$ and define $\tilde{\sigma}(p) = i$.
*Step*6. If $p = |N|$, stop.
If $p < |N|$, set $p = p + 1$ and, subsequently, set $C_p^1 = C_{p-1}^1 \backslash \{\tilde{\sigma}(p-1)\}$. Next,
return to step 2.

The notation $\tilde{\sigma}$ is used for an order provided by the algorithm. The algorithm generates this order by filling up all positions in the order from front to back. For every position, the set of candidate players is narrowed down in a few steps. In the algorithm, $C_p^4 \subseteq C_p^3 \subseteq C_p^2 \subseteq C_p^1 \in 2^N$ are the sets of candidate players for the $p$th position in this order.

Roughly speaking, the algorithm puts the jobs in an alternating sequence, that is in a way that high set-up time meets low cost of spending a unit of time in the system and vice versa. Hereby it takes into account that a job with high set-up time should be left over for the last position in the sequence. The basic idea underlying this algorithm seems to work well in a setting with arbitrary weights and two setup times as well. Greedy filling positions from front to end by placing after a high setup time a job with lowest available weight and after a low setup time a job with highest available weight does not always result in an optimal order, however, even if a job with high set-up time would be left over for the last position in the sequence.[2] We leave this more general setting for further research.

*Example 2.1* Consider the JiT sequencing situation $\Psi = (N, \alpha, s, s_0)$, where we have $\alpha = (2, 2, 1, 1)$, $s = (3, 1, 3, 1)$ and $s_0 = 3$. We have $N_h^H = \{1\}$, $N_l^H = \{2\}$, $N_h^L = \{3\}$, and $N_l^L = \{4\}$. As $|N_h| = 2$, we have $C_1^2 = C_1^1 = N$ (see Table 2). In step 3 we obtain $C_1^3 = \{3, 4\}$ as $s_0 = s^h$. Step 4 further narrows down the set of candidate

[2] For example consider a 6-player situation with $\alpha = (1, 2, 3, 4, 5, 6)$ and $s = (1, 2, 1, 2, 1, 1)$ where these greedy approaches result in nonoptimal orders 1–6–5–4–2–3 and 1–6–5–4–3–2, with costs equal to 27 and 25, respectively. A possible optimal order of the jobs would be 2–1–5–6–3–4 with costs equal to 24.

**Table 2** Sets of candidate players in example 2.1

| $p$ | 1 | 2 | 3 | 4 |
|---|---|---|---|---|
| $C_p^1$ | $N$ | $\{1, 2, 3\}$ | $\{1, 3\}$ | $\{3\}$ |
| $C_p^2$ | $N$ | $\{1, 2, 3\}$ | $\{1, 3\}$ | $\{3\}$ |
| $C_p^3$ | $\{3, 4\}$ | $\{1, 2\}$ | $\{1\}$ | $\{3\}$ |
| $C_p^4$ | $\{4\}$ | $\{2\}$ | $\{1\}$ | $\{3\}$ |
| $\tilde{\sigma}(p)$ | 4 | 2 | 1 | 3 |

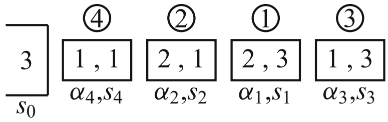


**Fig. 3** Order $\tilde{\sigma}$ provided by algorithm 1 in example 2.1

players for the first position, as $C_1^4 = \{4\}$. Therefore, we obtain $\tilde{\sigma}(1) = 4$. Now that player 4 is placed, we have $C_2^2 = C_2^1 = \{1, 2, 3\}$. In step 3 of iteration 2, we obtain $C_2^3 = \{1, 2\}$ and in step 4 we obtain $C_2^4 = \{2\}$ so $\tilde{\sigma}(2) = 2$. In the third iteration, $C_3^2 = C_3^1 = \{1, 3\}$ and $C_3^4 = C_3^3 = \{1\}$ so $\tilde{\sigma}(3) = 1$ and $\tilde{\sigma}(4) = 3$. It is easily seen that a player with high set-up time is placed last. Furthermore, players with high costs are placed behind players with low set-up time and the other way around (see Fig. 3). We obtain $\gamma_N(\tilde{\sigma}) = 10$ which is indeed optimal. Also, note the importance of step 4 of the algorithm: if we would place an arbitrary player in $C_2^3$ at position 2, we could have ended up with the order $\sigma'$ such that $\sigma'(1) = 4, \sigma'(2) = 1, \sigma'(3) = 2, \sigma'(4) = 3$, with $\gamma_N(\sigma') = 12$. $\triangleleft$

Now we are ready to prove that Algorithm 1 provides an optimal order, for every sequencing situation $(N, \alpha, s, s_0) \in JiT^{2,2}$.

**Theorem 2.2** *Let $\Psi = (N, \alpha, s, s_0) \in JiT^{2,2}$. Then Algorithm 1 provides an optimal order $\tilde{\sigma}$ for $N$.*

The proof of Theorem 2.2 implies the following.

**Proposition 2.1** *If $N_h^H \cup N_l^L \neq \emptyset$, then every order provided by Algorithm 1 satisfies the sufficient conditions of Theorem 2.1.*

## 3 JiT sequencing games

In the previous section we addressed the problem of finding an optimal order for $JiT^{2,2}$ sequencing situations. An additional question is how the total costs of such an optimal order should be allocated among the players. To answer this question, we will use the framework of transferable utilty games. Let us first recall some basic concepts within cooperative game theory used in the later analysis.

A transferable utility game $(N, v)$ is defined by a finite player set $N$ and a function $v$ on the set $2^N$ of all subsets of $N$ assigning to each coalition $S \in 2^N$ a value $v(S) \in \mathbb{R}$ such that $v(\emptyset) = 0$. The imputation set $I(v)$ of a game $(N, v)$ is given by all individually rational and efficient allocations, so

$$I(v) = \left\{ x \in \mathbb{R}^N \,\middle|\, \sum_{i \in N} x_i = v(N), x_i \geq v(\{i\}) \text{ for all } i \in N \right\}.$$

For a game $(N, v)$, the core $C(v)$ is defined as the set of those imputations, for which no coalition has an incentive to split off:

$$C(v) = \left\{ x \in \mathbb{R}^N \,\middle|\, \sum_{i \in N} x_i = v(N), \sum_{i \in S} x_i \geq v(S) \text{ for all } S \in 2^N \right\}.$$

A game $(N, v)$ is called balanced if its core is nonempty. A game $(N, v)$ is called convex if $v(S \cup \{i\}) - v(S) \leq v(T \cup \{i\}) - v(T)$ for all $i \in N$ and $S \subset T \subset N \backslash \{i\}$. Every convex game has a nonempty core. If for $i \in N$ and $j \in N$ it holds that $v(S \cup \{i\}) = v(S \cup \{j\})$ for all $S \in 2^{N \backslash \{i,j\}}$, then player $i$ and $j$ are called symmetric.

We define the excess of coalition $S \in 2^N$ with respect to allocation $x \in I(v)$ by $E(S, x) = v(S) - \sum_{i \in S} x_i$. The excess measures the dissatisfaction of coalition $S$ with respect to allocation $x$. Let $\omega(x) \in \mathbb{R}^{2^{|N|}}$ be the vector of excesses of $x \in I(v)$, arranged in weakly decreasing order. For a game $(N, v)$ such that $I(v) \neq \emptyset$, the nucleolus $\eta(v)$ (Schmeidler 1969) is the unique imputation $x \in I(v)$ such that $\omega(x)$ is lexicographically smaller than $\omega(y)$ for all $y \in I(v)$. So, the nucleolus is the individual rational and efficient allocation that minimizes the highest dissatisfactions in a hierarchical manner. For every game $(N, v)$ such that $C(v) \neq \emptyset$, we have $\eta(v) \in C(v)$. Furthermore, if player $i \in N$ and player $j \in N$ are symmetric in the game $(N, v)$ then $\eta_i(v) = \eta_j(v)$.

For the grand coalition we employ a worst case approach and consider the players in $N$ to be part of a larger system with players in the system before and after the players in $N$. The worst case approach comprises that we assume that the system is left behind with high set-up time by the players outside the grand coalition. We provide a game theoretic analysis of those instances of $JiT^{2,2}$ where $s_0 = s^h$, denoted by $JiT_h^{2,2}$. We assume that by cooperating, every coalition $S \in 2^N$ can form any order $\sigma \in \Pi(S)$ at the first $|S|$ spots in the sequence. Thus we employ a pessimistic view for both the grand coalition and for subcoalitions, in the sense that the set-up time for the first player in the order $\sigma$ equals $s_0 = s^h$. This setup allows us to measure the value of every coalition consistently over all coalitions, and independent of the players outside the coalition.[3]

Let $\Psi = (N, \alpha, s, s_0)$ be a JiT sequencing situation. We will define the costs for coalition $S \in 2^N$ as the costs of an optimal order in the JiT sequencing problem $(S, \alpha', s', s'_0)$, where $\alpha' \in \mathbb{R}^S$ and $s' \in \mathbb{R}^{S \cup \{0\}}$ are such that $\alpha'_i = \alpha_i$ for all $i \in S$, and $s'_i = s_i$ for all $i \in S \cup \{0\}$. Given a JiT sequencing situation $\Psi = (N, \alpha, s, s_0)$ and a coalition $S \in 2^N$, we denote by $\sigma_S^*$ an optimal order of the situation $(S, \alpha', s', s'_0)$. Hence, formally, the *JiT sequencing game* $(N, v^\Psi)$ is defined by

[3] If, on the other hand, one would assume that $s_0 = s^l$, this exogenous feature would cause free-rider problems and coalitional stability in the sense of the core cannot be obtained in general. In other words, the corresponding cooperative JiT sequencing game would not provide an adequate model for the allocation problem at hand: other techniques would have to be employed to arrive at suitable allocation proposals.

**Table 3** The JiT sequencing game $(N, v^{\Psi})$ of example 3.1

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| $S$ | $\{1\}$ | $\{2\}$ | $\{3\}$ | $\{4\}$ | $\{1,2\}$ | $\{1,3\}$ | $\{1,4\}$ | $\{2,3\}$ |
| $v^{\Psi}(S)$ | 0 | 0 | 0 | 0 | 4 | 0 | 4 | 2 |
| $S$ | $\{2,4\}$ | $\{3,4\}$ | $\{1,2,3\}$ | $\{1,2,4\}$ | $\{1,3,4\}$ | $\{2,3,4\}$ | $N$ | |
| $v^{\Psi}(S)$ | 4 | 2 | 4 | 8 | 4 | 6 | 8 | |

$$v^{\Psi}(S) = \sum_{i \in S} \gamma_i(\sigma^*_{\{i\}}) - \gamma_S(\sigma^*_S),$$

for all $S \in 2^N$. Clearly, $v^{\Psi}(\{i\}) = 0$ for every $i \in N$.

*Example 3.1* Reconsider the JiT sequencing situation of Example 2.1, where $\alpha = (2,2,1,1)$, $s = (3,1,3,1)$ and $s_0 = 3$. It is seen that $\gamma_1(\sigma^*_{\{1\}}) = \gamma_2(\sigma^*_{\{2\}}) = 6$ and $\gamma_3(\sigma^*_{\{3\}}) = \gamma_4(\sigma^*_{\{4\}}) = 3$. Take $S = \{1,2,4\}$. The optimal order $\sigma^*_S$ is such that $\sigma^*_S(1) = 4, \sigma^*_S(2) = 2$, and $\sigma^*_S(3) = 1$, which results in total costs $\gamma_S(\sigma^*_S) = 7$. Hence, we have $v^{\Psi}(S) = (6+6+3) - 7 = 8$. The JiT sequencing game $(N, v^{\Psi})$ is given by Table 3. Note that $(N, v^{\Psi})$ is not convex, since

$$v^{\Psi}(\{3,4\}) - v^{\Psi}(\{4\}) = 2 > 0 = v^{\Psi}(N) - v^{\Psi}(\{1,2,4\}).$$

$\triangleleft$

For $\Psi = (N, \alpha, s, s_0) \in JiT_h^{2,2}$ we can explicitly express the value of each coalition in terms of the number of players in the different player classes in the JiT sequencing situation.[4]

**Proposition 3.1** *Let $\Psi = (N, \alpha, s, s_0) \in JiT_h^{2,2}$. Then*

$$v^{\Psi}(S) = \begin{cases} (|S_l^H| + |S_l^L|)(s^h - s^l)\alpha^H & \text{if } S_h^H \neq \emptyset \text{ and } |S_h^H| \geq |S_l^L|; \\ |S_h^H|(s^h - s^l)(\alpha^H - \alpha^L) + |S_l^H|(s^h - s^l)\alpha^H & \text{if } S_h^H \neq \emptyset \text{ and } |S_h^H| < |S_l^L|; \\ \quad +|S_l^L|(s^h - s^l)\alpha^L & \\ |S_l^H|(s^h - s^l)\alpha^H + |S_l^L|(s^h - s^l)\alpha^L & \text{if } S_h^H = \emptyset, S_l^L \neq \emptyset \\ & \text{and } S_h^L \neq \emptyset; \\ |S_l^H|(s^h - s^l)\alpha^H + (|S_l^L| - 1)(s^h - s^l)\alpha^L & \text{if } S_h^H = \emptyset, S_l^L \neq \emptyset \\ & \text{and } S_h^L = \emptyset; \\ (|S_l^H| - 1)(s^h - s^l)\alpha^H + (s^h - s^l)\alpha^L & \text{if } S_h^H = \emptyset, S_l^L = \emptyset, S_l^H \neq \emptyset \\ & \text{and } S_h^L \neq \emptyset; \\ (|S_l^H| - 1)(s^h - s^l)\alpha^H & \text{if } S = S_l^H \text{ and } S \neq \emptyset; \\ 0 & \text{if } S = S_h^L, \end{cases}$$

*for all $S \in 2^N$.*

[4] The proof of this proposition can be found in the appendix.

We use the expressions from Proposition 3.1 to show that every JiT sequencing game has a nonempty core. To this end, we define the *large instance based allocation rule* $\theta$ on the class of JiT sequencing situations $JiT_h^{2,2}$.

**Definition 3.1** Let $\Psi = (N, \alpha, s, s_0) \in JiT_h^{2,2}$. Then, for all $i \in N$,

$$\theta_i(\Psi) = (s^h - s_i)\alpha_i + \begin{cases} \frac{(\alpha^H-\alpha^L)(s^h-s^l)}{2} & \text{if } i \in N_h^H \cup N_l^L \\ & \quad \text{and } |N_h^H| = |N_l^L|; \\ (\alpha^H - \alpha^L)(s^h - s^l) & \text{if } i \in N_h^H \text{ and } |N_h^H| < |N_l^L|, \\ & \text{or if } i \in N_l^L \\ & \quad \text{and } |N_h^H| > |N_l^L|; \\ -\frac{1}{|N_l^L|}(s^h - s^l)\alpha^L & \text{if } i \in N_l^L \text{ and } |N_h| = 0; \\ -\frac{1}{|N_l^H|}(\alpha^H - \alpha^L)(s^h - s^l) & \text{if } i \in N_l^H, |N_h^L| > 0 \text{ and} \\ & \quad |N_h^H| = |N_l^L| = 0; \\ -\frac{1}{|N_l^H|}(s^h - s^l)\alpha^H & \text{if } i \in N_l^H \text{ and } N = N_l^H; \\ 0 & \text{otherwise.} \end{cases}$$

The common part of the expression for $\theta(\Psi)$, $(s^h - s_i)\alpha_i$, gives an estimation of the cost savings that can be attributed to player $i$. This estimation is based on the marginal costs of player $i$ entering in a fictive, 'large' coalition. The part $s^h\alpha_i$ are the stand-alone costs of player $i$. Now assume there is an order $\sigma \in \Pi(N)$ where $s_{\sigma(k)} = s_i$ for some $k$. If player $i$ is placed in between player $\sigma(k)$ and player $\sigma(k+1)$, then the marginal costs equal $s_i\alpha_i$. Hence, we estimate the cost savings by $(s^h - s_i)\alpha_i$.

The second part serves as a correction to this estimation: a player in $N_h^H$ and a player in $N_l^L$ together are responsible for more cost savings than we already allocated to them. These additional cost savings go to the minority, the players in $N_h^H$ if $|N_h^H| < |N_l^L|$ and the players in $N_l^L$ if $|N_l^L| < |N_h^H|$, and is shared equally if $|N_h^H| = |N_l^L|$. Note that if $i \in N_h^H$ and $j \in N_l^L$ then $\theta_i(\Psi) + \theta_j(\Psi) = (s^h - s^l)\alpha^H$ and that for all $i \in N_h^L$, $\theta_i(\Psi) = 0$. The other corrections are due to boundary cases of the player set: for example, if there are no players in both $N_h^H$ and $N_l^L$, then the cost savings attributed to players in $N_l^H$ is overestimated, and is corrected.

For every JiT sequencing situation $\Psi \in JiT_h^{2,2}$, the large instance based allocation rule provides a core element for the corresponding game $(N, v^\Psi)$.

**Theorem 3.1** *Let $\Psi = (N, \alpha, s, s_0) \in JiT_h^{2,2}$. Then $\theta(\Psi) \in C(v^\Psi)$.*

*Proof* We consider four different cases, and use Proposition 3.1 and Definition 3.1 in each of these cases.

(i) Assume $|N_h^H| = 0$, $|N_l^L| = 0$ and $|N_h^L| > 0$. For $S \in 2^N$ we have

$$\begin{aligned}\theta_S(\Psi) - v^\Psi(S) &\geq \sum_{i \in S_l^H} \left( (s^h - s^l)\alpha^H - \frac{1}{|N_l^H|}(s^h - s^l)(\alpha^H - \alpha^L) \right) \\ &\quad -(|S_l^H| - 1)(s^h - s^l)\alpha^H - (s^h - s^l)\alpha^L \\ &= |S_l^H| \left( (s^h - s^l)\alpha^H - \frac{1}{|N_l^H|}(s^h - s^l)(\alpha^H - \alpha^L) \right) \\ &\quad -(|S_l^H| - 1)(s^h - s^l)\alpha^H - (s^h - s^l)\alpha^L \\ &= \left( 1 - \frac{|S_l^H|}{|N_l^H|} \right)(s^h - s^l)(\alpha^H - \alpha^L) \\ &\geq 0,\end{aligned}$$

with equality if $S = N$, and therefore $\theta(\Psi) \in C(v^\Psi)$.

(ii) Assume $|N_h^H| + |N_l^L| > 0$ and $|N_h| > 0$. For $S \in 2^N$ we have

$$\begin{aligned}\theta_S(\Psi) - v^\Psi(S) &\geq |S_l^H|(s^h - s^l)\alpha^H + |S_l^L|(s^h - s^l)\alpha^L \\ &\quad + \min\{|S_h^H|, |S_l^L|\}(s^h - s^l)(\alpha^H - \alpha^L) \\ &\quad -|S_l^H|(s^h - s^l)\alpha^H - |S_l^L|(s^h - s^l)\alpha^L \\ &\quad - \min\{|S_h^H|, |S_l^L|\}(s^h - s^l)(\alpha^H - \alpha^L) \\ &\geq 0,\end{aligned}$$

again with equality if $S = N$, and therefore $\theta(\Psi) \in C(v^\Psi)$.

(iii) Assume $|N_h| = 0$ and $|N_l^L| > 0$. For $S \in 2^N$ we have

$$\begin{aligned}\theta_S(\Psi) - v^\Psi(S) &\geq |S_l^H|(s^h - s^l)\alpha^H + |S_l^L|((s^h - s^l)\alpha^L - \frac{1}{|N_l^L|}(s^h - s^l)\alpha^L) \\ &\quad -|S_l^H|(s^h - s^l)\alpha^H - (|S_l^L| - 1)(s^h - s^l)\alpha^L \\ &= \left( 1 - \frac{|S_l^L|}{|N_l^L|} \right)(s^h - s^l)\alpha^L \\ &\geq 0,\end{aligned}$$

with equality if $S = N$, and therefore $\theta(\Psi) \in C(v^\Psi)$.

(iv) Finally, assume $|N_h^H| = |N_l^L| = |N_h^L| = 0$. For $S \in 2^N$ we have

$$\begin{aligned}\theta_S(\Psi) - v^\Psi(S) &= |S_l^H|((s^h - s^l)\alpha^H - \frac{1}{|N_l^H|}(s^h - s^l)\alpha^H) \\ &\quad -(|S_l^H| - 1)(s^h - s^l)\alpha^H\end{aligned}$$

**Table 4** The JiT sequencing game $(N, v^{\Psi})$ of example 3.2

| $S$ | $\{1\}$ | $\{2\}$ | $\{3\}$ | $\{1,2\}$ | $\{1,3\}$ | $\{2,3\}$ | $N$ |
|---|---|---|---|---|---|---|---|
| $v^{\Psi}(S)$ | 0 | 0 | 0 | 8 | 2 | 2 | 10 |

$$= \left(1 - \frac{|S_l^H|}{|N_l^H|}\right)(s^h - s^l)\alpha^H$$
$$\geq 0,$$

with equality if $S = N$, and therefore $\theta(\Psi) \in C(v^{\Psi})$.

□

*Example 3.2* Consider the JiT sequencing situation $\Psi = (N, \alpha, s, s_0) \in JiT_h^{2,2}$ such that $N = \{1, 2, 3\}$, $\alpha = (1, 4, 1)$, $s = (1, 1, 3)$ and $s_0 = 3$. This means that $N_l^L = \{1\}$, $N_l^H = \{2\}$, $N_h^L = \{3\}$, and $N_h^H = \emptyset$. The corresponding JiT sequencing game is provided in Table 4.

Clearly, since $3 \in N_h^L$, $\theta_3(\Psi) = 0$. Since $0 = |N_h^H| < |N_l^L| = 1$, and $1 \in N_l^L$, we have

$$\theta_1(\Psi) = (s^h - s_1)\alpha_1 = 2,$$

while, since $2 \in N_l^H$ and $N_l^L \neq \emptyset$

$$\theta_2(\Psi) = (s^h - s_2)\alpha_2 = 8.$$

Hence, $\theta(\Psi) = (2, 8, 0)$. It is readily checked that

$$C(v^{\Psi}) = Conv\{(8, 0, 2), (0, 8, 2), (2, 8, 0), (8, 2, 0)\}$$

while for the nucleolus, we have that

$$\eta(v^{\Psi}) = \left(4\frac{1}{2}, 4\frac{1}{2}, 1\right). \tag{7}$$

◁

## 4 Complete JiT sequencing situations

In this section we analyze the core and the allocation prescribed by the large instance based allocation rule $\theta$ for *complete* JiT sequencing games, i.e., JiT sequencing games corresponding to JiT sequencing situations for which $N_h^H \neq \emptyset$, $N_l^H \neq \emptyset$, $N_h^L \neq \emptyset$, and $N_l^L \neq \emptyset$.

We first reformulate $\theta$ for the specific case of a complete JiT sequencing situation.

**Lemma 4.1** *Let $\Psi = (N, \alpha, s, s_0) \in JiT_h^{2,2}$ be complete. Then*

*(i) If $i \in N_h^H$,*

$$\theta_i(\Psi) = \begin{cases} (\alpha^H - \alpha^L)(s^h - s^l) & \text{if } |N_h^H| < |N_l^L|; \\ \frac{(\alpha^H - \alpha^L)(s^h - s^l)}{2} & \text{if } |N_h^H| = |N_l^L|; \\ 0 & \text{if } |N_h^H| > |N_l^L|. \end{cases}$$

*(ii) If $i \in N_l^L$,*

$$\theta_i(\Psi) = (s^h - s^l)\alpha^L + \begin{cases} 0 & \text{if } |N_h^H| < |N_l^L|; \\ \frac{(\alpha^H - \alpha^L)(s^h - s^l)}{2} & \text{if } |N_h^H| = |N_l^L|; \\ (\alpha^H - \alpha^L)(s^h - s^l) & \text{if } |N_h^H| > |N_l^L|. \end{cases}$$

*(iii) If $i \in N_l^H$,*

$$\theta_i(\Psi) = (s^h - s^l)\alpha^H.$$

*(iv) If $i \in N_h^L$,*

$$\theta_i(\Psi) = 0.$$

Our first result on the core of complete JiT sequencing situations is the following.

**Theorem 4.1** *Let $\Psi = (N, \alpha, s, s_0) \in JiT_h^{2,2}$ be complete and such that $|N_h^H| \neq |N_l^L|$. Then $\theta(\Psi)$ is the unique core element of $(N, v^\Psi)$.*

*Proof* For every $x \in C(v^\Psi)$ and $i \in N$ it holds that $x_i \leq v^\Psi(N) - v^\Psi(N\backslash\{i\})$. By Theorem 3.1, $\theta(\Psi) \in C(v^\Psi)$. It suffices to show that $\theta_i(\Psi) = v^\Psi(N) - v^\Psi(N\backslash\{i\})$ for every $i \in N$, as this implies that for every $x \in C(v^\Psi)$ with $x \neq \theta(\Psi)$ we have $\sum_{i\in N} x_i < \sum_{i\in N} \theta(\Psi) = v(N)$ which contradicts the core condition $\sum_{i\in N} x_i = v(N)$.

First consider the case $|N_h^H| > |N_l^L|$. By Proposition 3.1, Definition 3.1, and Lemma 4.1 we obtain that

$$v^\Psi(N) - v^\Psi(N\backslash\{i\}) = \begin{cases} 0 & \text{if } i \in N_h^H; \\ (s^h - s^l)\alpha^H & \text{if } i \in N_l^L; \\ (s^h - s^l)\alpha^H & \text{if } i \in N_l^H; \\ 0 & \text{if } i \in N_h^L; \end{cases}$$
$$= \theta_i(\Psi).$$

Now consider the case $|N_l^L| > |N_h^H|$. Then, by Proposition 3.1, Definition 3.1, and Lemma 4.1,

$$\begin{aligned} v^{\Psi}(N) - v^{\Psi}(N\backslash\{i\}) &= \begin{cases} (s^h - s^l)(\alpha^H - \alpha^L) & \text{if } i \in N_h^H; \\ (s^h - s^l)\alpha^L & \text{if } i \in N_l^L; \\ (s^h - s^l)\alpha^H & \text{if } i \in N_l^H; \\ 0 & \text{if } i \in N_h^L; \end{cases} \\ &= \theta_i(\Psi). \end{aligned}$$

□

To describe the core for complete JiT sequencing games with $|N_h^H| = |N_l^L| (> 1)$ we define an upper vector $\overline{\theta} \in \mathbb{R}^N$ and a lower vector $\underline{\theta} \in \mathbb{R}^N$. For a JiT sequencing situation $\Psi = (N, \alpha, s, s_0) \in JiT_h^{2,2}$ with $|N_h^H| = |N_l^L| > 1$, and define, for all $j \in N$,

$$\overline{\theta}_j(\Psi) = (s^h - s_j)\alpha_j + \begin{cases} (s^h - s^l)(\alpha^H - \alpha^L) & \text{if } j \in N_h^H; \\ 0 & \text{if } j \in N\backslash N_h^H, \end{cases}$$

$$\underline{\theta}_j(\Psi) = (s^h - s_j)\alpha_j + \begin{cases} (s^h - s^l)(\alpha^H - \alpha^L) & \text{if } j \in N_l^L; \\ 0 & \text{if } j \in N\backslash N_l^L. \end{cases}$$

Then we have the following result.[5]

**Theorem 4.2** *Let $\Psi = (N, \alpha, s, s_0) \in JiT_h^{2,2}$ be complete and such that $|N_h^H| = |N_l^L| > 1$. Then $C(v^{\Psi}) = Conv\{\overline{\theta}(\Psi), \underline{\theta}(\Psi)\}$ and*

$$\theta(\Psi) = \frac{1}{2}(\overline{\theta}(\Psi) + \underline{\theta}(\Psi)).$$

*Moreover, $\theta(\Psi)$ is equal to the nucleolus $\eta(v^{\Psi})$.*

*Proof* First we establish that $Conv\{\overline{\theta}(\Psi), \underline{\theta}(\Psi)\} \subseteq C(v^{\Psi})$, by showing that $\overline{\theta}(\Psi) \in C(v^{\Psi})$ and $\underline{\theta}(\Psi) \in C(v^{\Psi})$.

Let $S \in 2^N$. Then we have that

$$\begin{aligned} \sum_{i\in S} \overline{\theta}_i(\Psi) - v^{\Psi}(S) &= \sum_{i\in S}(s^h - s_i)\alpha_i + |S_h^H|(s^h - s^l)(\alpha^H - \alpha^L) - v^{\Psi}(S) \\ &\geq |S_l^H|(s^h - s^l)\alpha^H + |S_l^L|(s^h - s^l)\alpha^L + |S_h^H|(s^h - s^l)(\alpha^H - \alpha^L) \\ &\quad - \left(|S_l^H|(s^h - s^l)\alpha^H + |S_l^L|(s^h - s^l)\alpha^L\right) \\ &\quad + \min\{|S_h^H|, |S_l^L|\}(s^h - s^l)(\alpha^H - \alpha^L) \end{aligned}$$

[5] In fact, the proof of this theorem does not use completeness of the underlying JiT sequencing situation. The results described in the theorem therefore hold for a wider class of JiT sequencing situations.

$$= \left(|S_h^H| - \min\{|S_h^H|, |S_l^L|\}\right)(s^h - s^l)(\alpha^H - \alpha^L)$$
$$\geq 0.$$

Note that the first inequality follows from Proposition 3.1, and that for $S = N$ the two inequalities hold with equality. Hence, $\overline{\theta}(\Psi) \in C(v^\Psi)$. A similar reasoning shows that $\overline{\theta}(\Psi) \in C(v^\Psi)$.

Now we show that $C(v^\Psi) \subseteq Conv\{\overline{\theta}(\Psi), \underline{\theta}(\Psi)\}$. It suffices to show that for every $x \in C(v^\Psi)$ it holds that:

(i) $x_i \geq 0$ for every $i \in N$.
(ii) $x_i = 0$ for every $i \in N_h^L$.
(iii) $x_i = (s^h - s^l)\alpha^H$ for every $i \in N_l^H$.
(iv) $x_i + x_j = (s^h - s^l)\alpha^H$ for every $i \in N_h^H$ and $j \in N_l^L$ and, therefore, $x_i = x_k$ for all $i, k \in N_h^H$, and $x_j = x_r$ for all $j, r \in N_l^L$.
(v) $(s^h - s^l)\alpha^L \leq x_i \leq (s^h - s^l)\alpha^H$ for every $i \in N_l^L$.

Note that (iv) and (v) together imply $0 \leq x_i \leq (s^h - s^l)(\alpha^H - \alpha^L)$ for every $i \in N_h^H$. We prove (i)–(v) point by point. Take $x \in C(v^\Psi)$.

(i) As $x_i \geq v(\{i\})$ and $v(\{i\}) = 0$ for every $i \in N$, we have $x_i \geq 0$ for every $i \in N$.
(ii) As $v^\Psi(N \backslash N_h^L) = v^\Psi(N)$, we obtain $x_i = 0$ for all $i \in N_h^L$.
(iii) Take $i \in N_l^H$, and take $j \in N_h^H$, $k \in N_l^L$. Then

$$\begin{aligned}\sum_{r\in N} x_r + x_i &= \sum_{r\in N_h^H \cup\{i\}} x_r + \sum_{r\in N\backslash N_h^H} x_r \\ &\geq v^\Psi(\{i, j, k\}) + v^\Psi(N\backslash\{j, k\}) \\ &= 2(s^h - s^l)\alpha^H + (|N_l^H| + |N_l^L|)(s^h - s^l)\alpha^H \\ &= v^\Psi(N) + (s^h - s^l)\alpha^H \\ &= \sum_{r\in N} x_r + (s^h - s^l)\alpha^H.\end{aligned}$$

The second equality holds by the observation that $N_h^H \cap (N\backslash\{j, k\}) \neq \emptyset$ and $N_l^L \cap (N\backslash\{j, k\}) \neq \emptyset$, which means that $N\backslash\{j, k\}$ is contained in the first case of Proposition 3.1. We obtain $x_i \geq (s^h - s^l)\alpha^H$ and using Proposition 3.1 we have $v^\Psi(N) - v^\Psi(N\backslash\{i\}) = (s^h - s^l)\alpha^H$. Therefore, $x_i = (s^h - s^l)\alpha^H$.

(iv) Take $i \in N_h^H$ and $j \in N_l^L$. As $v^\Psi(\{i, j\}) = (s^h - s^l)\alpha^H = v^\Psi(N) - v^\Psi(N\backslash\{i, j\})$ we have $x_i + x_j = (s^h - s^l)\alpha^H$.

(v) Take $j \in N_h^H$ and $i, k \in N_l^L$ with $j \neq k$. The observation that $v(\{i, j, k\}) = (s^h - s^l)(\alpha^H - \alpha^L) + 2(s^h - s^l)\alpha^L$ and $v(N\backslash\{j, k\}) = (|N_l^H| + |N_l^L| - 1)(s^h - s^l)\alpha^H$ leads with a similar reason as with $(iii)$ to $x_i \geq (s^h - s^l)\alpha^L$. As $v^\Psi(\{i, j\}) = (s^h - s^l)\alpha^H = v^\Psi(N) - v^\Psi(N\backslash\{i, j\})$ Furthermore, since

$$v(N) - v(N\backslash N_l^L) = |N_l^L|(s^h - s^l)\alpha^H,$$

it follows that $x_i \leq (s^h - s^l)\alpha^H$.

Lastly, from Lemma 4.1 we have

$$\theta_i(\Psi) = (s^h - s_i)\alpha_i + \begin{cases} \frac{(\alpha^H - \alpha^L)(s^h - s^l)}{2} & \text{if } i \in N_h^H \cup N_l^L \\ 0 & \text{if } i \in N_l^H \cup N_h^L, \end{cases}$$

so $\theta(\Psi) = \frac{1}{2}(\overline{\theta}(\Psi) + \underline{\theta}(\Psi))$ follows directly from the definition of $\overline{\theta}(\Psi)$ and $\underline{\theta}(\Psi)$.

Finally, we show that $\theta(\Psi) = \eta(v^\Psi)$ by showing that $\omega(\theta(\Psi)) \leq_L \omega(x)$ for every $x \in C(v^\Psi)$. By Theorem 4.2, $C(v^\Psi) = Conv\{\overline{\theta}(\Psi), \underline{\theta}(\Psi)\}$. So, take $c \in [0, 1]$ and define $x^c = c\overline{\theta}(\Psi) + (1-c)\underline{\theta}(\Psi)$. By Lemma 3.1, the excesses are

$$E(S, x^c) = \begin{cases} -c(|S_h^H| - |S_l^L|)A & \text{if } S_h^H \neq \emptyset \text{ and } |S_h^H| \geq |S_l^L|; \\ -(1-c)(|S_l^L| - |S_h^H|)A & \text{if } S_h^H \neq \emptyset \text{ and } |S_h^H| < |S_l^L|; \\ -(1-c)|S_l^L|A & \text{if } S_h^H = \emptyset,\ S_l^L \neq \emptyset \text{ and } S_h^L \neq \emptyset; \\ -(1-c)|S_l^L|A - (s^h - s^l)\alpha^L & \text{if } S_h^H = \emptyset,\ S_l^L \neq \emptyset \text{ and } S_h^L = \emptyset; \\ -A & \text{if } S_h^H = \emptyset,\ S_l^L = \emptyset,\ S_l^H \neq \emptyset \text{ and } S_h^L \neq \emptyset; \\ -(s^h - s^l)\alpha^H & \text{if } S_h^H = \emptyset,\ S_l^L = \emptyset,\ S_l^H \neq \emptyset \text{ and } S_h^L = \emptyset; \\ 0 & \text{if } S_h^H = \emptyset,\ S_l^L = \emptyset \text{ and } S_l^H = \emptyset, \end{cases}$$

where $A = (s^h - s^l)(\alpha^H - \alpha^L) > 0$.

It is readily checked that the highest excess equals 0. This excess occurs, independent of the value of $c$, for every coalition $S \in 2^N$ such that either $|S_h^H| = |S_l^L| > 0$ or $S = S_h^L$. For $c = 0$ or $c = 1$, there are additional coalitions with excess equal to zero whereas for $c \in (0, 1)$ all other coalitions have a negative excess. Hence, both $x^0 \neq \eta(v^\Psi)$ and $x^1 \neq \eta(v^\Psi)$. Since $-(s^h - s^l)\alpha^H < 0$ and $-(s^h - s^l)\alpha^L < 0$, for $c \in (0, 1)$ the second highest excess equals either $-cA$ or $-(1-c)A$, or a multiple of these values. Hence, the second highest excess is minimized for $c = \frac{1}{2}$, implying that $\eta(v) = x^{\frac{1}{2}} = \theta(\Psi)$. □

Note that both Theorems 4.1 and 4.2 do not cover the case of a complete JiT sequencing situation with $|N_h^H| = |N_l^L| = 1$. The next example shows that in this case the core need not be one point or a line segment and that the coincidence between the large instance based rule and the nucleolus is lost.

*Example 4.1* Reconsider the complete JiT sequencing situation $\Psi = (N, \alpha, s, s_0) \in JiT_h^{2,2}$ analyzed before in Examples 2.1 and 3.1. Here $|N_h^H| = |N_l^L| = 1$. It is readily checked that

$$C(v^\Psi) = Conv\{(2, 4, 0, 2), (2, 2, 0, 4), (0, 4, 0, 4)\}$$

while

$$\overline{\theta}(\Psi) = (2, 4, 0, 2),\ \underline{\theta}(\Psi) = (0, 4, 0, 4), \quad \text{and} \quad \theta(\Psi) = (1, 4, 0, 3)$$

and

$$\eta(\Psi) = \left(1\frac{1}{3}, 3\frac{1}{3}, 0, 3\frac{1}{3}\right).$$

◁

**Appendix: Proofs**

*Proof of Theorem 2.1* Assume $s_0 = s^h$ and first consider the case where $\max_{i\in N} s_i = s^l$. This implies that $|N_h^H| = |N_h^L| = 0$. For $\sigma \in \Pi(N)$ we obtain

$$\gamma_N(\sigma) = \sum_{i\in N} s^l\alpha_i + (s_0 - s^l)\alpha_{\sigma(1)} = \sum_{i\in N} s^l\alpha_i + (s^h - s^l)\alpha_{\sigma(1)}.$$

Assume there exists an order $\sigma' \in \Pi(N)$ such that $|M^{hH}(\sigma')| = 0$, and take such a $\sigma' \in \Pi(N)$. As $s_0 = s^h$, we obtain that $\alpha_{\sigma'(1)} = \alpha^L$ and therefore $\sigma'(1) \in N_l^L$. Hence, for all $\sigma \in \Pi(N)$

$$\gamma_N(\sigma') = \sum_{i\in N} s^l\alpha_i + (s^h - s^l)\alpha^L \leq \gamma_N(\sigma),$$

so $\sigma'$ is optimal.

Now assume there exists an order $\sigma'' \in \Pi(N)$ such that $|M^{lL}(\sigma'')| = 0$ and $|M^{hH}(\sigma'')| > 0$, and take such a $\sigma'' \in \Pi(N)$. Then either $|N_l^L| = 0$, which means that $N = N_l^H$ and every order is optimal, or $|N_l^L| = 1$ with $\sigma'(1) \in N_l^L$. In the last case $\gamma_N(\sigma'') = \sum_{i\in N} s^l\alpha_i + (s_0 - s^l)\alpha^L \leq \gamma_N(\sigma)$ for all $\sigma \in \Pi(N)$, and $\sigma''$ is optimal.

Now consider the case where $\max_{i\in N} s_i = s^h$. Take an arbitrary $\sigma \in \Pi(N)$. Take $B, D \in \mathbb{N}$ such that $B = |M^{hH}(\sigma)|$ and $D = |M^{lL}(\sigma)|$. Note that by Eqs. (2) and (4) we have

$$B - D = |M^{hH}(\sigma)| - |M^{lL}(\sigma)| = |N_h| - |N^L| + \mathbb{1}_{[s_{\sigma(|N|)}=s^l]}.$$

By (3) and (4) it holds that

$$\begin{aligned}
\gamma_N(\sigma) &= |M^{hH}(\sigma)|s^h\alpha^H + |M^{lH}(\sigma)|s^l\alpha^H + |M^{hL}(\sigma)|s^h\alpha^L + |M^{lL}(\sigma)|s^l\alpha^L \\
&= Bs^h\alpha^H + (|N_l| - \mathbb{1}_{[s_{\sigma(|N|)}=s^l]} - D)s^l\alpha^H \\
&\quad + (|N_h| + \mathbb{1}_{[s_{\sigma(|N|)}=s^l]} - B)s^h\alpha^L + Ds^l\alpha^L \\
&\geq (B - \min\{B, D\})s^h\alpha^H + (|N_l| - \mathbb{1}_{[s_{\sigma(|N|)}=s^l]} - D + \min\{B, D\})s^l\alpha^H \\
&\quad + (|N_h| + \mathbb{1}_{[s_{\sigma(|N|)}=s^l]} - B + \min\{B, D\})s^h\alpha^L + (D - \min\{B, D\})s^l\alpha^L \\
&= \max\{0, |N_h| - |N^L| + \mathbb{1}_{[s_{\sigma(|N|)}=s^l]}\}s^h\alpha^H \\
&\quad + \min\{|N_l| - \mathbb{1}_{[s_{\sigma(|N|)}=s^l]}, |N^H|\}s^l\alpha^H \\
&\quad + \min\{|N^L|, |N_h| + \mathbb{1}_{[s_{\sigma(|N|)}=s^l]}\}s^h\alpha^L \\
&\quad + \max\{|N^L| - |N_h| - \mathbb{1}_{[s_{\sigma(|N|)}=s^l]}, 0\}s^l\alpha^L \\
&\geq \max\{0, |N_h| - |N^L|\}s^h\alpha^H + \min\{|N_l|, |N^H|\}s^l\alpha^H \\
&\quad + \min\{|N^L|, |N_h|\}s^h\alpha^L + \max\{|N^L| - |N_h|, 0\}s^l\alpha^L,
\end{aligned}$$

where the first inequality follows from the observation that $(s^h - s^l)(\alpha^H - \alpha^L) > 0$. If $|N_h| - |N^L| \geq 0$, and therefore $|N_l| - |N^H| \leq 0$, then the second inequality follows from $(s^h - s^l)\alpha^H > 0$. If $|N_h| - |N^L| < 0$, then the second inequality follows from $(s^h - s^l)\alpha^L > 0$. The first inequality holds with equality if either $B = 0$ or $D = 0$, and the second inequality holds with equality if $s_{\sigma(|N|)} = s^h$. This shows that every order $\sigma \in \Pi(N)$ with $s_{\sigma(|N|)} = s^h$ and either $|M^{hH}(\sigma)| = 0$ or $|M^{lL}(\sigma)| = 0$ is optimal.

Next, assume $s_0 = s^l$ and first consider the case where $\max_{i \in N} s_i = s^l$. In this case, for any order $\sigma \in \Pi(N)$ we have

$$\gamma_N(\sigma) = \sum_{i \in N} s^l \alpha_i.$$

So, every order is optimal.

Now consider the case where $\max_{i \in N} s_i = s^h$. Take an arbitrary $\sigma \in \Pi(N)$. Take $B, D \in \mathbb{N}$ such that $B = |M^{hH}(\sigma)|$ and $D = |M^{lL}(\sigma)|$. Note that by Eq. (2) and (6) we have

$$B - D = |M^{hH}(\sigma)| - |M^{lL}(\sigma)| = |N_h| - |N^L| - \mathbb{1}_{[s_{\sigma(|N|)}=s^h]}.$$

By (5) and (6) it holds that

$$\begin{aligned}
\gamma_N(\sigma) &= |M^{hH}(\sigma)|s^h\alpha^H + |M^{lH}(\sigma)|s^l\alpha^H + |M^{hL}(\sigma)|s^h\alpha^L + |M^{lL}(\sigma)|s^l\alpha^L \\
&= Bs^h\alpha^H + (|N_l| + \mathbb{1}_{[s_{\sigma(|N|)}=s^h]} - D)s^l\alpha^H \\
&\quad + (|N_h| - \mathbb{1}_{[s_{\sigma(|N|)}=s^h]} - B)s^h\alpha^L + Ds^l\alpha^L \\
&\geq (B - \min\{B, D\})s^h\alpha^H + (|N_l| + \mathbb{1}_{[s_{\sigma(|N|)}=s^h]} - D + \min\{B, D\})s^l\alpha^H \\
&\quad + (|N_h| - \mathbb{1}_{[s_{\sigma(|N|)}=s^h]} - B + \min\{B, D\})s^h\alpha^L + (D - \min\{B, D\})s^l\alpha^L \\
&= \max\{0, |N_h| - |N^L| - \mathbb{1}_{[s_{\sigma(|N|)}=s^h]}\}s^h\alpha^H \\
&\quad + \min\{|N_l| + \mathbb{1}_{[s_{\sigma(|N|)}=s^h]}, |N^H|\}s^l\alpha^H \\
&\quad + \min\{|N^L|, |N_h| - \mathbb{1}_{[s_{\sigma(|N|)}=s^h]}\}s^h\alpha^L \\
&\quad + \max\{|N^L| - |N_h| + \mathbb{1}_{[s_{\sigma(|N|)}=s^h]}, 0\}s^l\alpha^L \\
&\geq \max\{0, |N_h| - |N^L| - 1\}s^h\alpha^H + \min\{|N_l| + 1, |N^H|\}s^l\alpha^H \\
&\quad + \min\{|N^L|, |N_h| - 1\}s^h\alpha^L + \max\{|N^L| - |N_h| + 1, 0\}s^l\alpha^L,
\end{aligned}$$

where the first inequality follows from the observation that $(s^h - s^l)(\alpha^H - \alpha^L) > 0$. If $|N_h| - |N^L| \leq 0$, then the second inequality follows from $(s^l - s^h)\alpha^L < 0$. If $|N_h| - |N^L| > 0$, then the second inequality follows from $(s^l - s^h)\alpha^H < 0$. The first inequality holds with equality if either $B = 0$ or $D = 0$, and the second inequality holds with equality if $s_{\sigma(|N|)} = s^h$.

Hence for both values of $\sigma_0$ every order $\sigma \in \Pi(N)$ with $s_{\sigma(|N|)} = s^h$ and either $|M^{hH}(\sigma)| = 0$ or $|M^{lL}(\sigma)| = 0$ is optimal. $\square$

*Proof of Theorem 2.2* Let $\tilde{\sigma} \in \Pi(N)$ be an order provided by Algorithm 1. In Step 2 of the algorithm, it is made sure that there is always a player with the highest available set-up time left to place at the last position. Hence, $s_{\tilde{\sigma}(|N|)} = s^h$, unless $N_h^H \cup N_h^L = \emptyset$ which implies that there is in fact only one value for $s_i$ and $s_{\tilde{\sigma}(|N|)} = s^l = \max_{i \in N} s_i$.

We prove that either optimality of $\tilde{\sigma}$ follows directly from Theorem 2.1, i.e., $|M^{hH}(\tilde{\sigma})| = 0$ or $|M^{lL}(\tilde{\sigma})| = 0$, or $N_h^H \cup N_l^L = \emptyset$. For the latter case we show that $\tilde{\sigma}$ is optimal as well.

Assume that optimality of $\tilde{\sigma}$ does not follow directly from Theorem 2.1, i.e., $|M^{hH}(\tilde{\sigma})| > 0$ and $|M^{lL}(\tilde{\sigma})| > 0$. Then there exist $p, r \in \{0, \ldots, |N| - 1\}$ such that $(\tilde{\sigma}(p), \tilde{\sigma}(p+1)) \in M^{hH}(\tilde{\sigma})$ and $(\tilde{\sigma}(r), \tilde{\sigma}(r+1)) \in M^{lL}(\tilde{\sigma})$.

Assume $r < p$. According to the algorithm, job $\tilde{\sigma}(r+1)$ is only placed behind job $\tilde{\sigma}(r)$ if there is no job $j$ with $\alpha_j = \alpha^H$ left that is not yet placed, or there is only one job $j$ with $\alpha_j = \alpha^H$ left, but this job has to be reserved for the last spot because it is the only remaining job with high set-up time. In the first case, we have a contradiction, since job $\tilde{\sigma}(p+1)$ is not yet placed. The second case also results in a contradiction, since both $s_{\tilde{\sigma}(p)} = s^h$ and $s_{\tilde{\sigma}(|N|)} = s^h$.

Now assume $p < r$. According to the algorithm, job $\tilde{\sigma}(p+1)$ is only placed behind job $\tilde{\sigma}(p)$ if there is no job $j$ with $\alpha_j = \alpha^L$ left that is not yet placed, or there is only one job $j$ with $\alpha_j = \alpha^L$ left, but this job has to be reserved for the last spot because it is the only remaining job with high set-up time. In the first case, we have a contradiction, since job $\tilde{\sigma}(r+1)$ is not yet placed.

The second case can only hold if $r+1 = |N|$. For all jobs $i \in \{\tilde{\sigma}(p+1), \ldots, \tilde{\sigma}(r)\}$ it then must hold that $s_i = s^l$, otherwise job $\tilde{\sigma}(r+1)$ would have been placed at position $p+1$. Furthermore, $\alpha_i = \alpha^H$ otherwise job $i$ would have been placed at position $p+1$ as this would avoid the combination of $s^h$ and $\alpha^H$. So, we obtain that $i \in N_l^H$ for all $i \in \{\tilde{\sigma}(p+1), \ldots, \tilde{\sigma}(r)\}$. Since the algorithm first places the jobs in $N_l^H$ before placing the jobs in $N_h^H$, and $\tilde{\sigma}(|N|) \notin N_h^H$, we obtain that $N_h^H = \emptyset$ and therefore $\tilde{\sigma}(p) \in N_h^L$. Furthermore, if there existed a job $i \in N_l^L$ then the algorithm would place every job in $N_l^H$ directly behind this job. But since $\tilde{\sigma}(p) \notin N_l^L$ this implies that $N_l^L = \emptyset$. Hence, the second case only allows players in $N_l^H$ and $N_h^L$, so $N_h^H \cup N_l^L = \emptyset$. If $s_0 = s^l$, then the algorithm first places all players in $N_l^H$ and then all players in $N_h^L$, which contradicts $\tilde{\sigma}(s) \in N_h^L$. So, $s_0 = s^h$ and the solution provided by the algorithm for this situation (first all players in $N_h^L$ but one, then all players in $N_l^H$ and finally the last player in $N_h^L$) is clearly optimal. □

*Proof of Proposition 3.1* Here, we will only prove the first case, the other cases follow from a similar reasoning. First of all, we have

$$\sum_{i \in S} \gamma_i(\sigma^*_{\{i\}}) = (|S_h^H| + |S_l^H|) s^h \alpha^H + (|S_h^L| + |S_l^L|) s^h \alpha^L,$$

for every $S \in 2^N$. □

Take $S \in 2^N$ such that $S_h^H \neq \emptyset$ and $|S_h^H| \geq |S_l^L|$. Since $S_h^H \neq \emptyset$, it holds for every (optimal) order $\tilde{\sigma}_S$ provided by Algorithm 1 that $s_{\sigma^*_S(|S|)} = s^h$. Furthermore, since

$S_h^H \neq \emptyset$ we have by Proposition 2.1 that either $|M^{hH}(\tilde{\sigma}_S)| = 0$ or $|M^{lL}(\tilde{\sigma}_S)| = 0$. It must hold that $|M^{lL}(\tilde{\sigma}_S)| = 0$, since $|S_h^H| \geq |S_l^L|$ together with (1) and (3) implies that $|M^{hH}(\tilde{\sigma}_S)| \geq |M^{lL}(\tilde{\sigma}_S)|$. So, we have

$$\begin{aligned}\gamma_S(\tilde{\sigma}_S) &= |M^{hH}(\tilde{\sigma}_S)|s^h\alpha^H + |M^{lH}(\tilde{\sigma}_S)|s^l\alpha^H + |M^{hL}(\tilde{\sigma}_S)|s^h\alpha^L + |M^{lL}(\tilde{\sigma}_S)|s^l\alpha^L \\ &= (|S_h^H| - |S_l^L|)s^h\alpha^H + (|S_l^H| + |S_l^L|)s^l\alpha^H + (|S_h^L| + |S_l^L|)s^h\alpha^L,\end{aligned}$$

and we may conclude that

$$\begin{aligned}v^\Psi(S) &= \sum_{i \in S} \gamma_i(\tilde{\sigma}_{\{i\}}) - \gamma_S(\tilde{\sigma}_S) \\ &= (|S_h^H| + |S_l^H|)s^h\alpha^H + (|S_h^L| + |S_l^L|)s^h\alpha^L, \\ &\quad - \Big((|S_h^H| - |S_l^L|)s^h\alpha^H + (|S_l^H| + |S_l^L|)s^l\alpha^H + (|S_h^L| + |S_l^L|)s^h\alpha^L\Big) \\ &= (|S_l^H| + |S_l^L|)(s^h - s^l)\alpha^H.\end{aligned}$$

## References

Ahn B, Hyun J (1990) Single facility multi-class job scheduling. Comput Oper Res 17:265–272
Borm P, Fiestras-Janeiro G, Hamers H, Sanchez E, Voorneveld M (2002) On the convexity of games corresponding to sequencing situations with due dates. Eur J Oper Res 136:616–634
Burkard R, Deiňeko V, van Dal R, Van der Veen J, Woeginger G (1998) Well-solvable special cases of the traveling salesman problem: a survey. SIAM Rev 40:496–546
Calleja P, Borm P, Hamers H, Klijn F, Slikker M (2002) On a new class of parallel sequencing situations and related games. Ann Oper Res 109:263–276
Çiftçi B (2009) A cooperative approach to sequencing and connection problems. Ph. D. thesis, Tilburg University, Tilburg
Curiel I, Hamers H, Klijn F (2002) Sequencing games: a survey. In: Borm P, Peters H (eds) Chapters in game theory. Kluwer Academic Publishers, Dordrecht, pp 27–50
Curiel I, Pederzoli G, Tijs S (1989) Sequencing games. Eur J Oper Res 40:344–351
Gupta J (1988) Single facility scheduling with multiple job classes. Eur J Oper Res 8:42–45
Lawler E, Lenstra J Rinnooy, Rinnooy Kan A, Shmoys D (1985) The traveling salesman problem. Wiley, Chichester
Núñez M, Rafels C (2005) The Böhm-Bawerk horse market: a cooperative analysis. Int J Game Theory 33:421–430
Psaraftis H (1980) A dynamic programming approach for sequencing groups of identical jobs. Oper Res 28:1347–1359
Schmeidler D (1969) The nucleolus of a characteristic function game. SIAM J Appl Math 17:1163–1170
Shapley L, Shubik M (1972) The assignment game I: the core. Int J Game Theory 1:111–130
Slikker M (2005) Balancedness of sequencing games with multiple parallel machines. Ann Oper Res 137:177–189
Smith W (1956) Various optimizers for single-stage production. Naval Res Logist Q 3:59–66
van der Veen J, Woeginger G, Zhang S (1998) Sequencing jobs that require common resources on a single machine: a solvable case of the TSP. Math Progr 82:235–254
Verdaasdonk, L. (2007). Telephone sequencing. MSc thesis, Department of Econometrics and OR, Tilburg University, Tilburg
von Böhm-Bawerk E (1891) Positive theory of capital (translated by W. Smart). Macmillan, London