Cooperative games and network structures

Musegaas, Marieke

Publication date:
2017

Document Version
Publisher's PDF, also known as Version of record

Cooperative Games and Network Structures

Cooperative Games and Network Structures

Proefschrift

ter verkrijging van de graad van doctor aan Tilburg University, op gezag van de rector magnificus, prof. dr. E.H.L. Aarts, in het openbaar te verdedigen ten overstaan van een door het college voor promoties aangewezen commissie in de aula van de Universiteit op vrijdag 19 mei 2017 om 14.00 uur door

Marieke Musegaas

geboren op 4 januari 1991 te Utrecht.

PROMOTIECOMMISSIE:

PROMOTOR:	prof. dr. P.E.M. Borm
COPROMOTOR:	dr. M. Quant
OVERIGE LEDEN:	dr. P. Calleja
	prof. dr. H.J.M. Hamers
	prof. dr. G. Schäfer
	dr. M. Slikker

Acknowledgments

The first of many thanks go to my two supervisors: Peter Borm and Marieke Quant. I would like to thank them for all their help, motivation and enthusiasm during my PhD years. I benefited from your expertise and experience in both research and teaching. It was great how the two of you were able to motivate me whenever my enthusiasm was (partly) gone.

Besides, I would like to thank the other members of my committee. Guido Schäfer, Herbert Hamers, Marco Slikker and Pedro Calleja, thank you for taking the time to read this thesis and giving valuable comments. I must admit that, in the end, I even enjoyed discussing my research with you during the predefense. It was good to see that all of you read the thesis in great detail.

I very much enjoyed the research visit with Arantza Estévez-Fernández to Gloria Fiestras-Janeiro in Vigo. Research was never so frustrating, but at the same time very challenging and interesting. Many thanks go to Gloria for hosting me at Universida de Vigo and for showing me around in this beautiful part of Spain.

Also, I would like to thank my co-author Bas Dietzenbacher for his critical comments and for the nice time we shared during conferences.

Many thanks go to the people from the PhD group: Ahmadreza Marandi, Aida Abiad, Alaa Abi Morshed, Bas van Heiningen, Emanuel Marcu, Hettie Boonman, Jan Kabátek, Maria Lavrutich, Nick Huberts (with whom I am friends already for many years), Shobeir Amirnequiee, Sybren Huijink, Trevor Zhen, Victoria van Eikenhorst and Zeinab Bakhtiari. Among this group is also my academic twin sister Marleen Balvert. Even during the last part of our PhD years many people still mixed us up. Thank you for all the nice tea breaks we had together and for all the support you gave me with respect to teaching and research. Another very special person among this group is Frans de Ruiter. Thank you for all the good memories we share together.

During these years I was happy to share an office with Olga Kuryatnikova for some time. I am a lucky person to have had such an amazing officemate. It was great to share with you the good and bad times caused by research, teaching and many other things in life. I also shared an office with Krzysztof Postek. Thank you for being my officemate for two years. But, more importantly, thank you for being a very good friend. Your unconditional support and encouragement still helps me a lot and is greatly appreciated.

I want to acknowledge all the people I have met being a member of T.S.A.V. Parcival. Thank you for being my running and training mates, Alja Poklar, Anke te Boekhorst, Anne Balter, Anne de Smit, Johanna Grames, Lisette Rentmeester, Mirthe Boomsma, Pieter Vromen, Roos Staps, and many others. And of course, also Armand van der Smissen, thank you for being the best running trainer. I also want to acknowledge the people from T.S.S.V. Braga. Thank you for the nice trainings on the ice skating rink. You made me fall in love again with speed skating.

Special thanks go to the people I recently met at Maranatha. I am grateful that I have met you during the very last part of my PhD years and that you showed me the great value of life. Especially, Michiel Peeters for making me feel welcome during the Maranatha activities and Kristína Melicherová for our almost weekly lunches.

Many people I already met in Tilburg before my PhD years. They also contributed in a very important way to my thesis by being a good friend to me. Thank you, Bas Verheul, Dieuwertje Verdouw, Emma Bunte, Iris Zwaans, Lindsay Overkamp, Maartje de Ronde, Nick Huberts (again!), Robbert-Jan Tijhuis and Tess Beukers. I really appreciate all the memorable moments we had together during holiday trips, escape rooms, game evenings and many more moments.

I would like to express my gratitude to my Daltonerf housemate, Annick van Ool. It was really good to share this nice place with you. Thank you for all the dinners we had together and also for being a good friend.

Also thanks to my two friends Evelien Barendregt and Stefan Inthorn whom I met at secondary school. They can finally see what I have been working on during all those years in Tilburg.

Finally and most importantly, I am thankful to my family, who always supported me during these PhD years. I am grateful to my mom, my dad and my brother for being always so patient and for their unconditional love. The most special thanks go to my granddad.

Marieke Musegaas
Rotterdam, March 2017

Contents

Chapter 1

Introduction

1.1 Cooperation and networks

Cooperative game theory is a mathematical tool to analyze the cooperative behavior within a group of players. In negotiating about meaningful and stable full cooperation between all members of this group, an important issue that has to be settled upon is how to allocate the joint revenues from cooperation back to the individual members in an adequate way. The most common model to answer the question of fair allocation of joint revenues is to consider a transferable utility (TU) game as first introduced by Von Neumann and Morgenstern (1944). A TU-game specifies the monetary value of each possible subgroup of the whole group of players, a so-called coalition. The monetary value of a coalition in principle represents the joint revenues this coalition can obtain by means of cooperation without any help of players outside the coalition. The exact coalitional monetary values typically are an important context-specific modeling choice since they will serve as benchmarks to address the question of how to allocate the known joint revenues of the group as a whole. In the game theoretic literature several general solution concepts have been introduced and analyzed. For example, the Shapley value (Shapley (1953)) assigns to each game an efficient weighted average of all possible marginal vectors. Core allocations (Gillies (1959)) are such that the players in every possible coalition according to this allocation jointly receive at least as much as the joint revenues they could obtain by acting as a separate group of cooperating players. Other game theoretic solution concepts are, among others, the τ-value (Tijs (1981)) and the nucleolus (Schmeidler (1969)).

Interactive combinatorial optimization problems on networks typically lead to allocation problems within a cooperative framework. In combinatorial optimization there is usually one (global) decision maker who has to find an optimal solution with respect to a given (global) objective function from a finite (but typically huge) set of feasible solutions. However, if the (global) objective function is derived from (local) objective functions of different agents involved in the underlying network system, then additionally a cooperative allocation problem has to be addressed. An adequately defined associated TU-game can help to analyze such an allocation problem. As examples of the interrelation between cooperative game theory and network structures we mention two well-studied classes from the literature: minimum cost spanning tree games (cf. Bird (1976), Suijs (2003), Norde, Moretti, and Tijs (2004)) and traveling salesman games (cf. Potters, Curiel, and Tijs (1992), Derks and Kuipers (1997), Kuipers, Solymosi, and Aarts (2000)).

Rankings in brain networks

Consider a brain network of neuronal structures and its corresponding connections. In order to understand the consequences of a possible lesion of one of these neuronal structures, we want to compute the influence of each neuronal structure to the connectivity of the network as a whole. As a brain network can be represented by a directed graph, graph theoretical concepts can be applied for the analysis. For example, Kötter and Stephan (2003) proposed a set of network participation indices, which are derived from simple graph theoretic measures, that characterize how a neuronal structure participates in the whole brain network. In fact, also game theoretical concepts can be used for this analysis by, for example, applying the Shapley value of an appropriately chosen TU-game associated to this brain network. The following example considers a fictive brain network as a didactic illustration.

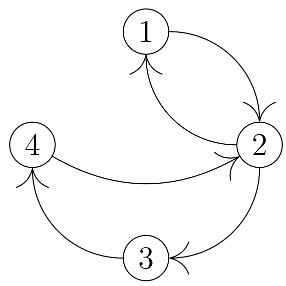


Figure 1.1: A didactic example of a brain network

Example 1.1.1. Consider the brain network as depicted in the directed graph in Figure 1.1. The vertices in this directed graph represent the set of neuronal structures and the arcs represent the connections between the neuronal structures. An arc means that a signal can be sent from one neuronal structure to another.

One can associate a brain network game, denoted by v, to this brain network as follows: the players in this game are the neuronal structures and the value of a coalition is determined by the number of ordered pairs (i, j) within the subgroup for which there exists a directed path from i to j in the induced subgraph. For example, for coalition $\{1, 2, 4\}$ we have four of such pairs, namely $(1, 2)$, $(2, 1)$, $(4, 1)$ and $(4, 2)$. Hence, $v(\{1, 2, 4\}) = 4$. Since for every neuronal structure there exists a directed path to every other neuronal structure, the value of the grand coalition equals 12.

The Shapley value of the corresponding brain network game (cf. Table 1.1) can be interpreted as a measure for the influence of a neuronal structure. According to the Shapley value neuronal structure 2 should be considered as most influential and 1 as the least influential. △

Neuronal structure	Shapley value
1	$2\frac{1}{6}$
2	$4\frac{1}{6}$
3	$2\frac{5}{6}$
4	$2\frac{5}{6}$

Table 1.1: Shapley values of the fictive brain network in Figure 1.1

In Chapter 3 of this thesis we consider brain network games in more detail. In that chapter we argue that the above described brain network game is an improvement upon a previously defined game corresponding to a brain network.

Three-valued simple games

The class of simple games, a subclass of TU-games, has widely been applied for decision rules in legislatures. In simple games there are two possible values for each coalition: 0 and 1 (‘losing’ or ‘winning’). A three-valued simple game goes one step further as there are three possible values: 0, 1 and 2. In the following example we model the legislative procedure of the current EU-28 Council as a three-valued simple game and we apply the Shapley value to measure the relative influence of each country.

Example 1.1.2. The EU Council represents the national governments of the member states of the European Union, so it sits in national delegations rather than political groups. The legislative procedure of the EU Council is as follows. Usually, the EU Council acts on the basis of a proposal from the European Commission. When, on the other hand, the EU Council does not act on the basis of a proposal from the European Commission, there are more restrictions for legislation to get accepted by the EU Council. Thus within the EU Council two different voting systems are in place simultaneously.

In this example we will illustrate the situation of the current EU-28 Council. As from 1 November 2014 the EU Council uses a voting system of double majority (member states and population) to pass new legislation, which works as follows. If the EU Council acts on the basis of a proposal from the European Commission, then it requires the support of at least 55% of the member states representing at least 65% of the EU population. When the EU Council does not act on the basis of a proposal from the European Commission, then it requires 72% of the member states representing at least 65% of the EU population. In case of EU-28, the 55% and 72% majorities of the member states correspond to at least 16 and 21 member states, respectively. Table 1.2 gives the population share of each EU-28 member state compared to the total population of EU-28 for the period 1 November 2014 to 31 December 2014.

Johnston and Hunt (1977) were the first to perform a detailed analysis to the distribution of power in the EU Council. For this, both Johnston and Hunt (1977) and Widgrén (1994) assumed that the EU Council only acts on the basis of a proposal from the European Commission, which is not always the case. Bilbao, Fernández, Jiménez, and López (2000) and Bilbao, Fernández, Jiménez, and López (2002) also considered the case when the legislation is not proposed by the European Commission. By the introduction of three-valued simple games, it is possible to model the legislative procedure of the EU Council as a single TU-game that takes into account both cases.

We define the value $v(S)$ of a coalition S of member states by:

- 2, if the coalition forms a double majority in case the EU Council does not act on the basis of a proposal from the European Commission,

- 1, if the coalition forms a double majority only in case the EU Council acts on the basis of a proposal from the European Commission,

Member state	Population share	Member state	Population share
Germany	0.1593	Austria	0.0167
France	0.1298	Bulgaria	0.0144
United Kingdom	0.1261	Denmark	0.0111
Italy	0.1181	Finland	0.0107
Spain	0.0924	Slovakia	0.0107
Poland	0.0762	Ireland	0.0091
Romania	0.0397	Croatia	0.0084
Netherlands	0.0332	Lithuania	0.0059
Belgium	0.0221	Slovenia	0.0041
Greece	0.0219	Latvia	0.0040
Czech Republic	0.0208	Estonia	0.0026
Portugal	0.0207	Cyprus	0.0017
Hungary	0.0196	Luxembourg	0.0011
Sweden	0.0189	Malta	0.0008

Table 1.2: Population shares of the EU-28, source: `http://www.consilium.europa.eu/uedocs/cms_data/docs/pressdata/EN/genaff/144960.pdf`

- 0, otherwise.

Hence, with w_i denoting the population share of member state i compared to the total EU-28 population (see Table 1.2), we have

$$v(S) = \begin{cases} 2 & \text{if } \sum_{i \in S} w_i \geq 0.65 \text{ and } |S| \geq 21, \\ 1 & \text{if } \sum_{i \in S} w_i \geq 0.65 \text{ and } 16 \leq |S| < 21, \\ 0 & \text{if } \sum_{i \in S} w_i < 0.65 \text{ or } |S| < 16, \end{cases}$$

for every coalition S. In Table 1.3 we listed the Shapley value, which can be interpreted as a measure for the relative influence of each country. △

In Chapter 4 of this thesis we consider three-valued simple games in more detail. In this chapter we also use three-valued simple games to model a parliamentary bicameral system.

Coloring games

Consider the situation where a number of agents all need to have access to some type of facility, but some agents might be in conflict. All facilities are similar, but if two

Member state	Shapley value	Member state	Shapley value
Germany	0.2215	Austria	0.0510
France	0.1694	Bulgaria	0.0490
United Kingdom	0.1643	Denmark	0.0461
Italy	0.1542	Finland	0.0458
Spain	0.1241	Slovakia	0.0457
Poland	0.1101	Ireland	0.0443
Romania	0.0731	Croatia	0.0438
Netherlands	0.0666	Lithuania	0.0416
Belgium	0.0559	Slovenia	0.0401
Greece	0.0557	Latvia	0.0400
Czech Republic	0.0547	Estonia	0.0389
Portugal	0.0547	Cyprus	0.0381
Hungary	0.0536	Luxembourg	0.0375
Sweden	0.0530	Malta	0.0374

Table 1.3: Shapley values of the three-valued simple game v representing the EU-28 Council legislative procedure

agents are in conflict, they cannot have access to the same facility. The problem is to find the minimum number of facilities that can serve all agents such that every agent has non-conflicting access to some facility. An example is the channel assignment in cellular telephone networks (cf. McDiarmid and Reed (2000)) where frequency bands (facilities) must be assigned to transmitters (agents) while avoiding interference (being in conflict). Hence, if unacceptable interference might occur between two transmitters, they should be assigned different frequency bands. The problem is to find the minimum number of frequency bands needed.

This combinatorial optimization problem can be formulated in a more general setting and is known as a minimum coloring problem of an undirected graph in which the vertices represent the agents. Two vertices are connected if and only if the corresponding agents are in conflict. When two vertices receive the same color it means that the corresponding agents have access to the same facility. The minimum number of colors needed to color all vertices in the graph is equal to the minimum number of facilities needed in order to give every agent access to some facility. The minimum coloring problem is known to be NP-hard and thus it is suspected that there does not exist a polynomial time algorithm for solving this problem.

The next question that arises is how to allocate the total costs among the agents, where we assume that the total costs are linearly increasing with the number of facilities used. If there is an agent that is in conflict with many other agents, then he causes a substantial part of the total coloring costs. Therefore, it might be fair that this agent also pays a substantial part of the total coloring costs. This issue of allocating the total coloring costs among all players can be analyzed by the following cost savings TU-game: assuming that initially every agent has its individual facility, the value of a coalition is determined by the number of colors that are saved due to cooperation. The following numerical example shows the computation of the corresponding TU-game.

Example 1.1.3. Consider the Petersen graph in Figure 1.2 where the vertices represent the agents. A minimum of three facilities is needed in order to give every agent access to some facility. Namely, agent 1, 7 and 10 are assigned to facility A, agent 2, 3 and 9 to facility B, and agent 4, 5, 6 and 8 to facility C. Note that this is an admissible coloring of the Petersen graph since no two adjacent vertices share the same color. Moreover, it is readily seen that it is not possible to color the Petersen graph with less than three colors.

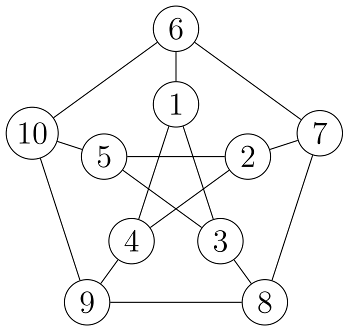


Figure 1.2: The Petersen graph

The next question is how to divide the costs of these three facilities in a fair way among all agents. For this, we consider the corresponding TU-game, denoted by v. Clearly, a one-person coalition needs access to one facility and no costs can be saved. Therefore, all one-person coalitions have value equal to 0. Since player 1 and 2 are not in conflict, they can share one facility if they decide to cooperate. Hence, the costs of one facility can can be saved, i.e., $v(\{1,2\}) = 1$. However, if player 1 and 3 decide to cooperate, they cannot save costs because they are in conflict and thus

$v(\{1,3\}) = 0$. Moreover, if player 4, 5 and 6 decide to cooperate, they save the costs of two facilities and thus $v(\{4,5,6\}) = 2$. △

In Chapter 5 of this thesis we consider in particular minimum coloring problems that lead to three-valued simple cost savings games.

Sequencing games

In a one-machine sequencing situation there are a number of jobs that have to be processed on a single machine. The processing time of a job is the time the machine takes to process this job. The objective in a sequencing situation is to find a processing order that minimizes a certain cost criterion. A widely used cost criterion is the total weighted completion time when the individual costs are linearly increasing with the time this job is in the system. As an example, consider a situation in which there is one office and a number of customers. All customers are waiting in a queue for the handling of a personal request, e.g., trucks that are waiting at a custom house for permission to cross the border. The handling time and the costs per time unit because of waiting can be different for each customer. The problem is to find an order that minimizes the total waiting costs of all customers together.

In order to find an optimal processing order, customers with high missed revenues per time unit should be handled as early as possible. On the other hand, customers with high handling time should be handled as late as possible, so that the waiting time for the other customers is as low as possible. The urgency to handle a customer is determined by the balance of these two concepts. More precisely, denote the processing time of job i by p_i. Moreover, let the costs of job i of spending t time units in the system be given by the linear cost function $c_i(t) = \alpha_i t$ with $\alpha_i > 0$. Then, a processing order that minimizes the total costs is an order where the jobs are processed in non-increasing order with respect to their urgency u_i defined by $u_i = \frac{\alpha_i}{p_i}$ (cf. Smith (1956)).

Sequencing games (as introduced by Curiel, Pederzoli, and Tijs (1989)) arise from one-machine sequencing situations by assigning a player to each job. By assuming the presence of an initial processing order, rearranging the jobs, in order to go from the initial processing order to an optimal processing order, will lead to cost savings. The question is how to allocate the total cost savings among the players in a fair way. To analyze this problem, we can study the following TU-game in which the value of a coalition is determined by the maximal cost savings this coalition can make by means of admissible rearrangements, which (classically) ensures that the players outside the coalition keep the same predecessors. Curiel et al. (1989) showed that such sequencing games have a non-empty core. The following numerical example shows the computation of the corresponding TU-game.

Example 1.1.4. Consider a one-machine sequencing situation with three players denoted by 1, 2 and 3 respectively, each with one job to be processed by the single machine, which is denoted by M. Take as initial order the order (1 2 3) (cf. Figure 1.3).

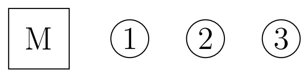


Figure 1.3: Example of a one-machine sequencing situation

The processing times are $p_1 = 3$, $p_2 = 2$ and $p_3 = 1$. Given these numbers, we can draw a Gantt chart of the initial order as in Figure 1.4. The numbers below the jobs of the players represent the processing times. The numbers on the bottom line in bold give the completion times of the players with respect to the initial order, i.e., the time that the corresponding player is in the system. For example, the completion time of player 2 is 5 because he first has to wait 3 time units for the completion of the job of player 1 and then is handled in 2 time units himself. After his job is processed, he can leave the system and thus he does not have to wait for the completion of job 3.

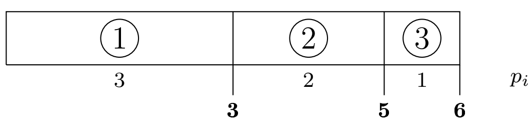


Figure 1.4: Gantt chart of the initial order (1 2 3)

Assuming the coefficients of the linear cost functions to be given by $\alpha_1 = 4$, $\alpha_2 = 6$ and $\alpha_3 = 5$, the Gantt chart implies that the individual costs are $4 \cdot 3 = 12$, $6 \cdot 5 = 30$ and $5 \cdot 6 = 30$ for player 1, 2 and 3, respectively. Consequently, the total joint costs with respect to this initial order are $12 + 30 + 30 = 72$.

Obviously, a one-person coalition cannot save costs by means of admissible rearrangements and thus all one-person coalitions have value equal to 0. However, if player 1 and 2 decide to cooperate, there is another possible order that is allowed:

(2 1 3) (cf. Figure 1.5). Note that this order is allowed because player 3, who is outside the coalition, keeps the same predecessors. The individual costs in the order (2 1 3) are $4 \cdot 5 = 20$ and $6 \cdot 2 = 12$ for player 1 and 2, respectively. As a consequence, the total costs for player 1 and 2 in the order (2 1 3) are 32, whereas the total costs for player 1 and 2 in the initial order were 42. Hence, the cost savings by means of admissible rearrangements for coalition $\{1,2\}$ equal 10. Using similar arguments coalition $\{2,3\}$ can save 4. For coalition $\{1,3\}$ only the order (1 2 3) is allowed because player 2, who is outside the coalition, needs to keep the same predecessors. Hence, player 1 and 3 are not allowed to swap position and the coalition $\{1,3\}$ cannot make cost savings.

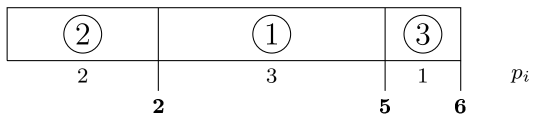


Figure 1.5: Gantt chart of the order (2 1 3)

For the grand coalition $\{1,2,3\}$ all possible orders are allowed. Using Smith (1956) we know that any optimal order is such that the players are in non-increasing order with respect to their urgency, where the urgencies are defined by $u_1 = \frac{4}{3}$, $u_2 = \frac{6}{2}$ and $u_3 = \frac{5}{1}$. Hence, (3 2 1) is an optimal order with individual costs $4 \cdot 6 = 24$, $6 \cdot 3 = 18$ and $5 \cdot 1 = 5$ for player 1, 2 and 3, respectively. As a consequence, the total costs in the order (3 2 1) are 47, whereas the total costs in the initial order were 72. Hence, the total cost savings are 25. Table 1.4 summarizes the values $v(S)$ of the cost savings of each possible coalition S.

S	$\{1\}$	$\{2\}$	$\{3\}$	$\{1,2\}$	$\{1,3\}$	$\{2,3\}$	$\{1,2,3\}$
$v(S)$	0	0	0	10	0	4	25

Table 1.4: The TU-cost savings game corresponding to the one-machine sequencing situation in Example 1.1.4

In a core allocation the value 25 is allocated among the players in such a way that each coalition receives at least its coalitional value. For example, the allocation $(1,23,1)$, 1 to player 1, 23 to player 2 and 1 to player 3 belongs to the core. This is because player 1 and 2 are allocated 24, which is more than the cost savings of 10 they can obtain on their own, and player 2 and 3 are allocated 24, which is also more than the cost savings of 4 they can obtain on their own, △

In Chapter 6 of this thesis we analyze in more detail a new variant of sequencing games, so-called Step out - Step in (SoSi) sequencing games, where the set of admissible orders for a coalition is modified. Now, any player is also allowed to step out from his position in the processing order and to step in at any position later in the processing order.

Example 1.1.5. Reconsider the one-machine sequencing situation of Example 1.1.4. Note that the cost savings of the coalitions in the corresponding SoSi sequencing game will be equal to the cost savings of the coalitions in the classical sequencing game as in Table 1.4 except for the coalition $\{1,3\}$. In particular, in the SoSi sequencing game the orders (2 1 3) and (2 3 1) are also allowed for coalition $\{1,3\}$. Although processing order (2 1 3) is allowed, this order will never be better for coalition $\{1,3\}$ than the initial order (1 2 3) because the costs for player 1 become higher and the costs for player 3 stay the same. As for processing order (2 3 1), note that player 2, who is outside the coalition, does not get worse off by this swap and even benefits from it because his completion time decreases. The individual costs in the order (2 3 1) are $4 \cdot 6 = 24$ and $5 \cdot 3 = 15$ for player 1 and 3, respectively. As a consequence, the total costs for player 1 and 3 in the order (2 3 1) are 39, whereas the total costs for player 1 and 3 in the initial order were 42. Hence, the highest possible cost savings for coalition $\{1,3\}$ are equal to 3. Table 1.5 summarizes the values $v(S)$ of the cost savings of each possible coalition S.

S	$\{1\}$	$\{2\}$	$\{3\}$	$\{1,2\}$	$\{1,3\}$	$\{2,3\}$	$\{1,2,3\}$
$v(S)$	0	0	0	10	3	4	25

Table 1.5: The SoSi sequencing game corresponding to the one-machine sequencing situation in Example 1.1.4

Note that the previous allocation $(1,23,1)$ does not belong to the core anymore since player 1 and 3 are allocated 2 jointly, which is strictly less than the cost savings of 3 they can obtain on their own. △

Since all coalitional values of the cost savings become larger in SoSi sequencing games while the value of the grand coalition stays the same, SoSi sequencing games might not have a non-empty core anymore. In Chapter 6 we study the core of SoSi sequencing games and, among other things, we provide a polynomial time algorithm determining an optimal processing order for a coalition in a SoSi sequencing game.

1.2 Overview

In Chapter 2 we introduce some general basic notions, concepts and definitions regarding cooperative games and network structures.

Chapter 3 considers the problem of computing the influence of a neuronal structure in a brain network. Abraham, Kötter, Krumnack, and Wanke (2006) computed this influence by using the Shapley value of a coalitional game corresponding to a directed network as a rating. Kötter, Reid, Krumnack, Wanke, and Sporns (2007) applied this rating to large-scale brain networks, in particular to the macaque visual cortex and the macaque prefrontal cortex. The aim of this chapter is to improve upon the above technique by measuring the importance of subgroups of neuronal structures in a different way. This new modeling technique not only leads to a more intuitive coalitional game, but also allows for specifying the relative influence of neuronal structures and a direct extension to a setting with missing information on the existence of certain connections.

In Chapter 4 we study a specific class of TU-games, called three-valued simple games. These games can be considered as a natural extension of simple games. We analyze to which extent well-known results on the core and the Shapley value for simple games can be extended to this new setting. To describe the core of a three-valued simple game we introduce (primary and secondary) vital players, in analogy to veto players for simple games. The vital core, which fully depends on (primary and secondary) vital players, is shown to be a subset of the core. Moreover, it is seen that the transfer property of Dubey (1975) can still be used to characterize the Shapley value for three-valued simple games. We illustrate three-valued simple games and the corresponding Shapley value in a parliamentary bicameral system.

Chapter 5 continues with simple and three-valued simple games. Namely, we characterize the class of conflict graphs inducing simple or three-valued simple minimum coloring games. We provide an upper bound on the number of maximum cliques of conflict graphs inducing such games. Moreover, a characterization of the core is provided in terms of the underlying conflict graph. In particular, in case of a perfect conflict graph the core of the corresponding three-valued simple minimum coloring game equals the vital core. Finally, we study for simple minimum coloring games the decomposition into unanimity games and derive an elegant expression for the Shapley value.

Chapter 6 introduces a new class of relaxed sequencing games: the class of Step out - Step in (SoSi) sequencing games. In this relaxation any player within a coalition is allowed to step out from his position in the processing order and to step in at any position later in the processing order. First, we show non-emptiness of the core of SoSi sequencing games by means of a more generally applicable result. Moreover, we provide a polynomial time algorithm to determine the value and an optimal processing order for an arbitrary coalition in a SoSi sequencing game. This algorithm is used to prove that SoSi sequencing games are convex. In particular, we use that in determining an optimal processing order of a coalition $S \cup \{i\}$, the algorithm can start from the optimal processing order found for coalition S and thus all information on this optimal processing order of S can be used.

Chapter 2

Preliminaries

2.1 Cooperative games

With N a non-empty finite set of players, a *transferable utility (TU) game* is a function $v : 2^N \to \mathbb{R}$ which assigns a number to each coalition $S \in 2^N$, where 2^N denotes the collection of all subsets of N. The value $v(S)$ in general represents the highest joint monetary payoff or cost savings the coalition S can jointly generate by means of optimal cooperation without any help of the players in $N \backslash S$. By convention, $v(\emptyset) = 0$. Let TU^N denote the class of all TU-games with player set N.

A game $v \in \mathrm{TU}^N$ is called *monotonic* if $v(S) \leq v(T)$ for all $S, T \in 2^N \backslash \{\emptyset\}$ with $S \subset T$. Hence, in a monotonic game the worth of a coalition increases when the coalition grows. The game v is called *superadditive* if $v(S \cup T) \geq v(S) + v(T)$ for all $S, T \in 2^N \backslash \{\emptyset\}$ with $S \cap T = \emptyset$. Hence, in a superadditive game breaking up a coalition into parts does not pay. It is desirable that TU-games satisfy the two basic properties of monotonicity and superadditivity, since they provide a clear incentive for cooperation in the grand coalition and thus provides a motivation to focus on fairly allocating the worth of the grand coalition. Note that every nonnegative superadditive game is monotonic. The game v is called *convex* (Shapley (1971)) if

$$v(S \cup \{i\}) - v(S) \leq v(T \cup \{i\}) - v(T), \tag{2.1}$$

for all $S, T \in 2^N \backslash \{\emptyset\}$, $i \in N$ such that $S \subset T \subseteq N \backslash \{i\}$, i.e., the incentive for joining a coalition increases as the coalition grows. Using recursive arguments it can be seen that in order to prove convexity it is sufficient to show (2.1) for the case $|T| = |S| + 1$ which boils down to

$$v(S \cup \{i\}) - v(S) \leq v(S \cup \{j\} \cup \{i\}) - v(S \cup \{j\}), \tag{2.2}$$

for all $S \in 2^N \backslash \{\emptyset\}$, $i, j \in N$ and $i \neq j$ such that $S \subseteq N \backslash \{i, j\}$.

A game $v \in \mathrm{TU}^N$ is called *simple* if

(i) $v(S) \in \{0, 1\}$ for all $S \subset N$,

(ii) $v(N) = 1$,

(iii) v is monotonic.

A coalition is *winning* if $v(S) = 1$ and *losing* if $v(S) = 0$. Let SI^N denote the class of all simple games with player set N. For $v \in \mathrm{SI}^N$ the set of *veto players* is defined by

$$\mathrm{veto}(v) = \bigcap \{S \mid v(S) = 1\}.$$

Hence, the veto players are those players who belong to every coalition with value 1.

An example of a simple game is a unanimity game, where for each $T \in 2^N \backslash \{\emptyset\}$, the *unanimity game* $u_T \in \mathrm{TU}^N$ is defined by

$$u_T(S) = \begin{cases} 1 & \text{if } T \subseteq S, \\ 0 & \text{otherwise}, \end{cases}$$

for all $S \in 2^N$. Unanimity games play an important role since every game $v \in \mathrm{TU}^N$ can be written in a unique way as a linear combination of unanimity games, i.e.,

$$v = \sum_{T \in 2^N \backslash \{\emptyset\}} c_T u_T,$$

where $c_T \in \mathbb{R}$ is uniquely determined for all $T \in 2^N \backslash \{\emptyset\}$ by the recursive formula (cf. Harsanyi (1958))

$$c_T = v(T) - \sum_{S : S \subset T, S \neq \emptyset} c_S.$$

The *core* (Gillies (1959)), $C(v)$, of a game $v \in \mathrm{TU}^N$ is defined by

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

Hence, the core consists of all possible stable allocations of $v(N)$ for which no coalition has an incentive to leave the grand coalition. Consequently, if the core is empty, then it is not possible to find a stable allocation of $v(N)$. From Shapley (1971) and Ichiishi (1981) it follows that $v \in \mathrm{TU}^N$ is convex if and only if

$$C(v) = \mathrm{Conv}(\{m^\sigma(v) \mid \sigma \in \Pi(N)\}), \tag{2.3}$$

where $\Pi(N) = \{\sigma : N \to \{1, \ldots, |N|\} \mid \sigma \text{ is bijective}\}$ is the set of all orders on N and the *marginal vector* $m^\sigma(v) \in \mathbb{R}^N$, for $\sigma \in \Pi(N)$, is defined by

$$m_i^\sigma(v) = v(\{j \in N \mid \sigma(j) \leq \sigma(i)\}) - v(\{j \in N \mid \sigma(j) < \sigma(i)\}),$$

for all $i \in N$. If $v \in \mathrm{SI}^N$, then the core is given by

$$C(v) = \mathrm{Conv}(\{e^{\{i\}} \mid i \in \mathrm{veto}(v)\}),$$

where for $S \in 2^N \backslash \{\emptyset\}$, the *characteristic vector* $e^S \in \mathbb{R}^N$ is defined as

$$e_i^S = \begin{cases} 1 & \text{if } i \in S, \\ 0 & \text{otherwise,} \end{cases}$$

for all $i \in N$.

A *one-point solution* f on the class G^N with $\mathrm{G}^N \subseteq \mathrm{TU}^N$ is a function $f : \mathrm{G}^N \to \mathbb{R}^N$. So, f assigns to each game $v \in G^N$ a unique vector $f \in \mathbb{R}^N$. A one-point solution f on G^N satisfies

- *efficiency* if $\sum_{i \in N} f_i(v) = v(N)$ for all $v \in \mathrm{G}^N$.
- *symmetry* if $f_i(v) = f_j(v)$ for all $v \in \mathrm{G}^N$ and every pair $i, j \in N$ of symmetric players in v, where players $i, j \in N$ are *symmetric* in v if $v(S \cup \{i\}) = v(S \cup \{j\})$ for all $S \subseteq N \backslash \{i, j\}$.
- the *dummy property* if $f_i(v) = v(\{i\})$ for all $v \in \mathrm{G}^N$ and every dummy player $i \in N$ in v, where player $i \in N$ is a *dummy player* in v if $v(S \cup \{i\}) = v(S) + v(\{i\})$ for all $S \subseteq N \backslash \{i\}$.
- *additivity* if $f(v + w) = f(v) + f(w)$ for all $v, w \in \mathrm{G}^N$ with $v + w \in \mathrm{G}^N$.

The *Shapley value* (Shapley (1953)) is a one-point solution on TU^N defined by

$$\Phi_i(v) = \frac{1}{|N|!} \sum_{\sigma \in \Pi(N)} m^\sigma(v), \tag{2.4}$$

for all $v \in \mathrm{TU}^N$. Alternatively, with $v = \sum_{T \in 2^N \setminus \{\emptyset\}} c_T u_T$, the Shapley value is given by

$$\Phi_i(v) = \sum_{T \in 2^N : i \in T} \frac{c_T}{|T|}, \tag{2.5}$$

for all $i \in N$. Shapley (1953) characterized the Shapley value as the unique one-point solution on the class of TU-games satisfying efficiency, symmetry, the dummy property and additivity.

A solution f on $G^N \subseteq \mathrm{TU}^N$ such that for all $v, w \in \mathrm{G}^N$ also $\max\{v, w\} \in \mathrm{G}^N$ and $\min\{v, w\} \in \mathrm{G}^N$, satisfies the *transfer property* if for all $v, w \in \mathrm{G}^N$ we have

$$f(\max\{v, w\}) + f(\min\{v, w\}) = f(v) + f(w),$$

where $\max\{v, w\}$ and $\min\{v, w\}$ are defined by $(\max\{v, w\})(S) = \max\{v(S), w(S)\}$ and $(\min\{v, w\})(S) = \min\{v(S), w(S)\}$, for all $S \subseteq N$. Dubey (1975) showed that the combination of the axioms of efficiency, symmetry, the dummy property and the transfer property fully determines the Shapley value on the class SI^N of simple games.

2.2 Network structures

An *undirected graph* is represented by a pair $G = (N, E)$, where N is a set of vertices and $E \subseteq \{\{i, j\} \mid i, j \in N, i \neq j\}$ is a set of edges. The graph G is *complete* if $E = \{\{i, j\} \mid i, j \in N, i \neq j\}$, that is if every two vertices are adjacent. K_N denotes the complete graph on the set N of vertices. A *cycle graph* C_n is a graph $G = (N, E)$ for which there exists a bijection $f : \{1, \ldots, n\} \to N$ such that

$$E = \{\{f(i), f(i+1)\} \mid i \in \{1, \ldots, n-1\}\} \cup \{\{f(1), f(n)\}\}.$$

An *odd cycle graph* is a cycle graph C_n where n is odd.

For $S \subseteq N$, the *induced subgraph* of G by S is the graph $G[S] = (S, E[S])$ where $E[S] = \{\{i, j\} \in E \mid i, j \in S\}$. The *complement* of G is the graph $\overline{G} = (N, \overline{E})$ where $\overline{E} = \{\{i, j\} \mid i, j \in N, i \neq j, \{i, j\} \notin E\}$. In this thesis we only consider undirected graphs that are connected on N, i.e., every pair of vertices is linked via a sequence of consecutive edges in E. However, note that it still might happen that some induced subgraph $G[S]$ is not connected on S via $E[S]$.

A *clique* in G is a set $S \subseteq N$ such that $G[S] = K_S$. A *maximum clique* of G is a clique S of the largest possible size, i.e., a clique for which $|S|$ is maximal. The number of vertices in a maximum clique is called the *clique number* of G and is denoted by $\omega(G)$. We denote the set of all maximum cliques in G by $\Omega(G)$.

A *directed graph* is represented by a pair $G = (N, A)$, where N is a set of vertices and $A \subseteq \{(i, j) \mid i, j \in N, i \neq j\}$ is a set of arcs. The *transitive closure* of A, denoted by A^{tr}, is the set of all ordered pairs (s, t) of vertices in N for which there exists a sequence of vertices $v_0 = s, v_1, v_2, \ldots, v_k = t$, such that $(v_{i-1}, v_i) \in A$ for all $i \in \{1, \ldots, k\}$. The graph G is called *strongly connected* if $A^{\text{tr}} = \{(i, j) \mid i, j \in N, i \neq j\}$, that is if for every two vertices i and j in N there is a directed path from i to j and from j to i in (N, A) as described above. For $S \subseteq N$, the *induced subgraph* of G by S is the graph $G[S] = (S, A[S])$ where $A[S] = \{(i, j) \in A \mid i, j \in S\}$. The induced subgraph $G[S]$ is called a *strongly connected component* of G if $G[S]$ is strongly connected and there does not exist a $T \subseteq N$ with $S \subset T$ such that $G[T]$ is strongly connected. We denote the number of strongly connected components in graph (N, A) by $\text{SCC}(N, A)$.

Chapter 3

Shapley ratings in brain networks

3.1 Introduction

In this chapter, based on Musegaas, Dietzenbacher, and Borm (2016), we consider the problem of computing the influence of a neuronal structure in a brain network. Previously described measures for the influence of neuronal structures are for example the three Network Participation Indices (NPIs) as introduced by Kötter and Stephan (2003). These NPIs, which are derived from simple graph theoretic measures, are density (degree of interconnectedness), transmission (the ratio of outdegree to indegree) and symmetry (reciprocal connectivity between a node and its neighbors). Some other nodal connectivity measures are the clustering coefficient (which measures the connectednes of a node's neighbors, Watts and Strogatz (1998)), betweenness centrality (which measures how central a node is within the network, Freeman (1977)) and dynamical importance (based on the maximum eigenvalue of the connectivity matrix, Restrepo, Ott, and Hunt (2006)).

However, in this chapter we explore the application of cooperative game theory in this field. Cooperative game theory has already been used for devising centrality measures in social networks. For example, Gómez, González-Arangüena, Manuel, Owen, Del Pozo, and Tejada (2003) defined a centrality measure in a social network as the difference between the actor's Shapley value in the graph-restricted game and the original game. The aim of this chapter is to improve upon the techniques underlying the game theoretic methodology proposed by Abraham, Kötter, Krumnack, and Wanke (2006). Note that this chapter has a style which deviates from the other chapters in this thesis as the main focus of this chapter is on an application of cooperative game theory and thus not on the theory itself.

Abraham et al. (2006) considered a coalitional game in which the worth of a coalition of vertices, the neuronal structures, is defined as the number of strongly connected components in its induced subnetwork within the whole brain network. Subsequently, Abraham et al. (2006) computed the influence of a neuronal structure in a brain network by using the Shapley value of this coalitional game as a rating. Kötter, Reid, Krumnack, Wanke, and Sporns (2007) applied this rating to large-scale brain networks, in particular to the macaque visual cortex and the macaque prefrontal cortex based on real-life data of Young (1992) and Walker (1940).

In this chapter we introduce an alternative coalitional game which in our opinion has several advantages. First of all, by satisfying superadditivity the game is more intuitive from a game theoretical point of view. Secondly, using the Shapley value of this game as an alternative rating it allows to directly specifying relative influence of neuronal structures. We apply our alternative rating model to the brain networks considered by Kötter et al. (2007) and, generally speaking, our results corroborate the findings of Kötter et al. (2007). Finally, a third advantage of the alternative approach is related to missing information on possible connections in a brain network. As this feature is a common problem, as argued by Kötter and Stephan (2003), we illustrate how our alternative approach allows for a direct incorporation of probabilistic considerations regarding missing information on the existence of certain connections.

The organization of this chapter is as follows. Section 3.2 formally introduces brain network games. In Section 3.3 we apply the Shapley rating based on the brain network game to two large-scale brain networks. Section 3.4 consists of an appendix with explanations about how to calculate the Shapley value for large-scale brain networks.

3.2 Shapley ratings in brain networks

A brain network is a *directed graph* (N, A) where the set of vertices N represents a set of neuronal structures and the set of arcs A represents the connections between the neuronal structures. In this chapter we only consider strongly connected brain networks. However, note that all analyses in this chapter are also valid in case a brain network is not strongly connected. Abraham et al. (2006) introduced a coalitional game (N, w^A) corresponding to a brain network (N, A) defined by

$$w^A(S) = \mathrm{SCC}(S, A[S]),$$

for all $S \subseteq N$. Hence, the worth of a coalition in w^A is defined by the number of strongly connected components (cf. Section 2.2) in its induced subgraph.

Alternatively, we define the *brain network game* (N, v^A) corresponding to (N, A) by

$$v^A(S) = |A[S]^{\text{tr}}|,$$

for all $S \subseteq N$. Hence, the worth of a coalition S in v^A is defined by the number of ordered pairs (i, j) of vertices in S for which there exists a directed path from i to j in $(S, A[S])$.

The following example shows the difference between the games (N, w^A) and (N, v^A).

Example 3.2.1. Consider the brain network (N, A) with $N = \{1, 2, 3, 4\}$ illustrated in Figure 3.1.[1] Note that (N, A) is strongly connected because for every vertex in the graph there exists a directed path to every other vertex. However, the subgraph induced by $\{1, 2, 3\}$ is not strongly connected and we have

$$A[\{1, 2, 3\}]^{\text{tr}} = \{(1, 2), (1, 3), (2, 1), (2, 3)\},$$

and thus $v^A(\{1, 2, 3\}) = 4$. Note that $\text{SCC}(\{1, 2, 3\}, A[\{1, 2, 3\}]) = 2$ because the subgraph induced by $\{1, 2, 3\}$ consists of two strongly connected components: the subgraphs induced by $\{1, 2\}$ and $\{3\}$. As a consequence, $w^A(\{1, 2, 3\}) = 2$. Table 3.1 presents the worth of every coalition in the games (N, w^A) and (N, v^A). Note that (N, w^A) is not superadditive since

$$w^A(\{2\}) + w^A(\{3, 4\}) = 3 > 1 = w^A(\{2, 3, 4\}).$$

It is readily checked that (N, v^A) is superadditive. △

S	$\{i\}$	$\{1,2\}$	$\{1,3\}$	$\{1,4\}$	$\{2,3\}$	$\{2,4\}$	$\{3,4\}$	$\{1,2,3\}$	$\{1,2,4\}$	$\{1,3,4\}$	$\{2,3,4\}$	N
$w^A(S)$	1	1	2	2	2	2	2	2	2	3	1	1
$v^A(S)$	0	2	0	0	1	1	1	4	4	1	6	12

Table 3.1: The coalitional games (N, w^A) and (N, v^A) corresponding to the brain network in Figure 3.1

In contrast to the coalitional game (N, w^A), we show in the following proposition that the brain network game (N, v^A) does satisfy superadditivity.

[1] This instance of a brain network is also used in Example 1 in Section 3.1 of Moretti (2013).

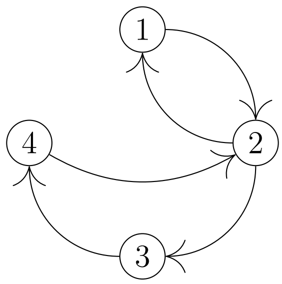


Figure 3.1: The brain network corresponding to Example 3.2.1

Proposition 3.2.1. *Let* (N, A) *be a brain network. Then, the brain network game* (N, v^A) *is superadditive.*

Proof. Let $S, T \in 2^N \backslash \{\emptyset\}$ with $S \cap T = \emptyset$. Since S and T are disjoint, we also have $A[S]^{\text{tr}} \cap A[T]^{\text{tr}} = \emptyset$. Therefore, $|A[S]^{\text{tr}}| + |A[T]^{\text{tr}}| = |A[S]^{\text{tr}} \cup A[T]^{\text{tr}}|$ and thus for proving $v^A(S) + v^A(T) \leq v^A(S \cup T)$ it is sufficient to show that

$$A[S]^{\text{tr}} \cup A[T]^{\text{tr}} \subseteq A[S \cup T]^{\text{tr}}.$$

For showing this, let $(i, j) \in A[S]^{\text{tr}} \cup A[T]^{\text{tr}}$, i.e., there is a directed path from i to j in either $G[S]$ or in $G[T]$. Then, there is also a directed path from i to j in $G[S \cup T]$ and thus $(i, j) \in A[S \cup T]^{\text{tr}}$. □

In the context of coalitional games corresponding to brain networks, the Shapley value can be interpreted as a measure for the influence of a neuronal structure. Abraham et al. (2006) considered the Shapley value $\Phi(w^A)$ as a rating for the neuronal structures in a brain network. Similarly, we consider the Shapley value $\Phi(v^A)$ as a rating.

Example 3.2.2. Reconsider the coalitional games (N, w^A) and (N, v^A) of Example 3.2.1. The Shapley rating $\Phi(w^A)$ is given by[2]

$$\Phi(w^A) = \left(\tfrac{1}{2}, -\tfrac{1}{6}, \tfrac{1}{3}, \tfrac{1}{3}\right),$$

while the Shapley rating $\Phi(v^A)$ is given by

$$\Phi(v^A) = \left(2\tfrac{1}{6}, 4\tfrac{1}{6}, 2\tfrac{5}{6}, 2\tfrac{5}{6}\right),$$

[2]Because of a mistake in the worth of $w^A(\{1, 2, 3\})$, the Shapley value is incorrectly stated by Moretti (2013).

both determining a ranking $(2,3,4,1)$ or $(2,4,3,1)$ (there is a tie for the second highest ranking). We note that a lower Shapley rating in w^A indicates a higher influence in a brain network. On the contrary, a higher Shapley rating in v^A indicates a higher influence.

Since a Shapley rating in w^A can be negative, as is the case in this example, it is not possible to determine the relative influence of two vertices on the basis of $\Phi(w^A)$. On the other hand, a Shapley rating in v^A can not be negative by definition because (N, v^A) is superadditive. Therefore, using $\Phi(v^A)$, we can say that the influence of vertex 2 in the brain network (N, A) is almost twice as large as the influence of vertex 1. △

A common problem in the analysis of brain networks is the fact that it is not known whether some specific connections (arcs) are present or not (cf. Kötter and Stephan (2003)). Using a certain probabilistic knowledge about these unknown connections, this lack of information can readily be incorporated in the brain network game.

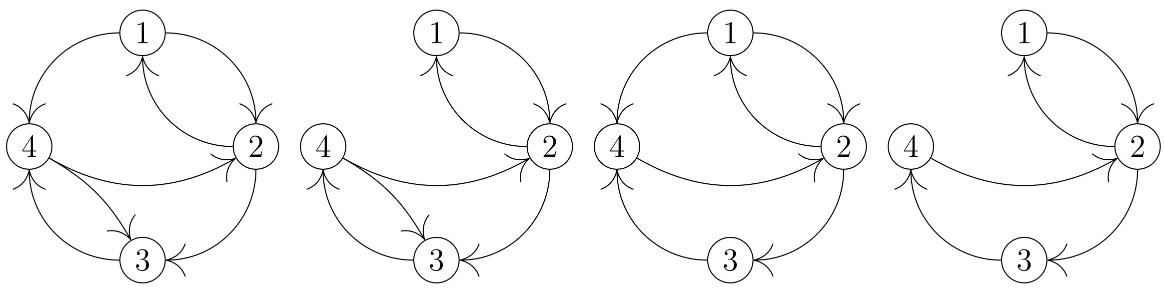
We assume that each possible arc (i, j) is present with probability $p_{ij} \in [0, 1]$. Clearly, for each present arc we set $p_{ij} = 1$ and for each absent arc we set $p_{ij} = 0$. All probabilities are summarized into a vector p. Given such a vector p, we define the *stochastic brain network game* (N, v^p) in which the worth of a coalition equals the expected (in the probabilistic sense) number of ordered pairs for which there exists a directed path in its induced subgraph. Without providing the exact mathematical formulations the following example illustrates how to explicitly determine the coalitional values in a stochastic brain network game.

Example 3.2.3. Reconsider the brain network presented in Example 3.2.1. Only now suppose that the arcs $(1, 4)$ and $(4, 3)$ are present with probability p_{14} and p_{43}, respectively. The complete corresponding vector p can be found in Table 3.2.

(i,j)	$(1,2)$	$(1,3)$	$(1,4)$	$(2,1)$	$(2,3)$	$(2,4)$	$(3,1)$	$(3,2)$	$(3,4)$	$(4,1)$	$(4,2)$	$(4,3)$
p_{ij}	1	0	p_{14}	1	1	0	0	0	1	0	1	p_{43}

Table 3.2: The vector p corresponding to the brain network in Example 3.2.3

In total there are four possible brain networks. These different brain networks are illustrated in Figure 3.2 and the corresponding probabilities for those networks are $p_{14}p_{43}$, $(1 - p_{14})p_{43}$, $p_{14}(1 - p_{43})$ and $(1 - p_{14})(1 - p_{43})$ for the networks in (a), (b), (c), and (d), respectively.[3]



(a) (N, A^1) (b) (N, A^2) (c) (N, A^3) (d) (N, A^4)

Figure 3.2: The possible brain networks corresponding to Example 3.2.3

The expected number of ordered pairs for which there exists a directed path in the induced subgraph of coalition $\{1,3,4\}$ is computed by taking the following weighted average

$$\begin{aligned} v^p(\{1,3,4\}) &= p_{14}p_{43} \cdot v^{A^1}(\{1,3,4\}) + (1-p_{14})p_{43} \cdot v^{A^2}(\{1,3,4\}) \\ &\quad + p_{14}(1-p_{43}) \cdot v^{A^3}(\{1,3,4\}) + (1-p_{14})(1-p_{43}) \cdot v^{A^4}(\{1,3,4\}) \\ &= p_{14}p_{43} \cdot 4 + (1-p_{14})p_{43} \cdot 2 + p_{14}(1-p_{43}) \cdot 2 + (1-p_{14})(1-p_{43}) \cdot 1 \\ &= 1 + p_{14} + p_{43} + p_{14}p_{43}. \end{aligned}$$

Table 3.3 presents the worth of every coalition. The Shapley rating of the game (N, v^p) is given by

$$\Phi_1(v^p) = 2\tfrac{1}{6} + \tfrac{1}{3}p_{14} + \tfrac{1}{12}p_{14}p_{43},$$

$$\Phi_2(v^p) = 4\tfrac{1}{6} - \tfrac{1}{6}p_{14} - \tfrac{1}{3}p_{43} - \tfrac{1}{4}p_{14}p_{43},$$

$$\Phi_3(v^p) = 2\tfrac{5}{6} - \tfrac{1}{2}p_{14} + \tfrac{1}{6}p_{43} + \tfrac{1}{12}p_{14}p_{43},$$

$$\Phi_4(v^p) = 2\tfrac{5}{6} + \tfrac{1}{3}p_{14} + \tfrac{1}{6}p_{43} + \tfrac{1}{12}p_{14}p_{43}.$$

For example, if $p_{14} = \frac{1}{2}$ and $p_{43} = \frac{1}{3}$, then

$$\Phi(v^p) = \left(2\tfrac{25}{72}, 3\tfrac{67}{72}, 2\tfrac{47}{72}, 3\tfrac{5}{72}\right),$$

with corresponding ranking $(2,4,3,1)$. △

[3]So we implicitly assume that the arc probabilities are independent of each other.

S	$\{i\}$	$\{1,2\}$	$\{1,3\}$	$\{1,4\}$	$\{2,3\}$	$\{2,4\}$	$\{3,4\}$	$\{1,2,3\}$	$\{1,2,4\}$	$\{1,3,4\}$	$\{2,3,4\}$	N
$v^p(S)$	0	2	0	p_{14}	1	1	$1+p_{43}$	4	$4+2p_{14}$	$1+p_{14}+p_{43}+p_{14}p_{43}$	6	12

Table 3.3: The stochastic brain network game (N, v^p) corresponding to the brain network in Example 3.2.3

3.3 Results and discussion

In this section we apply the Shapley rating based on the brain network game (N, v^A) to the two large-scale brain networks considered by Kötter et al. (2007) and we compare the results.

The first large-scale brain network is the macaque visual cortex with thirty neuronal structures as depicted in the directed graph in Figure 3.3 (cf. Figure 1 of Kötter et al. (2007) and Young (1992), based on data compiled by Felleman and Van Essen (1991)). Note that if no direction is drawn for an arc, it means that the signal can go both ways. The five brain regions with the highest ranking obtained by means of the Shapley value of the coalitional games (N, w^A) and (N, v^A) can be found below in Table 3.4(a) and (b) respectively.[4] Note that both ratings agree on the top 5; only with respect to the positions 3 and 5 there are some minor differences.

(a) Top 5 of $\Phi(w^A)$

Ranking	Brain region
1.	V4
2.	FEF
3.	46
4.	V2
5.	Vp

(b) Top 5 of $\Phi(v^A)$

Ranking	Brain region
1.	V4
2.	FEF
3.	Vp
4.	V2
5.	46

Table 3.4: Top five rankings of the macaque visual cortex based on the Shapley ratings $\Phi(w^A)$ and $\Phi(v^A)$

The entire Shapley rating $\Phi(v^A)$ of the macaque visual cortex can be found in Figure 3.6. Correspondingly, we can roughly divide the brain regions in five classes based on the relative difference with the brain region with the highest Shapley rating.

[4] Some details about how we computed the Shapley value for this large game can be found in the appendix in Section 3.4.

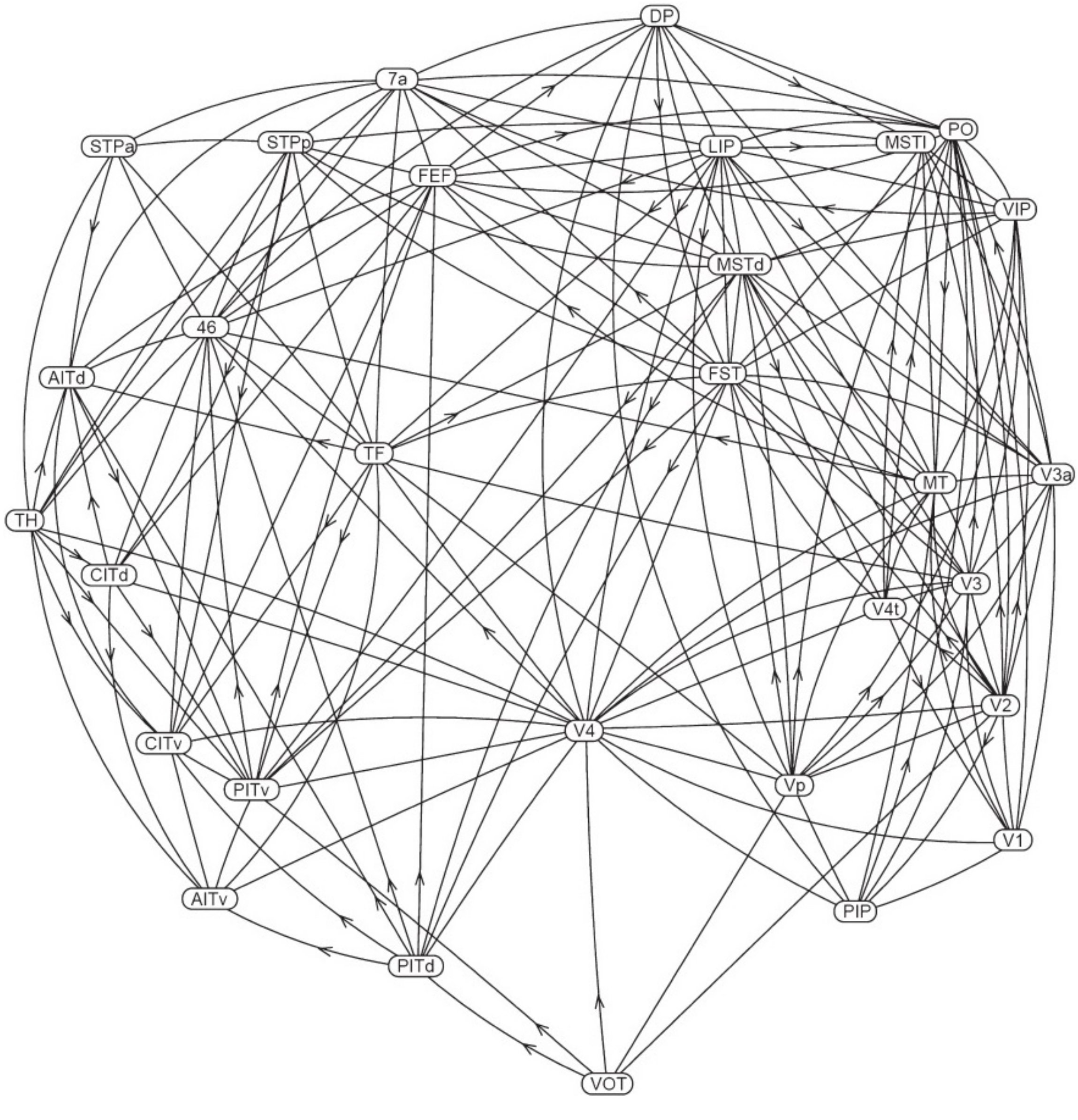


Figure 3.3: The directed graph representing the macaque visual cortex

We consider the following five classes based on the differences in terms of percentage: 0%–5%, 5%–10%, 10%–15%, 15%–20%, 20% and higher (see the corresponding lines in Figure 3.6). The first class consists of the single brain region V4 with the highest Shapley rating. The second class consists of the brain regions FEF to TF as ordered in Figure 3.6 that differ 5%–10% with $V4$. The brain regions in the third class are MSTd to V3, in the fourth class we have MSTl to PITd and in the fifth class we have the single brain region VOT with a relative influence which is 23% lower than that of V4.

The second large-scale brain network is the macaque prefrontal cortex with twelve neuronal structures as illustrated in Figure 3.4 (cf. Figure 3(a) of Kötter et al. (2007) and Walker (1940)). In this case there is a lack of information about the presence or absence of nine connections, which is indicated by the dashed arcs. To get some insight, Kötter et al. (2007) considered two extreme cases. First, they assume that connections with unknown presence are absent. Second, they assume that those connections are present. For both extreme cases the Shapley ratings are calculated separately. Our stochastic brain network game provides a way to incorporate lack of information into one Shapley rating on the basis of probabilistic information. For simplicity, we assume that each connection with unknown presence is absent with probability $\frac{1}{2}$. Note that, in case more information would become available, more adequate probabilities can be readily inserted. Having the complete vector p of arc probabilities, one readily computes the corresponding stochastic brain network game (N, v^p) and the corresponding Shapley rating $\Phi(v^p)$. The ranking based on the Shapley rating $\Phi(v^p)$ can be found in Table 3.5.

Ranking	Brain region
1.	9
2.	24
3.	12
4.	10
5.	46
6.	25
7.	11
8.	8B
9.	13
10.	8A
11.	45
12.	14

Table 3.5: Ranking of the macaque prefrontal cortex based on the Shapley rating $\Phi(v^p)$

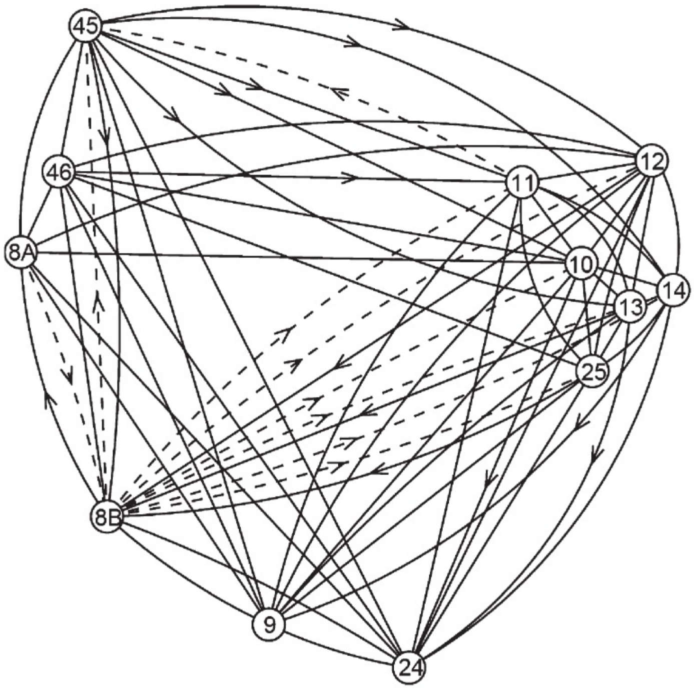


Figure 3.4: The directed graph representing the macaque prefrontal cortex

3.4 Appendix

For computing the Shapley value for a large brain network game[5], like the macaque visual cortex with thirty neuronal structures, we used the following formula for the Shapley value:

$$\Phi_i(v) = \sum_{S \in 2^N \setminus \{\emptyset\}} a_{i,S} v(S), \tag{3.1}$$

where

$$a_{i,S} = \begin{cases} \frac{(|S|-1)!(|N|-|S|)!}{|N|!} & \text{if } i \in S, \\ \frac{-|S|!(|N|-|S|-1)!}{|N|!} & \text{if } i \notin S. \end{cases}$$

Note that this formula of the Shapley value has computational advantages compared to the standard formula in (2.4). Namely, using the formula in (2.4), one considers all $|N|!$ marginal vectors. Moreover, for every marginal vector, one has to compute $|N|$ marginal contributions. Hence, this results in calculating $|N|! \cdot |N|$ marginal

[5]Upon request the Julia code is available.

contributions. While using the formula in (3.1), one considers all $2^{|N|}$ coalitions. Note that we can consider all $2^{|N|}$ coalitions in a structured way, a so-called depth-first search. The depth-first search for all $2^{|N|}$ coalitions with $N = \{1,2,3,4\}$ is illustrated in Figure 3.5. As this figure illustrates, we will visit the coalitions in the following order: $\{\emptyset\}$, $\{1\}$, $\{1,2\}$, $\{1,2,3\}$, $\{1,2,3,4\}$, $\{1,2,4\}$, $\{1,3\}$, $\{1,3,4\}$, $\{1,4\}$, $\{2\}$, $\{2,3\}$, $\{2,3,4\}$, $\{2,4\}$, $\{3\}$, $\{3,4\}$, $\{4\}$. As a consequence, if we store at any moment the two most recent coalitional values, we only need to calculate 15 marginal contributions (the number of arrows in Figure 3.5). To conclude, calculating the Shapley value using (3.1) requires the computation of $2^{|N|}$ marginal contributions, which is considerably less than the $|N|! \cdot |N|$ marginal contributions using (2.4).

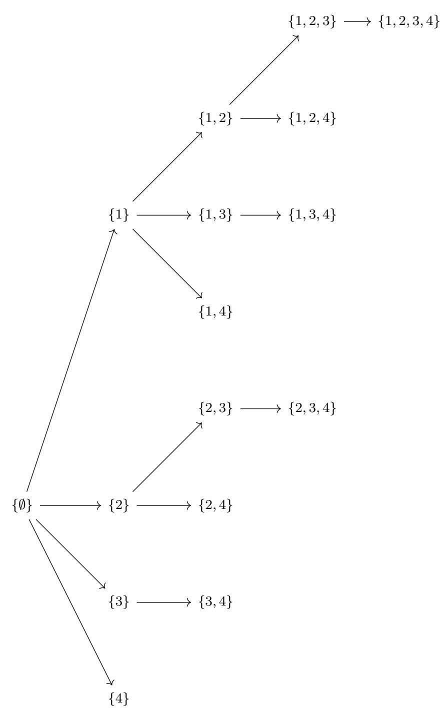


Figure 3.5: The so-called depth-first search for all $2^{|N|}$ coalitions with $N = \{1,2,3,4\}$

In order to calculate the value of a coalition, we use Warshall's algorithm (cf. Warshall (1962)). This algorithm finds the transitive closure of a directed graph by using at most $|N|^3$ steps. Moreover, since we consider the coalitions in a structured way by means of depth-first search, we only need to calculate $2^{|N|}$ marginal contributions. If we store at any moment the two most recent results of Warshall's algorithm, calculating a marginal contribution requires at most $|N|^2$ steps. To conclude, calculating the Shapley value requires at most $2^{|N|} \cdot |N|^2$ steps.

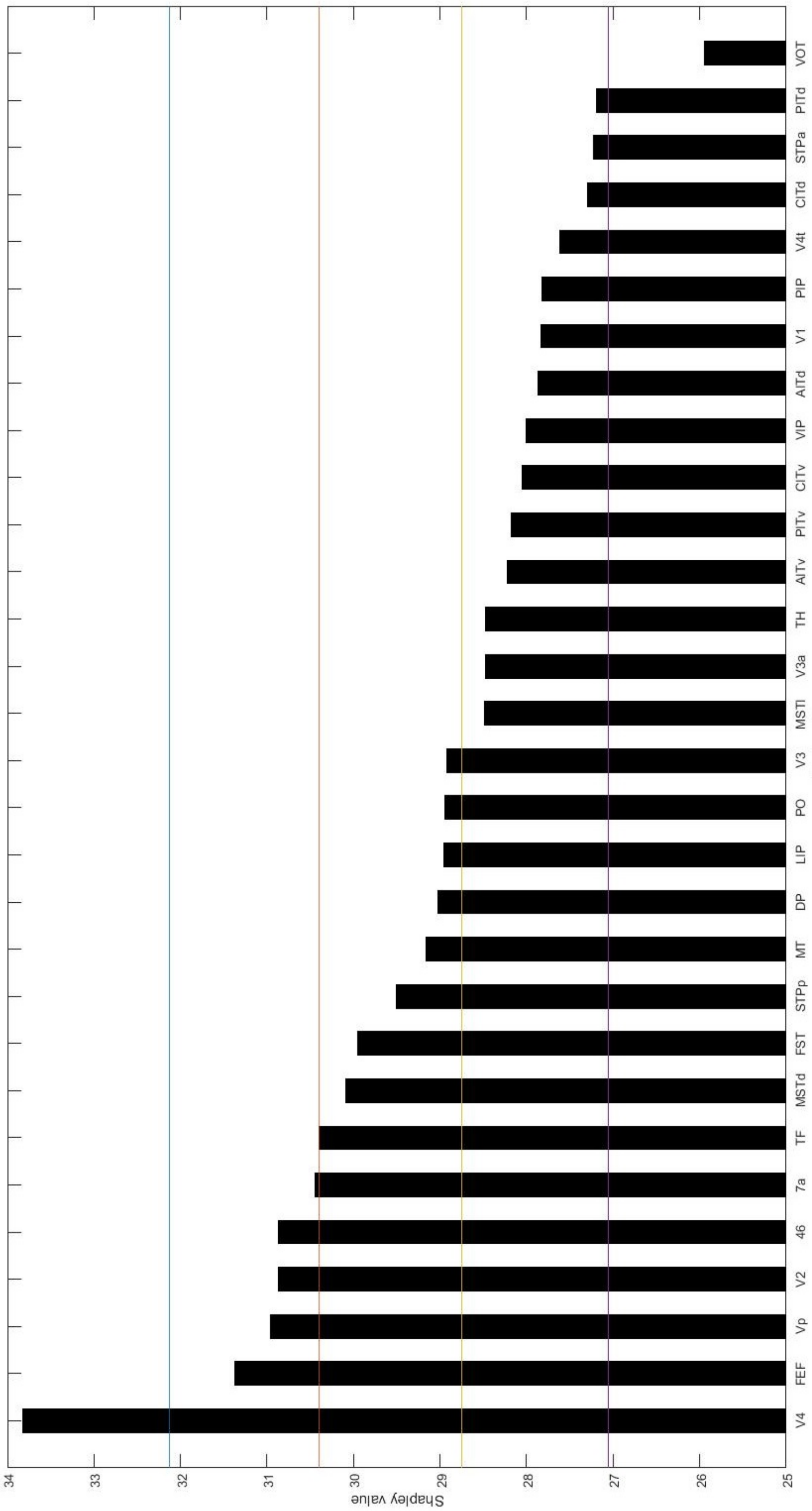


Figure 3.6: Shapley rating of the macaque visual cortex.

Chapter 4

Three-valued simple games

4.1 Introduction

In this chapter, based on Musegaas, Borm, and Quant (2015b) we analyze a class of transferable utility games, called three-valued simple games. The class of three-valued simple games is a natural extension of the class of simple games, introduced by Von Neumann and Morgenstern (1944) and widely applied in the literature to model decision rules in legislatures and other decision-making bodies. In a simple game, a coalition is either 'winning' or 'losing', i.e., there are two possible values for each coalition. The concept of three-valued simple games goes one step further than simple games in the sense that there are three, instead of only two, possible values.

This chapter formally defines the class of three-valued simple games and focuses on analyzing the core and the Shapley value of these games. We study how the results for simple games can be extended to three-valued simple games. We extend the notion of veto players in simple games, to the notion of vital players, primary vital players and secondary vital pairs in three-valued simple games. It is known that in simple games the core is fully determined by the veto players. In a similar way, we introduce the vital core which fully depends on primary/secondary vital players/pairs. The vital core is shown to be a subset of the core. We discuss a class of three-valued simple games such that the core and the vital core coincide.

Dubey (1975) characterized the Shapley value on the class of simple games. The essence of this characterization is the transfer property. We will show that the transfer property can also be used for a characterization of the Shapley value for three-valued simple games. In order to obtain this characterization we introduce a new axiom, called unanimity level efficiency. We prove that the combination of the axioms of efficiency, symmetry, the dummy property, the transfer property and unanimity level efficiency fully determines the Shapley value for a three-valued simple game. Moreover, also the logical independence of these five axioms is shown. At last, as an illustration, a parliamentary bicameral system is modelled as a three-valued simple game and analyzed on the basis of the Shapley value.

The organization of this chapter is as follows. Section 4.2 formally introduces three-valued simple games and investigates the core of such games. Section 4.3 provides a characterization for the Shapley value on the class of three-valued simple games. In Section 4.4 we end with some concluding remarks.

4.2 The core of three-valued simple games

In this section we define three-valued simple games, a new subclass of TU-games. After that, we investigate the core of such games. For this, we extend the concept of veto players in simple games, to the concept of vital players, primary vital players and secondary vital pairs in three-valued simple games.

A game $v \in \mathrm{TU}^N$ is called *three-valued simple* if

(i) $v(S) \in \{0, 1, 2\}$ for all $S \subset N$,

(ii) $v(N) = 2$,

(iii) v is monotonic.

Let TSI^N denote the class of all three-valued simple games with player set N. The concept of three-valued simple games goes one step further than simple games in the sense that there are three, instead of only two, possible values. Next to the value 0, we have chosen the values 1 and 2. Of course, the relative proportion between these two values may depend on the application at hand and the concept of three-valued simple games (and its results) can be generalized to three-valued TU-games with coalitional values 0, 1 or β with $\beta > 1$.

Recall that the core of a simple game is fully determined by its set of veto players. We characterize the core of a three-valued simple game using the concept of vital players which is similar to the concept of veto players in simple games. Proposition 4.2.1 below states that only the vital players of a three-valued simple game can receive a positive payoff in the core, while all other players receive zero. Here, for

$v \in \mathrm{TSI}^N$ the set of *vital players* is defined by

$$\mathrm{Vit}(v) = \bigcap \{S \mid v(S) = 2\}.$$

Hence, the vital players are those players who belong to every coalition with value 2.

Proposition 4.2.1. *Let $v \in TSI^N$. If $x \in C(v)$ and $i \in N\backslash Vit(v)$, then $x_i = 0$.*

Proof. Let $x \in C(v)$ and $i \in N\backslash\mathrm{Vit}(v)$. Since $i \notin \mathrm{Vit}(v)$, there exists a $T \subseteq N\backslash\{i\}$ with $v(T) = 2$. Clearly, since $x \in C(v)$, we have $x \geq 0$. Then, because of efficiency and stability of $x \in C(v)$,

$$x_i \leq v(N) - \sum_{j \in T} x_j \leq v(N) - v(T) = 0,$$

so $x_i = 0$. □

Using the concept of vital players, Proposition 4.2.2 provides a sufficient condition for emptiness of the core of a three-valued simple game.

Proposition 4.2.2. *Let $v \in TSI^N$. If $Vit(v) = \emptyset$ or $v(N\backslash Vit(v)) > 0$, then $C(v) = \emptyset$.*[1]

Proof. First, assume $\mathrm{Vit}(v) = \emptyset$. Suppose $C(v) \neq \emptyset$ and let $x \in C(v)$. Then, from Proposition 4.2.1 we know $x_i = 0$ for all $i \in N$. Consequently, $\sum_{i \in N} x_i = 0$ which contradicts the efficiency condition of $x \in C(v)$.

Second, assume $v(N\backslash\mathrm{Vit}(v)) > 0$. Suppose $C(v) \neq \emptyset$ and let $x \in C(v)$. Then, from Proposition 4.2.1 we know $x_i = 0$ for all $i \in N\backslash\mathrm{Vit}(v)$ and therefore $\sum_{i \in N\backslash\mathrm{Vit}(v)} x_i = 0 < v(N\backslash\mathrm{Vit}(v))$, which contradicts the stability condition of $x \in C(v)$. □

From Proposition 4.2.2 it follows that only the set of permissible three-valued simple games may have a non-empty core, where a game $v \in \mathrm{TSI}^N$ is called *permissible* if the following two conditions are satisfied

(i) $\mathrm{Vit}(v) \neq \emptyset$,

[1] Note that the condition is only a sufficient condition and not a necessary condition. Consider for example the game $v \in \mathrm{TSI}^N$, with $N = \{1, 2, 3\}$, given by

$$v(S) = \begin{cases} 2 & \text{if } S = N, \\ 1 & \text{otherwise.} \end{cases}$$

Then, $C(v) = \emptyset$ but $\mathrm{Vit}(v) = N \neq \emptyset$ and $v(N\backslash\mathrm{Vit}(v)) = v(\emptyset) = 0$.

(ii) $v(N\backslash \mathrm{Vit}(v)) = 0$.

However, note that a permissible three-valued simple game can still have an empty core. From now on we focus only on the set of permissible three-valued simple games and define for every permissible three-valued simple game, a reduced game where the player set is reduced to the set of vital players. We define this reduced game in such a way that the core of a permissible three-valued simple game equals the core of the reduced game, when extended with zeros for all players outside the set of vital players (see Proposition 4.2.4).

For a permissible game $v \in \mathrm{TSI}^N$ the *reduced three-valued simple game* $v_r \in \mathrm{TU}^{\mathrm{Vit}(v)}$ is defined by

$$v_r(S) = v(S \cup (N\backslash \mathrm{Vit}(v))),$$

for all $S \subseteq \mathrm{Vit}(v)$. The following proposition states that a reduced permissible game v_r is also a three-valued simple game and, interestingly, allows for only one coalition with value 2.

Proposition 4.2.3. *Let $v \in TSI^N$ be permissible. Then, $v_r \in TSI^{Vit(v)}$ with*

$$v_r(S) \in \{0, 1\},$$

for all $S \subset Vit(v)$.

Proof. From the definition of v_r it immediately follows that $v_r \in TSI^{\mathrm{Vit}(v)}$. Suppose that there exists an $S \subset \mathrm{Vit}(v)$ with $v_r(S) = 2$. Then $v(S \cup (N\backslash \mathrm{Vit}(v))) = 2$ and consequently, using the definition of $\mathrm{Vit}(v)$, we have $\mathrm{Vit}(v) \subseteq S$, which is a contradiction. □

In a reduced three-valued simple game the number of coalitions with value 2 is reduced to one, only the grand coalition has value 2, and thus $\mathrm{Vit}(v_r) = \mathrm{Vit}(v)$. This property makes it easier to characterize the core of reduced three-valued simple games compared to non-reduced three-valued simple games.

For a permissible game $v \in \mathrm{TSI}^N$ and for an $x \in \mathbb{R}^{\mathrm{Vit}(v)}$ we define $\overline{x}^0 \in \mathbb{R}^N$ as

$$\overline{x}_i^0 = \begin{cases} x_i & \text{if } i \in \mathrm{Vit}(v), \\ 0 & \text{if } i \in N\backslash \mathrm{Vit}(v). \end{cases}$$

For a set $X \subseteq \mathbb{R}^{\mathrm{Vit}(v)}$, we define $\overline{X}^0 \subseteq \mathbb{R}^N$ as $\overline{X}^0 = \{\overline{x}^0 \mid x \in X\}$.

Proposition 4.2.4. *Let $v \in TSI^N$ be permissible. Then,*

$$C(v) = \overline{C(v_r)}^0$$

Proof. ("$\subseteq$") Let $x \in C(v)$ and let $S \subseteq \text{Vit}(v)$. From Proposition 4.2.1 we have

$$\sum_{i \in S} x_i = \sum_{i \in S \cup (N \setminus \text{Vit}(v))} x_i \geq v(S \cup (N \setminus \text{Vit}(v))) = v_r(S),$$

where the inequality follows from stability of $x \in C(v)$. Because of efficiency of $x \in C(v)$ and due to Proposition 4.2.1 we have

$$\sum_{i \in \text{Vit}(v)} x_i = \sum_{i \in N} x_i = v(N) = 2 = v_r(\text{Vit}(v)),$$

where the last equality follows from Proposition 4.2.3. Hence, $x \in \overline{C(v_r)}^0$.

("$\supseteq$") Let $x \in \overline{C(v_r)}^0$ and let $S \subseteq N$. Then,

$$\sum_{i \in S} x_i = \sum_{i \in S \cap \text{Vit}(v)} x_i \geq v_r(S \cap \text{Vit}(v)) = v((S \cap \text{Vit}(v)) \cup (N \setminus \text{Vit}(v))) \geq v(S),$$

where the first inequality follows from stability of $x \in C(v_r)$ and the second inequality follows from monotonicity of v and the fact that $S \subseteq (S \cap \text{Vit}(v)) \cup (N \setminus \text{Vit}(v))$. Because of efficiency of $x \in C(v_r)$ we have

$$\sum_{i \in N} x_i = \sum_{i \in \text{Vit}(v)} x_i = v_r(\text{Vit}(v)) = 2 = v(N),$$

where the penultimate equality follows from Proposition 4.2.3. Hence, $x \in C(v)$. □

Proposition 4.2.4 states that the core of a permissible three-valued simple game follows from the core of the corresponding reduced game by extending the vectors with zeros for the non-vital players. This proposition also implies that the core of a permissible three-valued simple game is non-empty if and only if the corresponding reduced game has a non-empty core. Proposition 4.2.4 is illustrated in the following example.

Example 4.2.1. Let $N = \{1, 2, 3, 4\}$ and consider the game $v \in \text{TSI}^N$ given by

$$v(S) = \begin{cases} 2 & \text{if } S \in \{\{1,2,3\}, N\}, \\ 1 & \text{if } S \in \{\{1,2\}, \{2,3\}, \{1,2,4\}, \{2,3,4\}\}, \\ 0 & \text{otherwise.} \end{cases}$$

Note that v is permissible since $\mathrm{Vit}(v) = \{1,2,3\} \neq \emptyset$ and $v(N\backslash\mathrm{Vit}(v)) = v(\{4\}) = 0$. The corresponding reduced three-valued simple game v_r is given in Table 4.1.

S	$\{1\}$	$\{2\}$	$\{3\}$	$\{1,2\}$	$\{1,3\}$	$\{2,3\}$	$\{1,2,3\}$
$v_r(S)$	0	0	0	1	0	1	2

Table 4.1: Reduced game v_r of the game v in Example 4.2.1

Since

$$C(v_r) = \mathrm{Conv}\left(\begin{pmatrix}0\\2\\0\end{pmatrix}, \begin{pmatrix}1\\1\\0\end{pmatrix}, \begin{pmatrix}0\\1\\1\end{pmatrix}, \begin{pmatrix}1\\0\\1\end{pmatrix}\right),$$

we have, according to Proposition 4.2.4, that

$$C(v) = \mathrm{Conv}\left(\begin{pmatrix}0\\2\\0\\0\end{pmatrix}, \begin{pmatrix}1\\1\\0\\0\end{pmatrix}, \begin{pmatrix}0\\1\\1\\0\end{pmatrix}, \begin{pmatrix}1\\0\\1\\0\end{pmatrix}\right). \qquad \triangle$$

As Example 4.2.1 suggests, the extreme points of the core of three-valued simple games have a specific structure which we will describe using the notion of the vital core. The extreme points depend in particular on the set of vital players that belong to every coalition with value 1 or 2 in v_r and the set of pairs of vital players such that for every coalition with value 1 in v_r at least one player of such a pair belongs to the coalition. For a permissible three-valued simple game $v \in \mathrm{TSI}^N$ we define the set of *primary vital players* of v by

$$\mathrm{PVit}(v) = \bigcap \{S \subseteq \mathrm{Vit}(v) \mid v_r(S) \in \{1,2\}\}.$$

and define the set of *secondary vital pairs* of v by

$$\mathrm{SVit}(v) = \{\{i,j\} \subseteq \mathrm{Vit}(v)\backslash\mathrm{PVit}(v) \mid i \neq j, \{i,j\} \cap S \neq \emptyset \text{ for all } S \text{ with } v_r(S) = 1\}.$$

Using the primary vital players and the secondary vital pairs, the *vital core* $VC(v)$ of a permissible game $v \in \mathrm{TSI}^N$ is defined by

$$\begin{aligned} VC(v) = \mathrm{Conv}(&\{2e^{\{i\}} \mid i \in \mathrm{PVit}(v)\} \\ &\cup \{e^{\{i,j\}} \mid i \in \mathrm{PVit}(v), j \in \mathrm{Vit}(v)\backslash\mathrm{PVit}(v)\} \\ &\cup \{e^{\{i,j\}} \mid \{i,j\} \in \mathrm{SVit}(v)\}). \end{aligned}$$

The vital core is a subset of the core as is seen in the following theorem.

Theorem 4.2.5. *Let $v \in TSI^N$ be permissible. Then,*

$$VC(v) \subseteq C(v).$$

Proof. Due to Proposition 4.2.4 together with the fact that $C(v)$ is a convex set, it is sufficient to show that $2e^{\{i\}} \in C(v_r)$ for all $i \in \mathrm{PVit}(v)$, $e^{\{i,j\}} \in C(v_r)$ for all $i \in \mathrm{PVit}(v)$ and $j \in \mathrm{Vit}(v) \backslash \mathrm{PVit}(v)$, and $e^{\{i,j\}} \in C(v_r)$ for all $\{i,j\} \in \mathrm{SVit}(v)$.[2]

Let $i \in \mathrm{PVit}(v)$ and $S \subset \mathrm{Vit}(v)$. If $i \in S$, then

$$\sum_{k \in S} 2e_k^{\{i\}} = 2 > 1 \geq v_r(S).$$

If $i \notin S$, then $v_r(S) = 0$ and

$$\sum_{k \in S} 2e_k^{\{i\}} = 0 = v_r(S).$$

Moreover, $\sum_{k \in \mathrm{Vit}(v)} 2e_k^{\{i\}} = 2 = v_r(\mathrm{Vit}(v))$. Hence, $2e^{\{i\}}$ belongs to $C(v_r)$.

Next, let $i \in \mathrm{PVit}(v)$, $j \in \mathrm{Vit}(v) \backslash \mathrm{PVit}(v)$ and $S \subset \mathrm{Vit}(v)$. If $i \in S$, then

$$\sum_{k \in S} e_k^{\{i,j\}} \geq 1 \geq v_r(S).$$

If $i \notin S$, then $v_r(S) = 0$ and

$$\sum_{k \in S} e_k^{\{i,j\}} \geq 0 = v_r(S).$$

Moreover, $\sum_{k \in \mathrm{Vit}(v)} e_k^{\{i,j\}} = 2 = v_r(\mathrm{Vit}(v))$. Hence, $e^{\{i,j\}}$ belongs to $C(v_r)$.

Finally, let $\{i,j\} \in \mathrm{SVit}(v)$ and let $S \subset \mathrm{Vit}(v)$. If $S \cap \{i,j\} \neq \emptyset$, then

$$\sum_{k \in S} e_k^{\{i,j\}} \geq 1 \geq v_r(S).$$

If $S \cap \{i,j\} = \emptyset$, then $v_r(S) = 0$ and

$$\sum_{k \in S} e_k^{\{i,j\}} = 0 = v_r(S).$$

Moreover, $\sum_{k \in \mathrm{Vit}(v)} e_k^{\{i,j\}} = 2 = v_r(\mathrm{Vit}(v))$. Hence, $e^{\{i,j\}}$ belongs to $C(v_r)$. □

[2] By the nature of the restricted game v_r we should actually write $2e^{\{i\}}|_{\mathrm{Vit}(v)}$ and $e^{\{i,j\}}|_{\mathrm{Vit}(v)}$ since we consider the restriction of the vectors $2e^{\{i\}}$ and $e^{\{i,j\}}$ to $\mathrm{Vit}(v)$.

The following example illustrates that there are three-valued simple games for which the vital core is empty, but the core is non-empty.

Example 4.2.2. Let $N = \{1, 2, 3, 4\}$ and consider the game $v \in \text{TSI}^N$ given by

$$v(S) = \begin{cases} 2 & \text{if } |S| = 4, \\ 1 & \text{if } |S| = 2 \text{ or } |S| = 3, \\ 0 & \text{otherwise.} \end{cases}$$

Note that $\text{Vit}(v) = N$, so v is permissible and the corresponding reduced three-valued simple game v_r is the same as v. Since $v_r(\{1,2\}) = 1$ and $v_r(\{3,4\}) = 1$, we have $\text{PVit}(v) = \emptyset$. Moreover, for $i, j \in N$ with $i \neq j$, we have $v_r(N \backslash \{i, j\}) = 1$. Hence, also $\text{SVit}(v) = \emptyset$ and consequently $VC(v) = \emptyset$. However, the core of v is non-empty since $\left(\frac{1}{2}, \frac{1}{2}, \frac{1}{2}, \frac{1}{2}\right) \in C(v)$. △

As the following example shows, the vital core and the core coincide for some three-valued simple games.

Example 4.2.3. Reconsider the three-valued simple game of Example 4.2.1. From the reduced game v_r (see Table 4.1) it follows that

$$\text{PVit}(v) = \{2\}$$

and

$$\text{SVit}(v) = \{\{1, 3\}\}.$$

Therefore, the vital core of v is given by

$$VC(v) = \text{Conv}\left(\begin{pmatrix} 0 \\ 2 \\ 0 \\ 0 \end{pmatrix}, \begin{pmatrix} 1 \\ 1 \\ 0 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 1 \\ 1 \\ 0 \end{pmatrix}, \begin{pmatrix} 1 \\ 0 \\ 1 \\ 0 \end{pmatrix}\right),$$

and, using the results from Example 4.2.1, we have $VC(v) = C(V)$. △

Theorem 4.2.6 below shows that the core and the vital core coincide for the class of double unanimity games, a specific subclass of three-valued simple games. For $T_1, T_2 \in 2^N \backslash \{\emptyset\}$, we define the *double unanimity game* $u_{T_1,T_2} \in \text{TSI}^N$ by

$$u_{T_1,T_2}(S) = \begin{cases} 2 & \text{if } T_1 \subseteq S \text{ and } T_2 \subseteq S, \\ 1 & \text{if } T_1 \subseteq S, T_2 \not\subseteq S \text{ or } T_1 \not\subseteq S, T_2 \subseteq S, \\ 0 & \text{otherwise,} \end{cases}$$

for all $S \in 2^N$. Note that a three-value simple double unanimity game is a natural extension of a simple unanimity game.

Theorem 4.2.6. *Let $v \in TSI^N$ be a double unanimity game. Then,*

$$C(v) = VC(v).$$

Proof. Let $T_1, T_2 \in 2^N \backslash \{\emptyset\}$ be such that $v = u_{T_1,T_2}$ (with player set N). Observe that the reduced three-valued simple game $v_r \in \text{TSI}^{T_1 \cup T_2}$ can also be expressed by $v_r = u_{T_1,T_2}$ (with player set $T_1 \cup T_2$). Therefore, the set of primary vital players of v is given by

$$\begin{aligned}\text{PVit}(v) &= \begin{cases} T_1 & \text{if } T_1 \subseteq T_2, \\ T_2 & \text{if } T_2 \subseteq T_1, \\ T_1 \cap T_2 & \text{otherwise,} \end{cases} \\ &= T_1 \cap T_2.\end{aligned}$$

The set of secondary vital pairs of v is given by

$$\begin{aligned}\text{SVit}(v) &= \begin{cases} \emptyset & \text{if } T_1 \subseteq T_2 \text{ or } T_2 \subseteq T_1, \\ \{\{i,j\} \subseteq T_1 \cup T_2 \mid i \in T_1 \backslash T_2, j \in T_2 \backslash T_1\} & \text{otherwise,} \end{cases} \\ &= \{\{i,j\} \subseteq T_1 \cup T_2 \mid i \in T_1 \backslash T_2, j \in T_2 \backslash T_1\}.\end{aligned}$$

Consequently,

$$\begin{aligned}VC(v) = \text{Conv}(&\{2e^{\{i\}} \mid i \in T_1 \cap T_2\} \\ &\cup \{e^{\{i,j\}} \mid i \in T_1 \cap T_2, j \in T_1 \backslash T_2 \text{ or } j \in T_2 \backslash T_1\} \\ &\cup \{e^{\{i,j\}} \mid i \in T_1 \backslash T_2, j \in T_2 \backslash T_1\}),\end{aligned}$$

From Theorem 4.2.5 we already know that $VC(v) \subseteq C(v)$, so we only need to prove $C(v) \subseteq VC(v)$. Since a double unanimity game is convex, (2.3) implies that it suffices to show that

$$\begin{aligned}m^\sigma(v) \in \{2e^{\{i\}} \mid i \in T_1 \cap T_2\} \cup \{e^{\{i,j\}} \mid i \in T_1 \cap T_2, j \in T_1 \backslash T_2 \text{ or } j \in T_2 \backslash T_1\} \\ \cup \{e^{\{i,j\}} \mid i \in T_1 \backslash T_2, j \in T_2 \backslash T_1\}),\end{aligned}$$

for all $\sigma \in \Pi(N)$.

Let $\sigma \in \Pi(N)$. Since $v(S) \in \{0, 1, 2\}$ for all $S \subseteq N$ and v is monotonic, $m^\sigma(v)$ either contains one two with all other coordinates zero (case 1) or it contains two ones with all other coordinates zero (case 2). We thus distinguish between these two cases.

Case 1: $m^\sigma(v) = 2e^{\{i\}}$ for some $i \in N$. Set

$$P = \{k \in N \mid \sigma(k) < \sigma(i)\}.$$

Then $v(P) = 0$ implying that $T_1 \not\subseteq P$ and $T_2 \not\subseteq P$. Moreover, $v(P \cup \{i\}) = 2$ and therefore $T_1 \subseteq P \cup \{i\}$ and $T_2 \subseteq P \cup \{i\}$. This is only possible if $i \in T_1 \cap T_2$, consequently $m^\sigma(v) = 2e^{\{i\}}$ with $i \in T_1 \cap T_2$.

Case 2: $m^\sigma(v) = e^{\{i,j\}}$ for some $i, j \in N$, $i \neq j$. Without loss of generality assume that $\sigma(i) < \sigma(j)$. Set

$$P = \{k \in N \mid \sigma(k) < \sigma(i)\}$$

and

$$Q = \{k \in N \mid \sigma(k) < \sigma(j)\}.$$

Since $v(P \cup \{i\}) = 1$, assume without loss of generality that $T_1 \subseteq P \cup \{i\}$ and $T_2 \not\subseteq P \cup \{i\}$. Since $v(P) = 0$ we have $T_1 \not\subseteq P$ and $T_2 \not\subseteq P$. From this together with $T_1 \subseteq P \cup \{i\}$ it can be concluded that $i \in T_1$, and thus $i \in T_1 \cap T_2$ or $i \in T_1 \backslash T_2$. Since $v(Q) = 1$ and $v(Q \cup \{j\}) = 2$, it can be concluded that $T_1 \subseteq Q$, $T_2 \not\subseteq Q$ and $T_2 \subseteq Q \cup \{j\}$. This is only possible if $j \in T_2 \backslash T_1$, consequently $m^\sigma(v) = e^{\{i,j\}}$ with $i \in T_1 \cap T_2$ or $i \in T_1 \backslash T_2$ and $j \in T_2 \backslash T_1$. □

In Chapter 5 we discuss another class of three-valued simple games such that the core and the vital core coincide. Example 4.2.4 illustrates that there exist three-valued simple games for which the vital core is a strict non-empty subset of the core.

Example 4.2.4. Let $v \in \mathrm{TSI}^N$ be given by $N = \{1, \dots, 7\}$ and

$$v(S) = \begin{cases} 2 & \text{if } S \in \{N\backslash\{6\}, N\backslash\{7\}, N\}, \\ 1 & \text{if } S \notin \{N\backslash\{6\}, N\backslash\{7\}, N\} \text{ and } \{1,3,5\} \subseteq S \text{ or } \{3,4,5\} \subseteq S \text{ or} \\ & \{1,2,3,6\} \subseteq S \text{ or } \{1,3,4,7\} \subseteq S \text{ or } \{2,3,4,6,7\} \subseteq S, \\ 0 & \text{otherwise.} \end{cases}$$

Note that v is permissible since $\text{Vit}(v) = \{1,2,3,4,5\} \neq \emptyset$ and $v(N \backslash \text{Vit}(v)) = v(\{6,7\}) = 0$. Hence, $v_r \in \text{TSI}^{\text{Vit}(v)}$ is given by

$$v_r(S) = \begin{cases} 2 & \text{if } S = \text{Vit}(v), \\ 1 & \text{if } S \in \{\{1,2,3\},\{1,3,4\},\{1,3,5\},\{2,3,4\},\{3,4,5\}, \\ & \{1,2,3,4\},\{1,2,3,5\},\{1,3,4,5\},\{2,3,4,5\}\}, \\ 0 & \text{otherwise.} \end{cases}$$

From this it follows that

$$\text{PVit}(v) = \{3\}$$

and

$$\text{SVit}(v) = \{\{1,4\}\}.$$

Consequently,

$$VC(v) = \text{Conv}\left(\begin{pmatrix} 0\\0\\2\\0\\0\\0\\0 \end{pmatrix}, \begin{pmatrix} 1\\0\\1\\0\\0\\0\\0 \end{pmatrix}, \begin{pmatrix} 0\\1\\1\\0\\0\\0\\0 \end{pmatrix}, \begin{pmatrix} 0\\0\\1\\1\\0\\0\\0 \end{pmatrix}, \begin{pmatrix} 0\\0\\1\\0\\1\\0\\0 \end{pmatrix}, \begin{pmatrix} 1\\0\\0\\1\\0\\0\\0 \end{pmatrix} \right).$$

Note that $\left(0, \frac{1}{2}, \frac{1}{2}, \frac{1}{2}, \frac{1}{2}, 0, 0\right)$ belongs to $C(v)$, but does not belong to $VC(v)$ and thus $VC(v) \subset C(v)$. △

4.3 The Shapley value for three-valued simple games

In this section we analyze the Shapley value for three-valued simple games. In the context of simple games and three-valued simple games, a one-point solution concept like the Shapley value can be interpreted as a measure for the relative influence of each player.

A characterization of the Shapley value for simple games is provided by Dubey (1975). The essence of this characterization is the transfer property. Dubey (1975) proved that the unique one-point solution concept on SI^N that satisfies efficiency, symmetry, the dummy property and the transfer property is the Shapley value (Shapley and Shubik (1954)). The aim of this section is to see if the transfer property can also be used for a characterization for three-valued simple games.

The combination of the four properties efficiency, symmetry, dummy property and transfer property[3] is not sufficient to characterize the Shapley value on the class of three-valued simple games: see Example 4.5.1 in the appendix in Section 4.5. To obtain a characterization of Φ on TSI^N, we introduce an additional fifth axiom: unanimity level efficiency.

A one-point solution concept $f : \mathrm{TSI}^N \to \mathbb{R}^N$ satisfies *unanimity level efficiency* if

$$\sum_{i \in S} f_i(u_{S,T}) = 1 + \frac{1}{2} \sum_{i \in S} f_i(u_{T,T}), \tag{4.1}$$

for all $S, T \in 2^N \backslash \{\emptyset\}$ with $S \subset T$. Unanimity level efficiency intuitively states that in a double unanimity game $u_{S,T}$ with $S \subset T$ the players in S can allocate a payoff of 1 between themselves, while for the remaining payoff of 1 (assuming efficiency) the players in S and $T \backslash S$ are treated equally. Formally, the unanimity level efficiency axiom compares the aggregate payoff of coalition S within a double unanimity game $u_{S,T}$, with $S \subset T$, to half of the payoffs to S in the double unanimity game $u_{T,T}$. Note that the double unanimity game $u_{T,T}$ is a rescaling of the unanimity game u_T and half of the payoffs to S in the double unanimity game $u_{T,T}$ equals the payoffs to S in the unanimity game u_T.

The combination of the axioms of efficiency, symmetry, the dummy property, the transfer property and unanimity level efficiency fully determines the Shapley value for a three-valued simple game.

Theorem 4.3.1. *The Shapley value Φ is the unique one-point solution concept on TSI^N satisfying the axioms efficiency, symmetry, the dummy property, the transfer property and unanimity level efficiency.*[4]

Proof. [5] We first prove that, on TSI^N, the Shapley value Φ satisfies the five axioms mentioned in the theorem. From Shapley (1953) it follows that Φ satisfies efficiency, symmetry and the dummy property on TU^N. Moreover, from Dubey (1975) it follows that Φ satisfies the transfer property on TU^N. Hence, Φ also satisfies efficiency,

[3] If $v, w \in \mathrm{TSI}^N$, then both $\max\{v, w\}$ and $\min\{v, w\}$ belong to TSI^N too.

[4] A proof of the logical independence of the five axioms for $|N| = 2$ can be found in the appendix in Section 4.5.

[5] The proof is similar to the proof of Theorem II in Dubey (1975).

symmetry, the dummy property and the transfer property on TSI^N, and thus it only remains to prove that Φ satisfies unanimity level efficiency on TSI^N. Note that, for $S, T \in 2^N$ with $S \subset T$, we have

$$\Phi_i(u_{S,T}) = \begin{cases} \frac{1}{|S|} + \frac{1}{|T|} & \text{if } i \in S \\ \frac{1}{|T|} & \text{if } i \in T \backslash S, \\ 0 & \text{otherwise,} \end{cases}$$

and

$$\Phi_i(u_{T,T}) = \begin{cases} \frac{2}{|T|} & \text{if } i \in T \\ 0 & \text{otherwise,} \end{cases}$$

for all $i \in N$. Consequently,

$$\sum_{i \in S} f_i(u_{S,T}) - \frac{1}{2} \sum_{i \in S} f_i(u_{T,T}) = |S| \left(\frac{1}{|S|} + \frac{1}{|T|} \right) - \frac{1}{2}|S| \left(\frac{2}{|T|} \right) = 1.$$

To finish the proof, we show that the five axioms exactly fix the allocation vector prescribed by the solution concept for every three-valued simple game. Let $f : \text{TSI}^N \rightarrow \mathbb{R}^N$ be a one-point solution concept that satisfies efficiency, symmetry, the dummy property, the transfer property and unanimity level efficiency on TSI^N. Let $v \in \text{TSI}^N$ and define

$$\begin{aligned} \text{MWC}_1(v) &= \{S \subset N \mid v(S) = 1 \text{ and } T \subset S \Rightarrow v(T) = 0\}, \\ \text{MWC}_2(v) &= \{S \subset N \mid v(S) = 2 \text{ and } T \subset S \Rightarrow v(T) \in \{0, 1\}\}, \end{aligned}$$

and set

$$\text{MWC}(v) = \text{MWC}_1(v) \cup \text{MWC}_2(v).$$

Observe the following:

(i) If $|\text{MWC}(v)| = 0$, then $v = u_{N,N}$. From efficiency and symmetry it follows that

$$f(u_{N,N}) = \frac{2}{|N|} e^N, \tag{4.2}$$

and thus $f(u_{N,N})$ is uniquely determined.

(ii) If $|\mathrm{MWC}(v)| = 1$, then we have either $|\mathrm{MWC}_1(v)| = 0$ and $|\mathrm{MWC}_2(v)| = 1$, or $|\mathrm{MWC}_1(v)| = 1$ and $|\mathrm{MWC}_2(v)| = 0$.

If $|\mathrm{MWC}_1(v)| = 0$ and $|\mathrm{MWC}_2(v)| = 1$, then $v = u_{T,T}$ for some $T \subset N$. From efficiency, symmetry, and the dummy property it follows that $f(u_{T,T}) = \frac{2}{|T|}e^T$ and thus $f(u_{T,T})$ is uniquely determined.

On the other hand, if $|\mathrm{MWC}_1(v)| = 1$ and $|\mathrm{MWC}_2(v)| = 0$, then $v = u_{S,N}$ for some $S \subset N$. From efficiency and symmetry together with unanimity level efficiency it follows that

$$\sum_{i \in S} f(u_{S,N}) = 1 + \frac{1}{2}\sum_{i \in S} f_i(u_{N,N}) = 1 + \frac{|S|}{|N|}.$$

Then, due to efficiency and symmetry, we know $f(u_{S,N}) = \frac{1}{|S|}e^S + \frac{1}{|N|}e^N$ and thus $f(u_{S,N})$ is uniquely determined.

Now let $v \in \mathrm{TSI}^N$ such that $|\mathrm{MWC}(v)| = m$ with $m \geq 2$. We use induction to show that $f(v)$ is uniquely determined. Assume that $f(w)$ is uniquely determined for all $w \in \mathrm{TSI}^N$ with $|\mathrm{MWC}(w)| < m$. Set $\mathrm{MWC}_1(v) = \{S_1, S_2, \ldots, S_p\}$ and $\mathrm{MWC}_2(v) = \{T_1, T_2, \ldots, T_q\}$ with $p + q = m$. Note that from monotonicity of v it follows that

$$T_j \not\subseteq S_i,$$

for all $i \in \{1, \ldots, p\}$ and $j \in \{1, \ldots, q\}$. We distinguish between two cases: $q = 0$ and $q > 0$.

Case 1: $q = 0$, i.e., $|\mathrm{MWC}_2(v)| = 0$ and $p = m$. Define[6]

$$w = \max\{u_{S_2,N}, \ldots, u_{S_p,N}\} \text{ and } w' = \min\{u_{S_1,N}, w\}.$$

Then, $w, w' \in \mathrm{TSI}^N$ and

$$w(S) = \begin{cases} 2 & \text{if } S = N, \\ 1 & \text{if there exists an } i \in \{2, \ldots, p\} \text{ such that } S_i \subseteq S, \\ 0 & \text{otherwise,} \end{cases}$$

[6]Define $\max\{v_1, \ldots, v_k\}$ by $(\max\{v_1, \ldots, v_k\})(S) = \max\{v_1(S), \ldots, v_k(S)\}$ for all $S \subseteq N$.

while

$$w'(S) = \begin{cases} 2 & \text{if } S = N, \\ 1 & \text{if there exists an } i \in \{2, \dots, p\} \text{ such that } S_1 \cup S_i \subseteq S, \\ 0 & \text{otherwise.} \end{cases}$$

Consequently, we have $\text{MWC}_1(w) = \{S_2, S_3, \dots, S_p\}$ and $\text{MWC}_2(w) = \emptyset$, and thus

$$|\text{MWC}(w)| = p - 1 = m - 1.$$

Moreover, $\text{MWC}_1(w') \subseteq \{S_1 \cup S_2, S_1 \cup S_3, \dots, S_1 \cup S_p\}$ and $\text{MWC}_2(w') = \emptyset$, and thus

$$|\text{MWC}(w')| \leq p - 1 = m - 1.$$

Since $v = \max\{u_{S_1,N}, \dots, u_{S_p,N}\} = \max\{u_{S_1,N}, w\}$ and by the transfer property, we have

$$f(u_{S_1,N}) + f(w) = f(\max\{u_{S_1,N}, w\}) + f(\min\{u_{S_1,N}, w\}) = f(v) + f(w'),$$

and thus

$$f(v) = f(w) + f(u_{S_1,N}) - f(w').$$

Since by our induction hypothesis the right hand side is uniquely determined, $f(v)$ is uniquely determined.

Case 2: $q > 0$, i.e., $|\text{MWC}_2(v)| > 0$. Define

$$w = \max\{u_{S_1,N}, \dots, u_{S_p,N}, u_{T_2,T_2}, \dots, u_{T_q,T_q}\} \text{ and } w' = \min\{u_{T_1,T_1}, w\}.$$

Then, $w, w' \in \text{TSI}^N$ and

$$w(S) = \begin{cases} 2 & \text{if there exists an } i \in \{2, \dots, q\} \text{ such that } T_i \subseteq S, \\ 1 & \text{if there exists an } i \in \{1, \dots, p\} \text{ such that } S_i \subseteq S \\ & \text{and } T_j \not\subseteq S \text{ for all } j \in \{2, \dots, q\}, \\ 0 & \text{otherwise,} \end{cases}$$

while

$$w'(S) = \begin{cases} 2 & \text{if there exists an } i \in \{2, \dots, q\} \text{ such that } T_1 \cup T_i \subseteq S, \\ 1 & \text{if there exists an } i \in \{1, \dots, p\} \text{ such that } T_1 \cup S_i \subseteq S \\ & \text{and } T_j \not\subseteq S \text{ for all } j \in \{2, \dots, q\}, \\ 0 & \text{otherwise.} \end{cases}$$

Since $T_j \not\subseteq S_i$ for all $i \in \{1, \ldots, p\}$ and $j \in \{1, \ldots, q\}$, we have $\text{MWC}_1(w) = \{S_1, S_2, \ldots, S_p\}$ and $\text{MWC}_2(w) = \{T_2, T_3, \ldots, T_q\}$, and thus

$$|\text{MWC}(w)| = p + (q - 1) = m - 1.$$

Moreover, $\text{MWC}_1(w') \subseteq \{T_1 \cup S_1, T_1 \cup S_2, \ldots, T_1 \cup S_p\}$ and $\text{MWC}_2(w') \subseteq \{T_1 \cup T_2, T_1 \cup T_3, \ldots, T_1 \cup T_q\}$, and thus

$$|\text{MWC}(w')| \leq p + (q - 1) = m - 1.$$

Since $v = \max\{u_{S_1,N}, \ldots, u_{S_p,N}, u_{T_1,T_1}, u_{T_2,T_2}, \ldots, u_{T_q,T_q}\} = \max\{u_{T_1,T_1}, w\}$ and by the transfer property, we have

$$f(u_{T_1,T_1}) + f(w) = f(\max\{u_{T_1,T_1}, w\}) + f(\min\{u_{T_1,T_1}, w\}) = f(v) + f(w'),$$

and thus

$$f(v) = f(w) + f(u_{T_1,T_1}) - f(w').$$

Since by our induction hypothesis the right hand side is uniquely determined, $f(v)$ is uniquely determined too. □

Applications of the Shapley value for measuring the power in legislative procedures are abundant in the literature. For instance, Bilbao et al. (2002) considered the legislative procedure of the EU-27 Council by means of a combinatorial method based on generating functions for computing the Shapley value efficiently. Likewise, Hausken and Mohr (2001) considered the legislative procedure in the European Council of Ministers. In particular, they decomposed the Shapley value into a matrix for which the elements in each row and in each column of the matrix sum up to the Shapley value of the corresponding player.

Three-valued simple games can be used to more adequately model a bicameral legislature, in which the legislators are divided into two separate houses, the lower house and the upper house, and a bill has to be approved by both houses. Example 4.3.1 shows how a three-valued simple game can be used to model the bicameral legislature in the Netherlands.

Example 4.3.1. In the bicameral legislature of the Netherlands, the States General of the Netherlands, the lower house is called the House of Representatives and the upper house is called the Senate. The House of Representatives consists of 150 members and the Senate consists of 75 members. The members of the Senate and House of Representatives each represent a political party. The members of the House of Representatives are elected directly by the Dutch citizens. The members of the Senate however are elected indirectly by the members of the provincial councils, who, in turn, are elected directly by the Dutch citizens.

The House of Representatives can accept or reject a bill. A bill is accepted by the House of Representatives only if there is a majority in favor. When a bill is accepted by the House of Representatives it is forwarded to the Senate. Subsequently the Senate can also accept or reject a bill, again via majority voting. If the bill is also accepted by the Senate, then the bill becomes a law.[7] Note that the House of Representatives and the Senate are not symmetric in the sense that the House of Representatives also has the right to propose or to revise a bill, while the Senate does not have this right.

The party breakdown of the House of Representatives and the Senate in September 2016 can be found in Table 4.2, where a_i and b_i denote the number of members in the House of Representatives and Senate respectively for party i.

The legislature of the Netherlands can be modelled by means of the following simple game $w \in \mathrm{SI}^N$ where the set of players N is the set of parties and the value $w(S)$ of a coalition $S \subseteq N$ equals:

- 1, if the members of all parties in the coalition form a majority in both the House of Representatives and Senate,
- 0, otherwise.

Hence, we have

$$w(S) = \begin{cases} 1 & \text{if } \sum_{i \in S} a_i \geq 76 \text{ and } \sum_{i \in S} b_i \geq 38, \\ 0 & \text{otherwise,} \end{cases}$$

for all $S \subseteq N$. Note however that this game does not take into account the asymmetry of the House of Representatives and the Senate.

Alternatively, a three-valued simple game does offer the opportunity to model the asymmetric bicameral legislature of the Netherlands into a single TU-game. Consider

[7]Note that in the current (2016) composition of the States General of the Netherlands, the government (kabinet-Rutte II, a coalition of VVD and PvdA) forms a majority in the House of Representatives, but not in the Senate. Hence, in order to accept a bill, the government needs the support of other political parties as well.

Party	a_i	b_i
VVD	40	13
PvdA	36	8
SP	15	9
CDA	13	12
D66	12	10
PVV	12	9
CU	5	3
GL	4	4
SGP	3	2
PvdD	2	2
GrKO	2	0
GrBvK	2	0
50PLUS	1	2
Houwers	1	0
Klein	1	0
Van Vliet	1	0
OSF	0	1
Total	150	75

Table 4.2: Party breakdown of the House of Representatives and the Senate of the Netherlands (updated to September 2016), sources: `https://www.houseofrepresentatives.nl/members_of_parliament/parliamentary_parties` and `https://www.eerstekamer.nl/begrip/english_2`.

the three-valued simple game $v \in \mathrm{TSI}^N$ where the set of players N is again the set of parties. Now, the value $v(S)$ of a coalition $S \subseteq N$ equals:

- 2, if the members of all parties in the coalition form a majority in both the House of Representatives and Senate,
- 1, if the members of all parties in the coalition form a majority in the House of Representatives but not in the Senate,
- 0, otherwise.

Hence, we have

$$v(S) = \begin{cases} 2 & \text{if } \sum_{i\in S} a_i \geq 76 \text{ and } \sum_{i\in S} b_i \geq 38, \\ 1 & \text{if } \sum_{i\in S} a_i \geq 76 \text{ and } \sum_{i\in S} b_i < 38, \\ 0 & \text{otherwise,} \end{cases}$$

for all $S \subseteq N$. For example, since $a_{\mathrm{VVD}} + a_{\mathrm{PvdA}} = 76 \geq 76$ and $b_{\mathrm{VVD}} + b_{\mathrm{PvdA}} = 21 < 38$, we have $v(\{\mathrm{VVD}, \mathrm{PvdA}\}) = 1$. Hence, the parties VVD and PvdA have a majority in the House of Representatives but not in the Senate. Note that in this three-valued simple game a majority in the House of Representatives but not in the Senate is treated differently from a majority in the Senate but not in the House of Representatives, while in both cases the bill is not accepted. This is because the House of Representatives has the additional right to propose or to revise a bill.

Table 4.3 provides the rankings according to the Shapley values of the resulting simple and three-valued simple game. Note that generally speaking there are no major differences in the resulting rankings. So, in this case, the basic modeling via a simple game can be seen as a good approximation. △

(a) Ranking according to $\Phi(w)$

Ranking	Party	$\Phi(w)$
1.	VVD	0.2419
2.	PvdA	0.1774
3.	CDA	0.1204
4.	SP	0.1067
5.	D66	0.1001
6.	PVV	0.0927
7.	GL	0.0405
8.	CU	0.0367
9.	SGP	0.0220
10.	PvdD	0.0186
11.	50PLUS	0.0154
12.	GrKO	0.0063
12.	GrBvK	0.0063
14.	OSF	0.0057
15.	Houwers	0.0031
15.	Klein	0.0031
15.	Van Vliet	0.0031

(b) Ranking according to $\Phi(v)$

Ranking	Party	$\Phi(v)/2$
1.	VVD	0.2724
2.	PvdA	0.2156
3.	SP	0.1006
4.	CDA	0.0991
5.	D66	0.0849
6.	PVV	0.0813
7.	CU	0.0345
8.	GL	0.0324
9.	SGP	0.0198
10.	PvdD	0.0151
11.	50PLUS	0.0105
12.	GrKO	0.0089
12.	GrBvK	0.0089
14.	Houwers	0.0044
14.	Klein	0.0044
14.	Van Vliet	0.0044
16.	OSF	0.0028

Table 4.3: The rankings of the parties in the bicameral legislature of the Netherlands based on $\Phi(w)$ and $\Phi(v)$, respectively

4.4 Concluding remarks

The concept of three-valued simple games can be generalized to three-valued TU-games with coalitional values 0, 1 or β with $\beta > 1$. For such games, the notions of vital players and the vital core can be extended in a natural way. It can be shown that if $\beta \geq 2$, then it still holds that the vital core forms a subset of the core. Moreover, by appropriately modifying the unanimity level efficiency axiom, also the characterization of the Shapley value can be generalized to any class of three-valued TU-games.

Note that we stopped at three-valued TU-games instead of considering the general class of k-valued TU-games because the resulting class of games would become too large and using special types of players like veto and vital players is not possible anymore.

It would also be interesting to further analyze whether it is possible to characterize the class of all one-point solution concepts on TSI^N that satisfies efficiency, symmetry, the dummy property and the transfer property, i.e., not requiring unanimity level efficiency.

4.5 Appendix

In order to prove the logical independence of the five axioms efficiency, symmetry, the dummy property, the transfer property and unanimity level efficiency on TSI^N with $|N| = 2$, we provide five examples to show the necessity of each of the five properties. Note that if $N = \{1, 2\}$, then there are exactly nine different three-valued simple games. These games are listed in Table 4.4.

S	$\{1\}$	$\{2\}$	$\{1,2\}$
$v_1(S)$	0	0	2
$v_2(S)$	0	1	2
$v_3(S)$	0	2	2
$v_4(S)$	1	0	2
$v_5(S)$	1	1	2
$v_6(S)$	1	2	2
$v_7(S)$	2	0	2
$v_8(S)$	2	1	2
$v_9(S)$	2	2	2

Table 4.4: All possible two person three-valued simple games

Since players 1 and 2 are symmetric players in v_1, v_5, v_9 and there are no symmetric players in the other games, the symmetry property of a solution f is satisfied if and only if

$$\begin{aligned} f_1(v_1) &= f_2(v_1), \\ f_1(v_5) &= f_2(v_5), \\ f_1(v_9) &= f_2(v_9). \end{aligned}$$

Likewise, since player 1 is a dummy player in v_3 and player 2 is a dummy player in v_7 and there are no dummy players in the other games, the dummy property of a solution f is satisfied if and only if

$$\begin{aligned} f_1(v_3) &= 0, \\ f_2(v_7) &= 0. \end{aligned}$$

Moreover, the transfer property boils down to the following nine equalities

$$\begin{aligned} f(v_2) + f(v_4) &= f(v_1) + f(v_5), \\ f(v_2) + f(v_7) &= f(v_1) + f(v_8), \\ f(v_3) + f(v_4) &= f(v_1) + f(v_6), \\ f(v_3) + f(v_5) &= f(v_2) + f(v_6), \\ f(v_3) + f(v_7) &= f(v_1) + f(v_9), \\ f(v_3) + f(v_8) &= f(v_2) + f(v_9), \\ f(v_5) + f(v_7) &= f(v_4) + f(v_8), \\ f(v_6) + f(v_7) &= f(v_4) + f(v_9), \\ f(v_6) + f(v_8) &= f(v_5) + f(v_9). \end{aligned}$$

Finally, the unanimity level efficiency property boils down to the following two equalities

$$\begin{aligned} f_2(v_2) &= 1 + \tfrac{1}{2} f_2(v_1), \\ f_1(v_4) &= 1 + \tfrac{1}{2} f_1(v_1). \end{aligned}$$

Example 4.5.1 (Necessity of unanimity level efficiency)**.** Consider the one-point solution concept f, where

$$\begin{aligned} &f(v_1) = f(v_2) = f(v_4) = f(v_5) = f(v_9) = (1, 1), \\ &f(v_3) = f(v_6) = (0, 2), \\ &f(v_7) = f(v_8) = (2, 0). \end{aligned}$$

Clearly $f(v_6) \neq \Phi(v_6)$ while f satisfies efficiency, symmetry, the dummy property and the transfer property, but does not satisfy unanimity level efficiency. △

Example 4.5.2 (Necessity of the transfer property)**.** Consider the one-point solution concept f, where

$$f(v_1) = f(v_5) = f(v_9) = (1, 1),$$
$$f(v_2) = \left(\tfrac{1}{2}, 1\tfrac{1}{2}\right),$$
$$f(v_3) = f(v_6) = (0, 2),$$
$$f(v_4) = \left(1\tfrac{1}{2}, \tfrac{1}{2}\right),$$
$$f(v_7) = f(v_8) = (2, 0).$$

Clearly $f(v_6) \neq \Phi(v_6)$ while f satisfies efficiency, symmetry, the dummy property and unanimity level efficiency, but does not satisfy the transfer property. △

Example 4.5.3 (Necessity of the dummy property)**.** Consider the one-point solution concept f, where

$$f(v_1) = f(v_5) = f(v_6) = f(v_8) = f(v_9) = (1, 1),$$
$$f(v_2) = f(v_3) = \left(\tfrac{1}{2}, 1\tfrac{1}{2}\right),$$
$$f(v_4) = f(v_7) = \left(1\tfrac{1}{2}, \tfrac{1}{2}\right).$$

Clearly $f(v_6) \neq \Phi(v_6)$ while f satisfies efficiency, symmetry, the transfer property and unanimity level efficiency, but does not satisfy the dummy property. △

Example 4.5.4 (Necessity of symmetry)**.** Consider the one-point solution concept f, where

$$f(v_1) = f(v_4) = f(v_7) = (2, 0),$$
$$f(v_2) = f(v_5) = f(v_8) = (1, 1),$$
$$f(v_3) = f(v_6) = f(v_9) = (0, 2),$$

Clearly $f(v_6) \neq \Phi(v_6)$ while f satisfies efficiency, the dummy property, the transfer property and unanimity level efficiency, but does not satisfy symmetry. △

Example 4.5.5 (Necessity of efficiency)**.** Consider the one-point solution concept f, where

$$f(v_1) = f(v_2) = f(v_4) = f(v_5) = f(v_9) = (2, 2),$$
$$f(v_3) = f(v_6) = (0, 4),$$
$$f(v_7) = f(v_8) = (4, 0),$$

Clearly $f(v_6) \neq \Phi(v_6)$ while f satisfies symmetry, the dummy property, the transfer property and unanimity level efficiency, but does not satisfy efficiency. △

Chapter 5

Minimum coloring games

5.1 Introduction

Consider a set of agents who all need access to some type of facility, but some agents might be in conflict. All facilities are similar, but if two agents are in conflict, they cannot have access to the same facility. The total costs are linearly increasing with the number of facilities used, so the aim is to find the minimum number of facilities that can serve all agents. This problem can be modelled by an undirected graph, called the *conflict graph*, in which the vertices represent the agents. Two vertices are adjacent if and only if the corresponding agents are in conflict. Next, we color all vertices in such a way that any two adjacent nodes receive different colors. Finding the minimum number of facilities such that every agent has non-conflicting access to some facility, is equivalent to finding a coloring of the vertices of this conflict graph that uses the smallest number of colors. This combinatorial optimization problem is known as the *minimum coloring problem*. A survey on minimum coloring problems can, for example, be found in Randerath and Schiermeyer (2004) and Pardalos, Mavridou, and Xue (1999). An application of the minimum coloring problem is, for example, scheduling courses at secondary schools, where some courses are compulsory and other courses are electives. Courses can be scheduled in any order, but pairs of courses are in conflict in the sense that they can not be assigned to the same time slot if there is a student who has chosen both courses.

To analyze how to divide the minimal joint costs among the agents, Deng, Ibaraki, and Nagamochi (1999) introduced minimum coloring games. A minimum coloring cost game can be seen as an example of a combinatorial optimization or operations research game. In a combinatorial optimization game, the value of each coalition is obtained by solving a combinatorial optimization problem on the corresponding substructure. A survey on operations research games can be found in Borm, Hamers, and Hendrickx (2001).

In Deng et al. (1999) the existence of core elements is investigated for the more general class of combinatorial optimization cost games where the value of a coalition is defined by an integer program. They showed that such games have a non-empty core if and only if the associated linear program has an integer optimal solution. Moreover, in case of bipartite conflict graphs, they characterized the core of the induced minimum coloring games as the convex hull of the characteristic vectors of the edges in the conflict graph. Deng, Ibaraki, Nagamochi, and Zang (2000) studied total balancedness of minimum coloring games and other combinatorial optimization games. They showed that a minimum coloring game is totally balanced if and only if the underlying conflict graph is perfect. In Okamoto (2003) concave minimum coloring games are characterized in terms of forbidden subgraphs. Moreover, for this case an explicit formula of the Shapley value is provided. In Bietenhader and Okamoto (2006) core largeness, extendability, and exactness of minimum coloring games are considered. Okamoto (2008) characterized the core of minimum coloring games on perfect conflict graphs as the convex hull of the characteristic vectors of the maximum cliques in the conflict graph, which is a generalization of the result by Deng et al. (1999). Additionally, Okamoto (2008) also investigated the nucleolus, the compromise value and the Shapley value of a minimum coloring game. The most recent work on minimum coloring games is by Hamers, Miquel, and Norde (2014). They provided a necessary and sufficient condition for a conflict graph such that the induced minimum coloring game has a population monotonic allocation scheme.

The minimum coloring games studied in the works above are cost games. However, if we assume that in the initial situation no agents share facilities, i.e., every vertex has its own color, then cooperation in sharing facilities between non-conflicting agents will lead to cost savings. In this chapter, based on Musegaas, Borm, and Quant (2016b), we define minimum coloring games as cost savings games instead of cost games and we focus on conflict graphs inducing coalitional cost savings in $\{0, 1\}$ or $\{0, 1, 2\}$, i.e., we focus on simple and three-valued simple minimum coloring games. We investigate three features of simple and three-valued simple minimum coloring games. First, we characterize the class of conflict graphs inducing such games. For this characterization, a distinction is made between perfect and imperfect conflict graphs, and the concept of maximum clique is used. We show that simple minimum coloring games are always induced by perfect graphs, while three-valued simple minimum coloring games can be induced by both perfect and imperfect graphs. In particular, there is only one class of imperfect conflict graphs inducing three-valued simple minimum coloring games. We also provide an upper bound on the number of maximum cliques for conflict graphs inducing simple or three-valued simple games. Second, we characterize the core in terms of the underlying conflict graph for these games. This characterization is also based on the concept of maximum clique. Since simple minimum coloring games are always induced by perfect graphs, the characterization of the core is readily derived. We show that for three-valued simple minimum coloring games induced by imperfect conflict graphs, the core is empty. On the other hand, for three-valued simple minimum coloring games induced by perfect conflict graphs, we show that the core equals the vital core (cf. Chapter 4). This strengthens the general relation between the vital core and the core for three-valued simple minimum coloring games as provided in Theorem 4.2.5. Third, for simple minimum coloring games we study the decomposition into unanimity games and derive an elegant expression for the Shapley value.

The organization of this chapter is as follows. Section 5.2 formally introduces minimum coloring games. In Section 5.3 simple minimum coloring games are investigated. Finally, Section 5.4 analyzes three-valued simple minimum coloring games.

5.2 Minimum coloring games

In this section we formally define minimum coloring games. We also provide a survey of game-theoretic characteristics of minimum coloring games and in particular recall the characterization of the core of minimum coloring games associated to perfect conflict graphs.

Let $G = (N, E)$ be an undirected graph, called the *conflict graph*. A *coloring* of G is a mapping $\gamma : N \to \mathbb{N}$ such that $\gamma(i) \neq \gamma(j)$ for every $\{i, j\} \in E$. The natural numbers assigned to the vertices correspond to the colors assigned to the vertices. A *minimum coloring* of G is a coloring γ that uses the smallest number of colors, i.e., a coloring for which $|\{\gamma(i) \mid i \in N\}|$ is minimal. The number of colors in a minimum coloring is called the *chromatic number* of G and is denoted by $\chi(G)$. The chromatic number of a graph is strongly related to the concept of a clique. Note that all vertices in a maximum clique are mutually adjacent and therefore each of them has to receive a different color in a minimum coloring, so $\chi(G) \geq \omega(G)$, for any graph G.

By assuming that initially every vertex has its own color, the minimum coloring of conflict graph $G = (N, E)$ results in optimal cost savings for N as a whole. To tackle the allocation problem of these cost savings one can analyze an associated TU-game v^G to a minimum coloring problem with conflict graph $G = (N, E)$, where the set of players is the set of vertices. For a coalition $S \subseteq N$, $v(S)$ reflects the maximal cost savings this coalition can generate, i.e., the number of colors that are saved with respect to the initial situation where $|S|$ colors were used. Hence, the value of coalition S is obtained by solving the minimum coloring problem with conflict graph $G[S]$. Correspondingly, the *minimum coloring game* $v^G \in \mathrm{TU}^N$ induced by the conflict graph $G = (N, E)$ is defined by

$$v^G(S) = |S| - \chi(G[S]),$$

for all $S \subseteq N$.

Example 5.2.1. Consider the conflict graph $G = (N, E)$ with $N = \{1, \ldots, 5\}$ as depicted in Figure 5.1. Note that the clique number equals $\omega(G) = 3$ and the set of all maximum cliques is given by $\Omega(G) = \{\{1, 3, 5\}, \{1, 4, 5\}, \{2, 4, 5\}\}$. As $\chi(G) \geq \omega(G)$, we have $\chi(G) \geq 3$. Consider the following coloring of G with three colors given by the function $\gamma : N \to \{1, 2, 3\}$ with

$$\gamma(1) = \gamma(2) = 1,$$
$$\gamma(3) = \gamma(4) = 2,$$
$$\gamma(5) = 3.$$

We may conclude that this coloring γ is a minimum coloring of G and $\chi(G) = 3$. As a consequence, the value of the grand coalition N is given by

$$v^G(N) = 5 - \chi(G) = 2.$$

For coalition $\{1, 2\}$, the induced subgraph $G[\{1, 2\}]$ contains no edges and thus only one color is needed to color the vertices, i.e., $\chi(G[\{1, 2\}]) = 1$. Hence, the value of coalition $\{1, 2\}$ is given by

$$v^G(\{1, 2\}) = 2 - \chi(G[\{1, 2\}]) = 1.$$

The complete game v^G is given by

$$v^G(S) = \begin{cases} 2 & \text{if } \{1, 2, 3, 4\} \subseteq S, \\ 1 & \text{if } \{1, 2\} \subseteq S \text{ or } \{2, 3\} \subseteq S \text{ or } \{3, 4\} \subseteq S, \text{ and } \{1, 2, 3, 4\} \not\subseteq S, \\ 0 & \text{otherwise,} \end{cases}$$

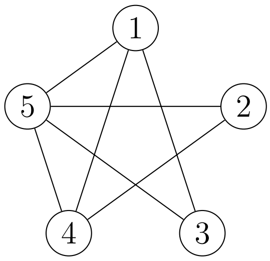


Figure 5.1: The conflict graph of Example 5.2.1

for all $S \subseteq N$. $\triangle$

Minimum coloring games are integer valued nonnegative monotonic games as is seen in the following proposition.

Proposition 5.2.1. *Let $G = (N, E)$ be a graph. Then, v^G is integer valued, nonnegative and monotonic.*

Proof. Integer valuedness and nonnegativity of v^G are straightforward consequences of the definition. As for monotonicity, let $S, T \in 2^N \backslash \{\emptyset\}$ with $S \subset T$. Note that if a minimum coloring for $G[S]$ uses $\chi(G[S])$ colors, then at most $|T \backslash S|$ additional colors are necessary for a minimum coloring of $G[T]$, i.e., $\chi(G[T]) \leq \chi(G[S]) + |T \backslash S|$. As a consequence,

$$v^G(S) = |S| - \chi(G[S]) \leq |S| - \chi(G[T]) + |T \backslash S| = |T| - \chi(G[T]) = v^G(T). \quad \square$$

A graph $G = (N, E)$ is called *perfect* if $\omega(G[S]) = \chi(G[S])$ for all $S \subseteq N$. If a graph is not perfect, then it is called *imperfect*. Chudnovsky, Robertson, Seymour, and Thomas (2006) characterized perfect graphs.

Theorem 5.2.2 (cf. Chudnovsky et al. (2006))**.** *A graph is perfect if and only if it does not contain an odd cycle graph of length at least five, or a complement of such graph, as an induced subgraph.*

Okamoto (2008) characterized the core of minimum coloring games induced by perfect conflict graphs.

Theorem 5.2.3 (cf. Okamoto (2008)). *Let $G = (N, E)$ be a perfect graph. Then,*

$$C(v^G) = Conv\left(\left\{e^{N \setminus S} \mid S \in \Omega(G)\right\}\right).^1$$

Example 5.2.2. Reconsider the conflict graph $G = (N, E)$ in Figure 5.1. Since G does not contain an odd cycle graph of length at least five, or a complement of such graph as an induced subgraph, we know from Theorem 5.2.2 that G is a perfect conflict graph. Since $\Omega(G) = \{\{1,3,5\},\{1,4,5\},\{2,4,5\}\}$, it follows from Theorem 5.2.3 that

$$C(v^G) = \text{Conv}(\{(0,1,0,1,0),(0,1,1,0,0),(1,0,1,0,0)\}).$$

△

5.3 Simple minimum coloring games

In this section we consider simple minimum coloring games. First, we characterize the class of conflict graphs inducing minimum coloring games that are simple by means of the chromatic number of the conflict graph. We also consider some other features of this class of conflict graphs, for example the number of maximum cliques. After that, we analyze the core of these induced minimum coloring games and we see that the veto players are exactly the players who are outside a maximum clique. Finally, we study the decomposition into unanimity games and derive an elegant expression for the Shapley value.

The following theorem gives a necessary and sufficient condition, in terms of the chromatic number, for a conflict graph to induce a simple minimum coloring game.

Theorem 5.3.1. *Let $G = (N, E)$ be a graph. Then, $v^G \in SI^N$ if and only if $\chi(G) = n - 1$.*

Proof. ("⇒") Let $v^G \in \text{SI}^N$. Then, $v^G(N) = 1$ and consequently $\chi(G) = n - v^G(N) = n - 1$.
("⇐") Let $\chi(G) = n-1$. Then, $v^G(N) = n - \chi(G) = 1$. According to Proposition 5.2.1 v^G is integer valued, nonnegative and monotonic, so in particular $v^G(S) \in \{0,1\}$ for all $S \subset N$, which implies $v^G \in \text{SI}^N$. □

[1] Note that the minimum coloring games studied in Okamoto (2008) are cost games, while we consider minimum coloring games as cost savings games. Hence, the minimum coloring games considered in this thesis are cost savings games with respect to the standard minimum coloring cost games studied in the literature.

Proposition 5.3.3 provides an upper bound on the number of maximum cliques for conflict graphs inducing simple games. This proposition also states that conflict graphs inducing simple games are perfect. In the proof of this proposition we use the following lemma, which gives the clique number and the chromatic number for odd cycle graphs of length at least five and their complements. The proof of this lemma is straightforward and therefore omitted.

Lemma 5.3.2. *Let* $k \in \mathbb{N}$ *with* $k \geq 2$. *Then,* $\omega(C_{2k+1}) = 2$, $\chi(C_{2k+1}) = 3$, $\omega(\overline{C_{2k+1}}) = k$ *and* $\chi(\overline{C_{2k+1}}) = k+1$.

Proposition 5.3.3. *Let* $G = (N, E)$ *be a graph. If* $v^G \in SI^N$, *then*

(i) *G is perfect,*

(ii) $|\Omega(G)| \leq 2$.

Proof. Let $v^G \in \mathrm{SI}^N$. Then, according to Theorem 5.3.1, $\chi(G) = n - 1$.

Part (i): Suppose that G is not perfect. Then, according to Theorem 5.2.2, there exists an $S \subseteq N$ such that $G[S] = C_{2k+1}$ or $G[S] = \bar{C}_{2k+1}$ with $k \geq 2$. Then, using Lemma 5.3.2, we have

$$v^G(S) = |S| - \chi(G[S]) = 2k + 1 - 3 = 2k - 2 \geq 2 > v^G(N),$$

in case $G[S] = C_{2k+1}$, or

$$v^G(S) = |S| - \chi(G[S]) = 2k + 1 - (k + 1) = k \geq 2 > v^G(N),$$

in case $G[S] = \bar{C}_{2k+1}$, which both contradict monotonicity of v^G. Hence, G is perfect.

Part (ii): Since G is perfect (see part (i)) and v^G is simple, we have $\omega(G) = \chi(G) = n - 1$. Suppose $|\Omega(G)| > 2$ and let k, l and m be three distinct vertices such that $N \backslash \{k\}$, $N \backslash \{l\}$ and $N \backslash \{m\}$ are maximum cliques of G. Since $G[N \backslash \{k\}] = K_{N \backslash \{k\}}$ and $G[N \backslash \{l\}] = K_{N \backslash \{l\}}$, we have

$$\{\{i, j\} \mid i, j \in N, i \neq j\} \backslash \{k, l\} \subseteq E.$$

Moreover, since $G[N \backslash \{m\}] = K_{N \backslash \{m\}}$ and $\{k, l\} \subseteq N \backslash \{m\}$ we have $\{k, l\} \in E$. This implies $G = K_N$ which contradicts $\chi(G) = n - 1$. Hence, $|\Omega(G)| \leq 2$. □

Note that the conditions in Proposition 5.3.3 are only sufficient conditions and not necessary conditions. Consider for example the conflict graph in Figure 5.4 in Section 5.4.1. This conflict graph is perfect and has two maximum cliques. However, this conflict graph does not induce a simple game because the value of the grand coalition in the induced minimum coloring game is 2.

Due to the fact that conflict graphs are assumed to be connected on N, we may conclude from Theorem 5.3.1 that a conflict graph inducing a simple game has at least three vertices.[2] Moreover, from the previous proposition in combination with Theorem 5.3.1 we may conclude that a conflict graph $G = (N, E)$ inducing a simple game has at least one and at most two maximum cliques of size $n-1$. So, there are two classes of conflict graphs on n vertices inducing a simple game. The first class consists of the conflict graphs with one maximum clique of size $n-1$. Note that this class consists of $n-3$ different conflict graphs (up to isomorphism[3]), because the vertex that is not in the maximum clique is adjacent to at least one vertex (because G is assumed to be connected on N) and at most $n-3$ vertices (because otherwise there are two maximum cliques). For an illustration with six vertices, see Figure 5.2(a), (b) and (c). The second class consists of the conflict graphs with two maximum cliques of size $n-1$. Note that this class consists of a unique conflict graph (up to isomorphism), namely the conflict graph with exactly one pair of vertices not being adjacent. For an illustration with six vertices, see Figure 5.2(d). Hence, for given $n \geq 3$, there are $n-2$ different conflict graphs (up to isomorphism) on n vertices inducing a simple game.

Using Theorem 5.2.3 and Proposition 5.3.3, one derives the following description of the core for simple minimum coloring games.

Corollary 5.3.4. *Let $G = (N, E)$ be a graph such that $v^G \in SI^N$.*

(i) If $\Omega(G) = \{N\backslash\{i\}\}$, then

$$C(v^G) = e^{\{i\}}.$$

(ii) If $\Omega(G) = \{N\backslash\{i\}, N\backslash\{j\}\}$ with $i \neq j$, then

$$C(v^G) = Conv\left(\left\{e^{\{i\}}, e^{\{j\}}\right\}\right).$$

[2] Note that in case a conflict graph is not connected on N, then the analysis boils down to analyzing a number of conflict graphs that are connected.

[3] Two graphs $G = (N, E)$ and $G' = (N', E')$ are called *isomorphic* if there exists a bijection $f : N \to N'$ such that $\{u, v\} \in E$ if and only if $\{f(u), f(v)\} \in E'$.

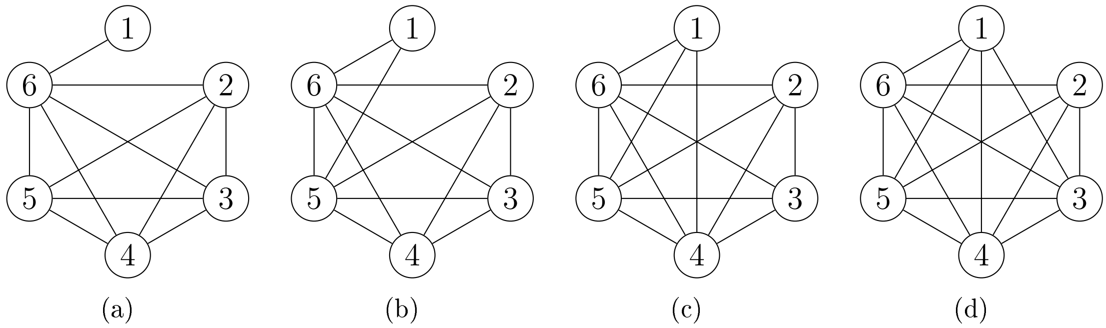


Figure 5.2: All conflict graphs (up to isomorphism) on six vertices inducing a simple game

Example 5.3.1. Consider the conflict graph $G = (N, E)$ in Figure 5.2(a), (b) or (c). Since $\Omega(G) = \{\{2, 3, 4, 5, 6\}\}$ and $v^G \in \mathrm{SI}^N$, it follows from Corollary 5.3.4 that

$$C(v^G) = \{(1, 0, 0, 0, 0, 0)\}.$$

Next, consider the conflict graph $G = (N, E)$ in Figure 5.2(d). Since $\Omega(G) = \{\{2, 3, 4, 5, 6\}, \{1, 3, 4, 5, 6\}\}$, and $v^G \in \mathrm{SI}^N$, it follows from Corollary 5.3.4 that

$$C(v^G) = \mathrm{Conv}\left(\{(1, 0, 0, 0, 0, 0), (0, 1, 0, 0, 0, 0)\}\right). \qquad \triangle$$

Not only the description of the core shows interesting features, simple minimum coloring games can also easily be decomposed into unanimity games in a structured way. The following theorem decomposes simple minimum coloring games into unanimity games. Here we use the notion of the neighborhood of a vertex in a graph (N, E). The *neighborhood* of a vertex $k \in N$ in $G = (N, E)$ is the set $\{l \mid l \in N, \{k, l\} \in E\}$ of vertices adjacent to vertex k and is denoted by $N_G(k)$. Consequently, $N_{\overline{G}}(k)$ consists of the set of vertices that are not adjacent to vertex k, excluding player k.

Theorem 5.3.5. *Let $G = (N, E)$ be a graph such that $v^G \in SI^N$.*

(i) If $\Omega(G) = \{N \backslash \{i\}\}$, then

$$v^G = \sum_{T \subseteq N_{\overline{G}}(i), T \neq \emptyset} (-1)^{|T|+1} u_{T \cup \{i\}}.$$

(ii) If $\Omega(G) = \{N \backslash \{i\}, N \backslash \{j\}\}$ with $i \neq j$, then

$$v^G = u_{\{i,j\}}.$$

Proof. As $v^G \in \mathrm{SI}^N$ we have according to Theorem 5.3.1 $\chi(G) = n - 1$.

Part (i): Assume $\Omega(G) = \{N\backslash\{i\}\}$. Then, $G[N\backslash\{i\}] = K_{N\backslash\{i\}}$, so the only missing edges in G are $\{\{i,k\} \mid k \in N_{\bar{G}}(i)\}$. Moreover, since $v^G \in \mathrm{SI}^N$ we have

$$v^G(S) = \begin{cases} 1 & \text{if } \{i,k\} \subseteq S \text{ for some } k \in N_{\bar{G}}(i), \\ 0 & \text{otherwise.} \end{cases}$$

As a consequence, if $v^G(S) = 1$ for some $S \subseteq N$, then $i \in S$ and $S \cap N_{\bar{G}}(i) \neq \emptyset$. Next, let $S \subseteq N$ and distinguish between three cases:

(a) $v^G(S) = 0$,

(b) $v^G(S) = 1$ and $S\backslash\{i\} \subseteq N_{\bar{G}}(i)$,

(c) $v^G(S) = 1$ and $S\backslash\{i\} \nsubseteq N_{\bar{G}}(i)$.

We will show that for every case we have $v^G(S) = \sum_{T \subseteq N_{\overline{G}}(i), T \neq \emptyset} (-1)^{|T|+1} u_{T \cup \{i\}}(S)$.

Case (a): $[v^G(S) = 0]$
Then, $\{i,k\} \nsubseteq S$ for every $k \in N_{\bar{G}}(k)$. So, if $i \in S$ and $T \subseteq N_{\bar{G}}(i)$, then $T \cap S = \emptyset$. As a consequence, $\{T \subseteq N_{\overline{G}}(i), T \neq \emptyset \mid T \cup \{i\} \subseteq S\} = \emptyset$ and thus

$$\sum_{T \subseteq N_{\overline{G}}(i), T \neq \emptyset} (-1)^{|T|+1} u_{T \cup \{i\}}(S) = 0 = v^G(S).$$

Case (b): $[v^G(S) = 1 \text{ and } S\backslash\{i\} \subseteq N_{\bar{G}}(i)]$
Note that since $v^G(S) = 1$ we have $i \in S$ and thus

$$\{T \subseteq N_{\overline{G}}(i), T \neq \emptyset \mid T \cup \{i\} \subseteq S\} = \{T \subseteq S\backslash\{i\}, T \neq \emptyset\}.$$

Moreover, since $v^G(S) = 1$ we have $S \cap N_{\bar{G}}(i) \neq \emptyset$ and thus $S\backslash\{i\} \neq \emptyset$. As a consequence,

$$\sum_{T \subseteq N_{\overline{G}}(i), T \neq \emptyset} (-1)^{|T|+1} u_{T \cup \{i\}}(S) = \sum_{T \subseteq S\backslash\{i\}, T \neq \emptyset} (-1)^{|T|+1}$$
$$= \sum_{r=1}^{|S|-1} \binom{|S|-1}{r} (-1)^{r+1} = \left[\sum_{r=0}^{|S|-1} \binom{|S|-1}{r} (-1)^{r+1}\right] + 1$$
$$= -\left[\sum_{r=0}^{|S|-1} \binom{|S|-1}{r} 1^{|S|-1-r} (-1)^r\right] + 1 = 1 = v^G(S),$$

where the penultimate equality follows from Newton's binomium[4].

Case (c): $[v^G(S) = 1 \text{ and } S\backslash\{i\} \not\subseteq N_{\bar{G}}(i)]$
Define $U = S \cap (N_{\overline{G}}(i) \cup \{i\})$, then

$$\{T \subseteq N_{\overline{G}}(i), T \neq \emptyset \mid T \cup \{i\} \subseteq S\} = \{T \subseteq N_{\overline{G}}(i), T \neq \emptyset \mid T \cup \{i\} \subseteq U\}.$$

Moreover, $v^G(U) = v^G(S) = 1$ and $U\backslash\{i\} \subseteq N_{\bar{G}}(i)$, so from the arguments in case (b) it follows that

$$\sum_{T \subseteq N_{\overline{G}}(i), T \neq \emptyset} (-1)^{|T|+1} u_{T \cup \{i\}}(S) = \sum_{T \subseteq N_{\overline{G}}(i), T \neq \emptyset} (-1)^{|T|+1} u_{T \cup \{i\}}(U) = v^G(U) = v^G(S).$$

Part (ii): Assume $\Omega(G) = \{N\backslash\{i\}, N\backslash\{j\}\}$ with $i \neq j$. Then, $G[N\backslash\{i\}] = K_{N\backslash\{i\}}$ and $G[N\backslash\{j\}] = K_{N\backslash\{j\}}$, so $\{\{k,l\} \mid k,l \in N, k \neq l\}\backslash\{i,j\} \subseteq E$. Suppose $\{i,j\} \in E$, then $G = K_N$ which contradicts $\chi(G) = n-1$. Hence, $\{i,j\}$ is the only missing edge in G, so $v^G \in \text{SI}^N$ is given by

$$v^G(S) = \begin{cases} 1 & \text{if } \{i,j\} \subseteq S, \\ 0 & \text{otherwise,} \end{cases}$$

i.e., $v^G = u_{\{i,j\}}$. □

Example 5.3.2. Reconsider the conflict graph $G = (N,E)$ in Figure 5.2(b). Since $\Omega(G) = \{N\backslash\{1\}\}$, $N_{\overline{G}}(1) = \{2,3,4\}$ and $v^G \in \text{SI}^N$, it follows from Theorem 5.3.5 that

$$v^G = u_{\{1,2\}} + u_{\{1,3\}} + u_{\{1,4\}} - u_{\{1,2,3\}} - u_{\{1,2,4\}} - u_{\{1,3,4\}} + u_{\{1,2,3,4\}}.$$

Next, reconsider the conflict graph $G = (N,E)$ in Figure 5.2(d). Since $\Omega(G) = \{N\backslash\{1\}, N\backslash\{2\}\}$ and $v^G \in \text{SI}^N$, it follows from Theorem 5.3.5 that $v^G = u_{\{1,2\}}$. △

Using Theorem 5.3.5 we obtain the elegant expression for the Shapley value of simple minimum coloring games provided in Corollary 5.3.6. The interpretation in case of one maximum clique $\{N\backslash\{i\}\}$ is as follows. Since all players in $N_G(i)$ are dummy players, nothing is assigned to them. Moreover, all players in $N_{\overline{G}}(i)$ are symmetric and player i is considered to be $|N_{\overline{G}}(i)|$ times more important than all players in $N_{\overline{G}}(i)$ together.

[4]Newton's binomium states that for every $a,b \in \mathbb{R}$ and for every $n \in \mathbb{N}$ we have $\sum_{r=0}^{n} \binom{n}{r} a^{n-r} b^r = (a+b)^n$.

Corollary 5.3.6. *Let $G = (N, E)$ be a graph such that $v^G \in SI^N$.*

(i) If $\Omega(G) = \{N \backslash \{i\}\}$, then for $k \in N$ we have

$$\Phi_k(v^G) = \begin{cases} \frac{|N_{\overline{G}}(i)|}{|N_{\overline{G}}(i)|+1} & \text{if } k = i, \\ \frac{1}{|N_{\overline{G}}(i)|(|N_{\overline{G}}(i)|+1)} & \text{if } k \in N_{\overline{G}}(i), \\ 0 & \text{if } k \in N_G(i). \end{cases}$$

(ii) If $\Omega(G) = \{N \backslash \{i\}, N \backslash \{j\}\}$ with $i \neq j$, then for $k \in N$ we have

$$\Phi_k(v^G) = \begin{cases} \frac{1}{2} & \text{if } k \in \{i, j\}, \\ 0 & \text{otherwise.} \end{cases}$$

Proof. Part (i): Assume $\Omega(G) = \{N \backslash \{i\}\}$ and let $k \in N$. Then, from Theorem 5.3.5 it follows that $v^G = \sum_{T \subseteq N_{\overline{G}}(i), T \neq \emptyset} (-1)^{|T|+1} u_{T \cup \{i\}}$. By (2.5) we have

$$\begin{aligned} \Phi_i(v^G) &= \sum_{T \subseteq N_{\overline{G}}(i), T \neq \emptyset} \frac{(-1)^{|T|+1}}{|T|+1} = \sum_{r=1}^{N_{\overline{G}}(i)} \frac{(-1)^{r+1}}{r+1} \binom{|N_{\overline{G}}(i)|}{r} \\ &= \sum_{r=1}^{|N_{\overline{G}}(i)|} \frac{(-1)^{r+1}}{|N_{\overline{G}}(i)|+1} \binom{|N_{\overline{G}}(i)|+1}{r+1} \\ &= \frac{1}{|N_{\overline{G}}(i)|+1} \left(\sum_{r=2}^{|N_{\overline{G}}(i)|+1} (-1)^r \binom{|N_{\overline{G}}(i)|+1}{r} \right) \\ &= \frac{1}{|N_{\overline{G}}(i)|+1} \left(\sum_{r=0}^{|N_{\overline{G}}(i)|+1} (-1)^r \binom{|N_{\overline{G}}(i)|+1}{r} - 1 + (|N_{\overline{G}}(i)|+1) \right) \\ &= \frac{1}{|N_{\overline{G}}(i)|+1} (0 - 1 + (|N_{\overline{G}}(i)|+1)) = \frac{|N_{\overline{G}}(i)|}{|N_{\overline{G}}(i)|+1}, \end{aligned}$$

where the second to last equality follows from Newton's binomium.

For $k \in N_G(i)$ we have $k \notin S$ for every $S \in \{T \cup \{i\} \mid T \subseteq N_{\overline{G}}(i), T \neq \emptyset\}$. Therefore, using (2.5), we have $\Phi_k(v^G) = 0$ for every $k \in N_G(i)$.

Because of efficiency we know $\sum_{k \in N_{\overline{G}}(i)} \Phi_k(v^G) = 1 - \frac{|N_{\overline{G}}(i)|}{|N_{\overline{G}}(i)|+1} = \frac{1}{|N_{\overline{G}}(i)|+1}$. Note that all players in $N_{\overline{G}}(i)$ are symmetric. Therefore, due to symmetry we have

$$\Phi_k(v^G) = \frac{\frac{1}{|N_{\overline{G}}(i)|+1}}{|N_{\overline{G}}(i)|} = \frac{1}{|N_{\overline{G}}(i)|(|N_{\overline{G}}(i)|+1)}.$$

Part (ii): Assume $\Omega(G) = \{N\backslash\{i\}, N\backslash\{j\}\}$ with $i \neq j$. Then, from Theorem 5.3.5 it follows that $v^G = u_{\{i,j\}}$. Therefore, from the definition of the Shapley value in (2.5) $\Phi(v^G)$ follows directly. □

Note that for a conflict graph $G = (N, E)$ that induces a simple game and has one maximum clique $N\backslash\{i\}$ we have $N_{\overline{G}}(i) \neq \emptyset$ and thus, using the previous corollary, $\Phi_i(v^G) < 1$. By Corollary 5.3.4 this implies $\Phi(v^G) \notin C(v^G)$.

5.4 Three-valued simple minimum coloring games

In this section we characterize the class of conflict graphs inducing three-valued simple minimum coloring games by means of the chromatic number of the conflict graph. For this, a distinction is made between perfect and imperfect conflict graphs. After that, we characterize the core in terms of the underlying conflict graph for these games.

The following theorem gives a necessary and sufficient condition, in terms of the chromatic number, for a conflict graph to induce a three-valued simple game.

Theorem 5.4.1. *Let $G = (N, E)$ be a graph. Then, $v^G \in TSI^N$ if and only if $\chi(G) = n - 2$.*[5]

Proof. ("⇒") Let $v^G \in \text{TSI}^N$. Then, $v^G(N) = 2$ and consequently $\chi(G) = n - v^G(N) = n - 2$.
("⇐") Let $\chi(G) = n-2$. Then, $v^G(N) = n-\chi(G) = 2$. According to Proposition 5.2.1 v^G is integer valued, nonnegative and monotonic, so in particular $v^G(S) \in \{0, 1, 2\}$ for all $S \subset N$, which implies $v^G \in \text{TSI}^N$. □

From now on, we distinguish between two classes of conflict graphs inducing three-valued simple minimum coloring games: perfect conflict graphs (Section 5.4.1) and imperfect conflict graphs (Section 5.4.2). For both classes, we consider in more detail the structure of these conflict graphs and the cores of the induced minimum coloring games.

[5]Note that Theorem 5.3.1 and Theorem 5.4.1 can be generalized for a more general class of integer valued, nonnegative and monotonic games.

5.4.1 Three-valued simple minimum coloring games induced by perfect conflict graphs

In this section we consider three-valued simple minimum coloring games induced by perfect conflict graphs. We consider a specific feature of this class of conflict graphs, namely the number of maximum cliques. After that we analyze the core and the vital core and we see that the core and the vital core coincide.

We start with providing an upper bound on the number of maximum cliques for perfect conflict graphs inducing three-valued simple games.

Proposition 5.4.2. *Let $G = (N, E)$ be a perfect graph. If $v^G \in TSI^N$, then $|\Omega(G)| \leq 4$.*

Proof. Let $v^G \in \text{TSI}^N$. Then, using Theorem 5.4.1, $\chi(G) = n - 2$. Hence, due to the fact that G is assumed to be connected on N, we have $n \geq 4$. Moreover, since G is perfect, we have $\omega(G) = \chi(G) = n - 2$, so at least two pairs of vertices are not adjacent in G. Without loss of generality we can assume that either $\{1, 2\} \notin E$ and $\{3, 4\} \notin E$, or $\{1, 2\} \notin E$ and $\{2, 3\} \notin E$. Therefore, we distinguish between these two cases and we show that for both cases we have $|\Omega(G)| \leq 4$.

Case (a): $[\{1, 2\} \notin E$ and $\{3, 4\} \notin E]$
Then the sets of vertices that can possibly form a maximum clique are

$$\{T \subseteq N \mid |T| = n - 2, \{1, 2\} \nsubseteq T, \{3, 4\} \nsubseteq T\}.$$

Therefore, a maximum clique is of the form $N \backslash \{i, j\}$ with $i \in \{1, 2\}$ and $j \in \{3, 4\}$. Hence, there are only four sets of vertices that can possibly form a maximum clique, i.e., $|\Omega(G)| \leq 4$.

Case (b): $[\{1, 2\} \notin E$ and $\{2, 3\} \notin E]$
Since $N \backslash \{2\}$ cannot form a clique, we know that there exists a pair of vertices $\{i, j\} \subseteq N \backslash \{2\}$ with $i \neq j$ and $\{i, j\} \notin E$. Hence, without loss of generality we can assume that $\{1, 3\} \notin E$, $\{3, 4\} \notin E$ or $\{4, 5\} \notin E$ (the last case is only possible if $n \geq 5$). Note that if $\{3, 4\} \notin E$ or $\{4, 5\} \notin E$, then we are back to case (a) and thus $|\Omega(G)| \leq 4$. If $\{1, 3\} \notin E$, then the sets of vertices that can possibly form a maximum clique are

$$\{T \subseteq N \mid |T| = n - 2, \{1, 2\} \nsubseteq T, \{1, 3\} \nsubseteq T, \{2, 3\} \nsubseteq T\}.$$

Therefore, a maximum clique is of the form $N \backslash \{i,j\}$ with $i \neq j$ and $\{i,j\} \subseteq \{1,2,3\}$. Hence, there are only three sets of vertices that can possibly form a maximum clique, i.e., $|\Omega(G)| \leq 3$. □

Note that the condition in Proposition 5.4.2 is only a sufficient condition and not a necessary condition. Consider for example the conflict graphs in Figure 5.2 in Section 5.3, which all are perfect and all have at most two maximum cliques. However, none of the conflict graphs induces a three-valued simple game because the value of the grand coalition in every induced minimum coloring game is 1.

Due to the fact that conflict graphs are assumed to be connected on N, we may conclude from Theorem 5.4.1 that a conflict graph inducing a three-valued simple game has at least four vertices. Moreover, from the previous proposition we may also conclude that a perfect conflict graph $G = (N, E)$ inducing a three-valued simple game has at most four maximum cliques of size $n - 2$. Figure 5.3 depicts all perfect conflict graphs (up to isomorphism) on four vertices inducing a three-valued simple game. All conflict graphs have a clique number of two. The conflict graph in Figure 5.3(a) has four maximum cliques and the conflict graphs in Figure 5.3(b) and (c) have three maximum cliques.

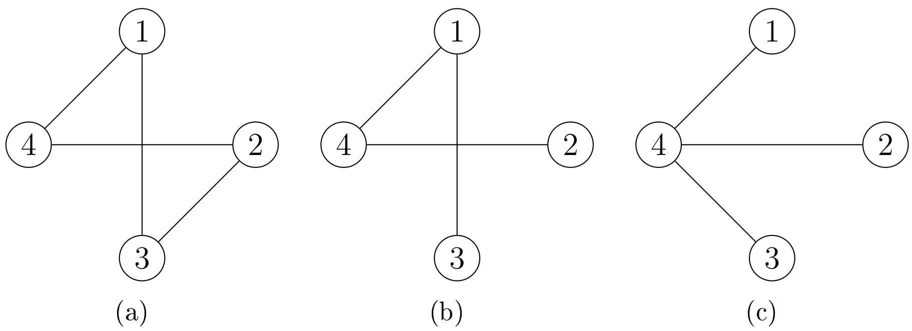


Figure 5.3: All perfect conflict graphs (up to isomorphism) on four vertices inducing a three-valued simple game

The following theorem provides a characterization of the vital core (cf. Section 4.2) for three-valued simple games induced by perfect conflict graphs.

Theorem 5.4.3. *Let $G = (N, E)$ be a perfect graph. If $v^G \in TSI^N$, then*

$$VC(v^G) = Conv\left(\left\{e^{N \backslash S} \mid S \in \Omega(G)\right\}\right).$$

Proof. Let $v^G \in \mathrm{TSI}^N$. Then, using Theorem 5.4.1 and the fact that G is perfect, we have $\omega(G) = \chi(G) = n-2$. In this proof we denote the intersection of all maximum cliques in G by Ω_G, i.e.,

$$\Omega_G = \bigcap\{S \mid S \in \Omega(G)\}.$$

We divide the proof into proving the following four statements:

(i) $\mathrm{Vit}(v^G) = N\backslash\Omega_G$,

(ii) v^G is permissible,

(iii) $\mathrm{PVit}(v^G) = \emptyset$,

(iv) $\mathrm{SVit}(v^G) = \{N\backslash S \mid S \in \Omega(G)\}$.

Note that if the statements (i) - (iv) all hold, then it immediately follows from the definition of the vital core that $VC(v^G) = \mathrm{Conv}\left(\left\{e^{N\backslash S} \mid S \in \Omega(G)\right\}\right)$.

Part (i): [$\mathrm{Vit}(v^G) = N\backslash\Omega_G$]
("$\subset$") Let $i \in \Omega_G$, i.e., i belongs to every maximum clique of G. This implies that if vertex i is removed, then the clique number decreases with one. Therefore,

$$\begin{aligned} v^G(N\backslash\{i\}) &= n-1-\chi(G[N\backslash\{i\}]) = n-1-\omega(G[N\backslash\{i\}]) \\ &= n-1-(\omega(G)-1) = n-1-(\chi(G)-1) = n-1-(n-3) = 2, \end{aligned}$$

where the second and the fourth equalities follow from the fact that G is perfect. Hence, there exists an $S \subseteq N\backslash\{i\}$ such that $v^G(S) = 2$, so $i \notin \mathrm{Vit}(v^G)$.
("$\supset$") Let $i \in N\backslash\Omega_G$, i.e., there exists a maximum clique of G to which i does not belong. This implies that if vertex i is removed, then the clique number does not change. Therefore,

$$\begin{aligned} v^G(N\backslash\{i\}) &= n-1-\chi(G[N\backslash\{i\}]) = n-1-\omega(G[N\backslash\{i\}]) \\ &= n-1-\omega(G) = n-1-\chi(G) = n-1-(n-2) = 1, \end{aligned}$$

where the second and the fourth equalities follow from the fact that G is perfect. Moreover, from monotonicity of v^G it follows that $v^G(S) \leq 1$ for all $S \subseteq N\backslash\{i\}$, so there does not exist an $S \subseteq N\backslash\{i\}$ with $v(S) = 2$. As a consequence, for all $S \subseteq N$ with $v(S) = 2$ we have $i \in S$ and thus $i \in \mathrm{Vit}(v^G)$.

Part (ii): [v^G is permissible]
Since $\omega(G) = n-2$, we know that N cannot be a maxium clique and thus $\Omega_G \neq N$. Hence, $\text{Vit}(v^G) = N \backslash \Omega_G \neq \emptyset$. Moreover, since Ω_G is the intersection of all maximum cliques in G, we know that Ω_G forms a clique as well and thus $\chi(G[\Omega_G]) = |\Omega_G|$. As a consequence, $v^G(N \backslash \text{Vit}(v^G)) = v^G(\Omega_G) = 0$. This implies that v^G is permissible.

Part (iii): [$\text{PVit}(v^G) = \emptyset$]
Since $\text{Vit}(v^G) = N \backslash \Omega_G$, we know that for every vital player there exists a maximum clique of G to which this player does not belong. As a consequence, for $i \in \text{Vit}(v^G)$, we have

$$\chi(G[N \backslash \{i\}]) = \omega(G[N \backslash \{i\}]) = \omega(G) = n-2,$$

where the first equality follows from the fact that G is perfect and the second equality follows from the fact that there exists a maximum clique of G to which i does not belong. Therefore,

$$\begin{aligned} v_r^G(\text{Vit}(v^G) \backslash \{i\}) &= v^G(\text{Vit}(v^G) \backslash \{i\} \cup (N \backslash \text{Vit}(v^G))) = v^G(N \backslash \{i\}) \\ &= n-1-\chi(G[N \backslash \{i\}]) = n-1-(n-2) = 1, \end{aligned}$$

for all $i \in \text{Vit}(v^G)$. As a consequence,

$$\text{PVit}(v^G) = \bigcap \left\{S \subseteq \text{Vit}(v) \mid v_r^G(S) \in \{1,2\}\right\} \subseteq \bigcap \left\{\text{Vit}(v^G) \backslash \{i\} \mid i \in \text{Vit}(v^G)\right\} = \emptyset.$$

Part (iv): [$\text{SVit}(v^G) = \{N \backslash S \mid S \in \Omega(G)\}$]
("$\supseteq$") Let $S \in \Omega(G)$. Since $\omega(G) = n-2$, we can denote $N \backslash S = \{i,j\}$ with $i \neq j$. Note $\{i,j\} \cap \Omega_G = \emptyset$ and thus $\{i,j\} \subseteq \text{Vit}(v^G)$. Moreover, since $\text{PVit}(v^G) = \emptyset$, we have $\{i,j\} \subseteq \text{Vit}(v^G) \backslash \text{PVit}(v^G)$. Suppose $\{i,j\} \notin \text{SVit}(v^G)$, then it follows from the definition of secondary vital pairs that there exists a $T \subseteq \text{Vit}(v^G) \backslash \{i,j\}$ with $v_r^G(T) = 1$. Since $T \subseteq \text{Vit}(v^G) \backslash \{i,j\}$, we have

$$(T \cup \Omega_G) \subseteq ((\text{Vit}(v^G) \backslash \{i,j\}) \cup \Omega_G) = N \backslash \{i,j\} = S.$$

Moreover, since S forms a maximum clique in G, we have $\chi(G[T \cup \Omega_G]) = |T \cup \Omega_G|$ and thus

$$\begin{aligned} v_r^G(T) &= v^G(T \cup (N \backslash \text{Vit}(v^G))) = v^G(T \cup \Omega_G) = |T \cup \Omega_G| - \chi(G[T \cup \Omega_G]) \\ &= |T \cup \Omega_G| - |T \cup \Omega_G| = 0, \end{aligned}$$

which contradicts $v_r^G(T) = 1$. Hence, $\{i,j\} \in \text{SVit}(v^G)$.
("$\subset$") Let $\{i,j\} \in \text{SVit}(v^G)$ and suppose $N\backslash\{i,j\}$ does not form a maximum clique in G, i.e., $\omega(G[N\backslash\{i,j\}]) < n-2$. Then,

$$\begin{aligned} v_r^G(\text{Vit}(v^G)\backslash\{i,j\}) &= v^G((\text{Vit}(v^G)\backslash\{i,j\}) \cup (N\backslash\text{Vit}(v^G))) = v^G(N\backslash\{i,j\}) \\ &= (n-2) - \chi(G[N\backslash\{i,j\}]) = (n-2) - \omega(G[N\backslash\{i,j\}]) \\ &> (n-2) - (n-2) = 0, \end{aligned}$$

where the penultimate equality again follows from the fact that G is perfect. Consequently, using the fact that a reduced game allows for only one coalition with value 2, namely the grand coalition $\text{Vit}(v^G)$, we have $v_r^G(\text{Vit}(v^G)\backslash\{i,j\}) = 1$ which contradicts the assumption that $\{i,j\} \in \text{SVit}(v^G)$. □

Example 5.4.1. Consider the perfect conflict graph $G = (N, E)$ with $N = \{1, \ldots, 6\}$ as depicted in Figure 5.4. Since $\Omega(G) = \{\{1,3,5,6\}, \{1,4,5,6\}\}$ and $v^G \in \text{TSI}^N$, it follows from Theorem 5.4.3 that

$$VC(v^G) = \text{Conv}\left(\{(0,1,0,1,0,0), (0,1,1,0,0,0)\}\right). \qquad \triangle$$

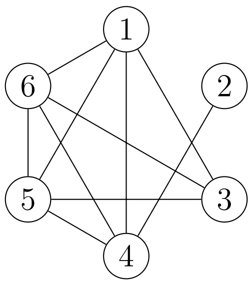


Figure 5.4: The conflict graph of Example 5.4.1

By combining Theorem 5.2.3 and Theorem 5.4.3 the general relation between the vital core and the core, as provided in Theorem 4.2.5, can be strengthened for three-valued simple minimum coloring games, i.e., for three-valued simple minimum coloring games induced by perfect conflict graphs the core equals the vital core.

Corollary 5.4.4. *Let $G = (N, E)$ be a perfect graph. If $v^G \in TSI^N$, then*

$$C(v^G) = VC(v^G).$$

5.4.2 Three-valued simple minimum coloring games induced by imperfect conflict graphs

In this section we consider three-valued simple minimum coloring games induced by imperfect conflict graphs. First, we characterize this class of conflict graphs. Moreover, we show that for a given number $n \geq 5$ of vertices the class of imperfect conflict graphs on n vertices inducing a three-valued simple game consists of a unique conflict graph. Next, we show that the induced minimum coloring games are never permissible and thus always have an empty core.

Note that all imperfect graphs have at least five vertices, because it must contain an odd cycle graph of length at least five, or a complement of such graph as an induced subgraph (cf. Theorem 5.2.2). The following theorem provides a necessary and sufficient condition for an imperfect conflict graph to induce a three-valued simple game. For this theorem, we use the notion of a dominating vertex. A vertex $i \in N$ is called *dominating* in G if $\{j \mid j \in N, \{i,j\} \in E\} = N \backslash \{i\}$, i.e., if i is adjacent to every other vertex.

Theorem 5.4.5. *Let $G = (N, E)$ be an imperfect graph. Then, $v^G \in TSI^N$ if and only if there exists an $S \subseteq N$ such that $G[S] = C_5$ and all vertices outside S are dominating.*

Proof. ("$\Leftarrow$") Let $S \subseteq N$ be such that $G[S] = C_5$ and let all vertices outside S be dominating. Since all vertices in $N \backslash S$ are dominating, we have $G[N \backslash S] = K_{N \backslash S}$ and thus $\chi(G[N \backslash S]) = n - 5$. Moreover, since each vertex in S is adjacent to each vertex in $N \backslash S$, we have

$$\chi(G) = \chi(G[N \backslash S]) + \chi(G[S]) = n - 5 + 3 = n - 2,$$

where the second equality follows from Lemma 5.3.2. Consequently, using Theorem 5.4.1, we have $v^G \in \mathrm{TSI}^N$.

("$\Rightarrow$") Let $v^G \in \mathrm{TSI}^N$. Since G is not perfect, we know from Theorem 5.2.2 that there exists an $S \subseteq N$ such that $G[S] = C_{2k+1}$ or $G[S] = \bar{C}_{2k+1}$ with $k \geq 2$. Suppose $k > 2$. Then, using Lemma 5.3.2, we have

$$v^G(S) = |S| - \chi(G[S]) = 2k + 1 - 3 = 2k - 2 > 2,$$

in case $G[S] = C_{2k+1}$, or

$$v^G(S) = |S| - \chi(G[S]) = 2k + 1 - (k + 1) = k > 2,$$

in case $G[S] = \bar{C}_{2k+1}$, which both contradict v^G being a three-valued simple game. Hence, $k = 2$ and thus $G[S] = C_5$ or $G[S] = \bar{C}_5$. Since C_5 and $\bar{C}_5$ are isomorphic to each other[6], and thus both graphs have the same clique and chromatic number, we can conclude that $G[S] = C_5$.

Now, suppose $\chi(G[N\backslash S]) < |N\backslash S| = n - 5$. Then

$$v^G(N) = n - \chi(G) \geq n - (\chi(G[S]) + \chi(G[N\backslash S])) > n - (3 + n - 5) = 2,$$

which contradicts v^G being a three-valued simple game. Hence, we may assume $\chi(G[N\backslash S]) = n - 5$ and thus $G[N\backslash S] = K_{N\backslash S}$. As a consequence, all players in $N\backslash S$ mutually adjacent, i.e.,

$$\{j \mid j \in N\backslash S, \{i,j\} \in E\} = (N\backslash S)\backslash\{i\},$$

for all $i \in N\backslash S$.

Next, suppose $\{i,j\} \notin E$ for some $i \in S$ and $j \in N\backslash S$. Then, since there exists a vertex in S that is not adjacent to a vertex in $N\backslash S$, those two players can receive the same color in the minimum coloring of G. Hence, $\chi(G) < \chi(G[S]) + \chi(G[N\backslash S])$ and thus

$$v^G(N) = n - \chi(G) > n - (\chi(G[S]) + \chi(G[N\backslash S])) = n - (3 + n - 5) = 2,$$

which again contradicts v^G being a three-valued simple game. Hence, we may assume that every player in $N\backslash S$ is adjacent to all players in S, i.e.,

$$\{j \mid j \in S, \{i,j\} \in E\} = S,$$

for all $i \in N\backslash S$. Consequently,

$$\{j \mid j \in N, \{i,j\} \in E\} = N\backslash\{i\},$$

for all $i \in N\backslash S$, i.e., all vertices outside S are dominating. □

[6]To see that C_5 and $\bar{C}_5$ are isomorphic to each other, consider the graphs $G = (N, E)$ and $G' = (N', E')$ with $N = N' = \{1, \dots, 5\}$,

$$E = \{\{1,2\}, \{1,5\}, \{2,3\}, \{3,4\}, \{4,5\}\},$$

and

$$E' = \{\{1,3\}, \{1,4\}, \{2,4\}, \{2,5\}, \{3,5\}.$$

If one takes the bijection $f : N \to N'$ with $f(1) = 1$, $f(2) = 3$, $f(3) = 5$, $f(4) = 2$ and $f(5) = 4$, then $\{u,v\} \in E$ if and only if $\{f(u), f(v)\} \in E'$.

The previous theorem implies that, for given $n \geq 5$, the class of imperfect conflict graphs on n vertices inducing a three-valued simple game consists of a unique conflict graph (up to isomorphism), namely the conflict graph that contains C_5 as an induced subgraph and all other vertices being dominated. Figure 5.5 depicts all such imperfect conflict graphs (up to isomorphism) on five, six and seven vertices.

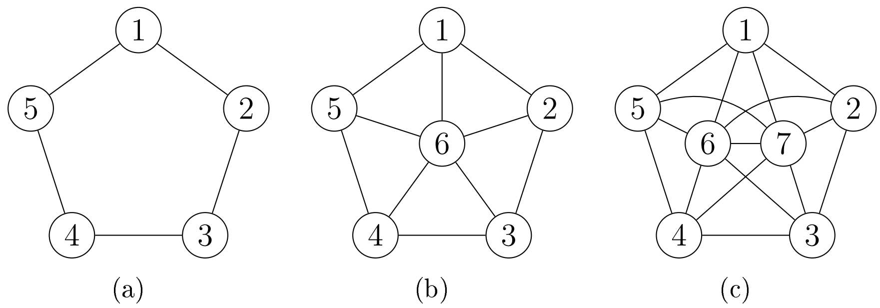


Figure 5.5: All imperfect conflict graphs (up to isomorphism) on five, six or seven vertices inducing a three-valued simple game

Using Theorem 5.4.5, the clique number and the number of maxiumum cliques for imperfect conflict graphs inducing three-valued simple games immediately follows, as is stated in the following corollary.

Corollary 5.4.6. *Let $G = (N, E)$ be an imperfect graph. If $v^G \in TSI^N$, then $\omega(G) = n - 3$ and $|\Omega(G)| = 5$.*

For three-valued simple minimum coloring games induced by imperfect conflict graphs, the core is empty as is seen in the following theorem.

Theorem 5.4.7. *Let $G = (N, E)$ be an imperfect graph. If $v^G \in TSI^N$, then v^G is not permissible and thus $C(v^G) = \emptyset$.*

Proof. Let $v^G \in \text{TSI}^N$. Then, Theorem 5.4.5 implies that there exists an $S \subseteq N$ such that $G[S] = C_5$. For $i \in S$, we have

$$v^G(S\backslash\{i\}) = 4 - \chi(G[S\backslash\{i\}]) = 4 - 2 = 2.$$

As a consequence,

$$\text{Vit}(v^G) = \bigcap\{S \subseteq N \mid v^G(S) = 2\} \subseteq \bigcap\{S\backslash\{i\} \mid i \in S\} = \emptyset,$$

so v^G is not permissible. Since only three-valued simple games that are permissible can have a non-empty core (see Proposition 4.2.2), we have $C(v^G) = \emptyset$. $\square$

Chapter 6

Step out - Step in sequencing games

6.1 Introduction

In this chapter one-machine sequencing situations are considered with a queue of players in front of a single machine, each with one job to be processed. Such a situation specifies for each player the processing time, time the machine takes to process the corresponding job of this player. In addition, it is assumed that each player has a linear cost function specified by an individual cost parameter. To minimize total joint costs, Smith (1956) showed that the players must be ordered with respect to weakly decreasing urgency, defined as the ratio of the individual cost parameter and the processing time. Assuming the presence of an initial order, this reordering will lead to cost savings. To analyze how to divide the maximal cost savings among the players, Curiel et al. (1989) introduced cooperative sequencing games. They show that sequencing games are convex and therefore have a non-empty core. This means that it is always possible to find a coalitionally stable cost savings division.

Several variants of classical classes of sequencing problems and games have been discussed in the literature. These classes are all based on different features of the underlying sequencing situations, for example by imposing restrictive assumptions on the jobs, grouping of jobs, stochastic data, dynamic/multistage situations, multiple jobs and/or machines. Namely, Hamers, Borm, and Tijs (1995) imposed ready times on the jobs, Borm, Fiestras-Janeiro, Hamers, Sánchez, and Voorneveld (2002) imposed due dates on the jobs and Hamers, Klijn, and Van Velzen (2005) imposed precedence constraints on the jobs. Both Gerichhausen and Hamers (2009) and Grundel, Çiftçi, Borm, and Hamers (2013) applied grouping of jobs, but in a different way. Gerichhausen and Hamers (2009) considered partitioning sequencing games where certain batches of jobs have some privilege over other batches of jobs. Grundel et al. (2013) introduced family sequencing situations and included the concept of set-up times in their model. Alparslan-Gök, Branzei, Fragnelli, and Tijs (2013) and Klijn and Sánchez (2006) considered stochastic data in sequencing games. Alparslan-Gök et al. (2013) considered sequencing games where uncertainty in the parameters (costs per time unit and/or processing time) is involved by means of interval data. Klijn and Sánchez (2006) considered uncertainty sequencing situations, i.e., sequencing situations in which no initial order is specified. In multi-stage sequencing situations, the order arrived at after each stage becomes the starting order for the next stage. Multi-stage sequencing situations are for example considered by Curiel (2015). In Lohmann, Borm, and Slikker (2014) another type of dynamic sequencing situations is considered, in which a player enters the system at the moment the player starts to prepare the machine for his job (there is a predecessor dependent set-up time) and leaves the system as soon as his job is finished. Çiftçi, Borm, Hamers, and Slikker (2013) considered machines which can simultaneously process multiple jobs. Multiple machine sequencing games are for example discussed in Estévez-Fernández, Mosquera, Borm, and Hamers (2008), Slikker (2006a) and Slikker (2005). Key problems in all of the above literature are finding optimal orders, finding allocation rules and deriving properties of the corresponding cooperative games.

A common assumption underlying the definition of the values of the coalitions in sequencing games is that two players of a certain coalition can only swap their positions if all players between them are also members of the coalition. Curiel, Potters, Prasad, Tijs, and Veltman (1993) argued that the resulting set of admissible reorderings for a coalition is too restrictive because there may be more reorderings possible which do not hurt the interests of the players outside the coalition. Relaxed sequencing games arise by relaxing the classical assumption about the set of admissible rearrangements for coalitions in a consistent way. In Curiel et al. (1993) four different relaxed sequencing games are introduced. These relaxations are based on requirements for the players outside the coalition regarding either their position in the processing order (position unchanged/may change) or their starting time (starting time unchanged/not increased). This means that a player in a certain coalition is allowed to jump over players outside the coalition as long as the exogenously imposed requirements are satisfied. As a consequence, a player may be moved to a position earlier in the processing order when another player moves backwards. Slikker (2006b) proved non-emptiness of the core for all four types of relaxed sequencing games considered in Curiel et al. (1993). In Van Velzen and Hamers (2003) two further classes of relaxed sequencing games are considered. In the first class there is a specific player allowed to switch with a player in front of him in the processing order if this player has a larger processing time, and with a player behind him in the processing order if this player has a smaller processing time. In the second class there are fixed time slots and thus only jobs with equal processing times can be switched. Van Velzen and Hamers (2003) proved that both classes of relaxed sequencing games have a non-empty core. Note that non-emptiness of the core of the first class actually also follows from Slikker (2006b). In fact, a lot of attention has been paid to non-emptiness of the core of relaxed sequencing games. However, surprisingly enough, up to now for none of the relaxed sequencing games described above attention has been paid to finding polynomial time algorithms determining optimal processing orders for all possible coalitions. In relaxed sequencing games the values of coalitions become larger because the set of admissible rearrangements is larger than in the classical case. As a consequence, while classical sequencing games are convex, relaxed sequencing games might not be convex anymore. To the best of our knowledge there is no general convexity result with respect to specific subclasses of relaxed sequencing games.

In this chapter, based on Musegaas, Borm, and Quant (2015a) and Musegaas, Borm, and Quant (2016a), another class of relaxed sequencing games is introduced: Step out - Step in (SoSi) sequencing games. This relaxation is intuitive from a practical point of view, because in this relaxation a member of a coalition is also allowed to step out from his position in the processing order and to step in at any position somewhere later in the processing order. In particular, each player outside the coalition will not obtain any new predecessors, possibly only fewer. For the time being we apply this relaxation on the classical sequencing situation as introduced by Curiel et al. (1989). However, this relaxation can also applied to other types of sequencing situations. We start with proving non-emptiness of the core for the class of relaxed sequencing games where the values of the coalitions are bounded from above by the value of this coalition in a classical sequencing game if this coalition would have been connected. As this general result can be applied to the class of SoSi sequencing games, every SoSi sequencing game has a non-empty core. Moreover, we provide a polynomial time algorithm determining an optimal processing order for a coalition and the corresponding value. The algorithm considers the players of the coalition in an order that is the reverse of the initial order, and for every player the algorithm checks whether moving the player to a position later in the processing order is beneficial. This algorithm works in a greedy way in the sense that every player is moved to the position giving the highest cost savings at that moment. Moreover, every player is considered in the algorithm exactly once and every player is moved to another position in the processing order at most once. Probably the main result of this chapter is the convexity of SoSi sequencing games. In the proof of this results we use specific features of the algorithm. In particular, for determining an optimal processing order for a coalition, one can use the information of the optimal processing orders of subcoalitions. More precisely, if one wants to know an optimal processing order for a coalition $S \cup \{i\}$, then the algorithm can start from the optimal processing order found for coalition S. In particular, this helps to analyze the marginal contribution of a player i to joining coalitions S, T with $S \subseteq T$ and $i \notin T$, and thus it helps to prove the convexity of SoSi sequencing games.

The organization of this chapter is as follows. Section 6.2 recalls basic definitions on one-machine sequencing situations and formally introduces SoSi sequencing games. Providing an upper bound on the values of the coalitions in a SoSi sequencing game, in Section 6.3 it is shown that every SoSi sequencing game has a non-empty core. Section 6.4 provides a polynomial time algorithm to determine the values of the coalitions in a SoSi sequencing game. Section 6.5 identifies a number of important key features of this algorithm that are especially useful in proving the convexity of SoSi sequencing games. In Section 6.6 the proof of convexity for SoSi sequencing games is provided. Finally, in Section 6.7 we provide some directions for future research. In particular we consider another type of relaxed sequencing games, so-called Step out sequencing games. Section 6.8 consists of an appendix with proofs belonging to several lemmas and theorems.

6.2 SoSi sequencing games

This section recalls basic definitions on one-machine sequencing situations. We also introduce SoSi sequencing games and we clarify the difference with classical sequencing games.

A *one-machine sequencing situation* can be summarized by a tuple (N, σ_0, p, α), where N is the set of players, each with one job to be processed on the single machine. A processing order of the players can be described by a bijection $\sigma : N \to \{1, \ldots, |N|\}$. More specifically, $\sigma(i) = k$ means that player i is in position k. The processing order $\sigma_0 \in \Pi(N)$ specifies the initial order, where $\Pi(N)$ denotes the set of all orders on N, i.e., the set of all processing orders. The processing time $p_i > 0$ of the job of player i is the time the machine takes to process this job. The vector $p \in \mathbb{R}^N_{++}$ summarizes the processing times. Furthermore, the costs of player i of spending t time units in the system is assumed to be determined by a linear cost function $c_i : [0, \infty) \to \mathbb{R}$ given by $c_i(t) = \alpha_i t$ with $\alpha_i > 0$. The vector $\alpha \in \mathbb{R}^N_{++}$ summarizes the coefficients of the linear cost functions. It is assumed that the machine starts processing at time $t = 0$, and also that all jobs enter the system at $t = 0$.

Let $C_i(\sigma)$ be the *completion time* of the job of player i with respect to processing order σ via the associated semi-active schedule, i.e., a schedule in which there is no idle time between the jobs. Hence, the completion time of player i equals

$$C_i(\sigma) = \sum_{j \in N: \sigma(j) \leq \sigma(i)} p_j.$$

A processing order is called *optimal* if the total joint costs $\sum_{i \in N} \alpha_i C_i(\sigma)$ are minimized. In Smith (1956) it is shown that in each optimal order the players are processed in non-increasing order with respect to their *urgency* u_i defined by $u_i = \frac{\alpha_i}{p_i}$. Moreover, with g_{ij} representing the gain made by a possible neighbor switch of i and j if player i is directly in front of player j, i.e., with

$$g_{ij} = \max\{\alpha_j p_i - \alpha_i p_j, 0\},$$

the maximal total cost savings are equal to

$$\sum_{i \in N} \alpha_i C_i(\sigma_0) - \sum_{i \in N} \alpha_i C_i(\sigma^*) = \sum_{i,j \in N: \sigma_0(i) < \sigma_0(j)} g_{ij}, \tag{6.1}$$

where σ^* denotes an optimal order.

To tackle the allocation problem of the maximal cost savings in a sequencing situation (N, σ_0, p, α) one can analyze an associated coalitional game (N, v). Here N naturally corresponds to the set of players in the game and, for a coalition $S \subseteq N$, $v(S)$ reflects the maximal cost savings this coalition can make with respect to the initial order σ_0. In order to determine these maximal cost savings, assumptions must be made on the possible reorderings of coalition S with respect to the initial order σ_0.

The classical (strong) assumption is that a member of a certain coalition $S \subseteq N$ can only swap with another member of the coalition if all players between these two players, according to the initial order, are also members of S. Given an initial order σ_0 the set of admissible orders for coalition S in a classical sequencing game is denoted by $\mathcal{A}^c(\sigma_0, S)$. This leads to the definition of a classical sequencing game. However, note that the resulting set of admissible reorderings for a coalition is quite restrictive, because there may be more reorderings possible which do not hurt the interests of the players outside the coalition.

In a SoSi sequencing game the classical assumption is relaxed by additionally allowing that a member of the coalition S steps out from his position in the processing order and steps in at any position later in the processing order. Hence, a processing order σ is called *admissible* for S in a SoSi sequencing game if

(i) $P(\sigma, i) \subseteq P(\sigma_0, i)$ for all $i \in N \backslash S$,

(ii) $\sigma^{-1}(\sigma(i) + 1) \in F(\sigma_0, i)$ for all $i \in N \backslash S$ with $\sigma(i) \neq |N|$,

where $P(\sigma, i) = \{j \in N \mid \sigma(j) < \sigma(i)\}$ denotes the set of predecessors of player i with respect to processing order σ and $F(\sigma, i) = \{j \in N \mid \sigma(j) > \sigma(i)\}$ denotes the set of followers. Condition (i) ensures that no player outside S obtains any new predecessors. As a result, a player who steps out from his position in the processing order, steps in at a position later, and thus not earlier, in the processing order. Note that there are actually two types of admissible swaps. The first one is a swap of adjacent pairs in the same component of S. The second type of admissible swaps consists of the swaps where a player moves to a later component and thus swaps with players outside S. Note that from an optimality point of view it is clear that we can assume without loss of generality that a member of S who steps out, only steps in at a position directly behind another member of the coalition S. Therefore, in order to make the proofs in this chapter less complex, we also required condition (ii). This condition states that each player outside S has a direct follower who was already a follower of him with respect to σ_0. Given an initial order σ_0 the set of admissible orders for coalition S is denoted by $\mathcal{A}(\sigma_0, S)$.

Correspondingly, the *Step out - Step in (SoSi) sequencing game* (N, v) is defined by

$$v(S) = \max_{\sigma \in \mathcal{A}(\sigma_0, S)} \sum_{i \in S} \alpha_i (C_i(\sigma_0) - C_i(\sigma)),$$

for all $S \subseteq N$, i.e., the value of a coalition is equal to the maximal cost savings a coalition can achieve by means of admissible rearrangements. A processing order

$\sigma^* \in \mathcal{A}(\sigma_0, S)$ is called *optimal* for coalition S if

$$\sum_{i\in S} \alpha_i(C_i(\sigma_0) - C_i(\sigma^*)) = \max_{\sigma\in\mathcal{A}(\sigma_0,S)} \sum_{i\in S} \alpha_i(C_i(\sigma_0) - C_i(\sigma)).$$

Note that a processing order is admissible for a coalition in a classical sequencing game if there is an equality in condition (i). Therefore, given a coalition, the corresponding set of admissible orders in a SoSi sequencing game is larger than the set of admissible orders in the corresponding classical sequencing game, i.e., $\mathcal{A}(\sigma_0, S) \supseteq \mathcal{A}^c(\sigma_0, S)$. As a consequence, the value of S in a SoSi sequencing game is at least its value in the classical sequencing game. For $\sigma \in \Pi(N)$, S is called *connected* with respect to σ if for all $i, j \in S$ and $k \in N$ such that $\sigma(i) < \sigma(k) < \sigma(j)$ it holds that $k \in S$. Note that for each coalition that is connected with respect to σ_0 the set of admissible orders in the SoSi sense equals the set of admissible orders in the classical sense. This means that the value of any connected coalition is the same in the SoSi sequencing game and in the classical sequencing game. Therefore, similar to (6.1), it readily can be concluded that the value of a coalition S that is connected with respect to σ_0 is given by

$$v(S) = \sum_{i,j\in S:\sigma_0(i)<\sigma_0(j)} g_{ij}. \tag{6.2}$$

Example 6.2.1. Consider a one-machine sequencing situation with $N = \{1, 2, 3\}$. The vector of processing times is $p = (3, 2, 1)$ and the vector of coefficients corresponding to the linear cost functions is $\alpha = (4, 6, 5)$. Assume that the initial order is $\sigma_0 = (1\ 2\ 3)$. It then follows that $g_{12} = 10$, $g_{13} = 11$ and $g_{23} = 4$.

Let (N, v) be the corresponding SoSi sequencing game. Table 6.1 provides the values of all coalitions. Note that the values of the coalitions in the game (N, v) are equal to the values of the coalitions in the classical sequencing game of this one-machine sequencing situation except for the only disconnected coalition, coalition $\{1, 3\}$. For instance, for the grand coalition N it follows from (6.2) that

$$v(N) = \sum_{i,j\in N:\sigma_0(i)<\sigma_0(j)} g_{ij} = g_{12} + g_{13} + g_{23} = 25.$$

The disconnected coalition $\{1, 3\}$ cannot save costs in the classical sequencing game because there exists no admissible order other than the initial order. However, in the SoSi sequencing game the set of admissible orders consists of two elements:

S	$\{1\}$	$\{2\}$	$\{3\}$	$\{1,2\}$	$\{1,3\}$	$\{2,3\}$	N
$v(S)$	0	0	0	10	3	4	25

Table 6.1: The SoSi sequencing game of Example 6.2.1

$\mathcal{A}(\sigma_0, \{1,3\}) = \{(1\ 2\ 3), (2\ 3\ 1)\}$.[1]

These processing orders are illustrated in Figure 6.1. Hence, the value of coalition $\{1,3\}$ is given by[2]

$$v(\{1,3\}) = \max\left\{0, \sum_{i\in\{1,3\}} \alpha_i(C_i((1\ 2\ 3)) - C_i((2\ 3\ 1)))\right\} = \max\{0, -12+15\} = 3.\triangle$$

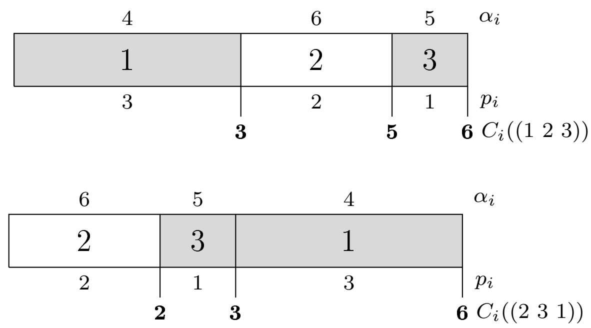


Figure 6.1: The two admissible orders for coalition $\{1,3\}$ in Example 6.2.1

6.3 Non-emptiness of the core

In this section we prove non-emptiness of the core for the class of relaxed sequencing games where the values of the coalitions are bounded from above by the gains made

[1] Processing order (2 3 1) means that player 2 is in the first position, player 3 in the second position and player 1 in the last position.

[2] Note that $v(\{1,3\}) \neq v(\{1\}) + v(\{3\})$ and thus SoSi sequencing games are not σ-component additive games. Therefore, proving non-emptiness of the core of SoSi sequencing games using standard techniques via σ-component additive games (cf. Le Breton, Owen, and Weber (1992)) does not work.

by all possible neighbor switches. Next, we show that the class of SoSi sequencing games belongs to this class. As a consequence, every SoSi sequencing game has a non-empty core.

Given an initial order σ_0, assume that the set of admissible orders $\mathcal{A}^r(\sigma_0, S)$ for coalition S satisfies

$$\mathcal{A}^c(\sigma_0, S) \subseteq \mathcal{A}^r(\sigma_0, S),$$

for all $S \subseteq N$. Then, the corresponding *relaxed sequencing game* (N, v) is defined by

$$v(S) = \max_{\sigma \in \mathcal{A}^r(\sigma_0, S)} \sum_{i \in S} \alpha_i (C_i(\sigma_0) - C_i(\sigma)),$$

for all $S \subseteq N$. Note that a SoSi sequencing game indeed is a relaxed sequencing game in the above sense. The following theorem proves that every connected marginal vector is a core element for any relaxed sequencing game (N, v) where the value of a coalition is bounded from above by the gains made by all possible neighbor switches, i.e.,

$$v(S) \leq \sum_{i,j \in S: \sigma_0(i) < \sigma_0(j)} g_{ij},$$

for all $S \subseteq N$. Notice that if a relaxed sequencing game (N, v) satisfies this condition, then we have

$$v(S) = \sum_{i,j \in S: \sigma_0(i) < \sigma_0(j)} g_{ij},$$

for every connected coalition $S \subseteq N$.

Theorem 6.3.1. *Let (N, σ_0, p, α) be a one-machine sequencing situation, let $\pi \in \Pi(N)$ be such that the set $\{j \in N \mid \pi(j) \leq \pi(i)\}$ is connected with respect to σ_0 for all $i \in N$ and let (N, v) be a corresponding relaxed sequencing game such that*

$$v(S) \leq \sum_{i,j \in S: \sigma_0(i) < \sigma_0(j)} g_{ij}, \tag{6.3}$$

for all $S \subseteq N$. Then, $m^\pi(v) \in C(v)$.

Proof. Let $i \in N$ and denote $T_i = \{j \in N \mid \pi(j) < \pi(i)\}$. Because π is connected it follows that either $T_i \subseteq P(\sigma_0, i)$ or $T_i \subseteq F(\sigma_0, i)$ and therefore we have

$$m_i^\pi(v) \quad = \quad v(T_i \cup \{i\}) - v(T_i)$$

$$= \sum_{j,k \in T_i \cup \{i\} : \sigma_0(j) < \sigma_0(k)} g_{jk} - \sum_{j,k \in T_i : \sigma_0(j) < \sigma_0(k)} g_{jk}$$
$$= \begin{cases} \sum_{j \in T_i} g_{ji} & \text{if } T_i \subseteq P(\sigma_0, i) \\ \sum_{j \in T_i} g_{ij} & \text{if } T_i \subseteq F(\sigma_0, i), \end{cases}$$

where the second equality follows from the fact that the coalitions T_i and $T_i \cup \{i\}$ are connected with respect to σ_0.

Then, for $S \subseteq N$ it holds that

$$\begin{aligned}
\sum_{i \in S} m_i^\pi(v) &= \sum_{\substack{i \in S: \\ T_i \subseteq P(\sigma_0,i)}} \sum_{j \in T_i} g_{ji} + \sum_{\substack{i \in S: \\ T_i \subseteq F(\sigma_0,i)}} \sum_{j \in T_i} g_{ij} \\
&\geq \sum_{\substack{i \in S: \\ T_i \subseteq P(\sigma_0,i)}} \sum_{j \in T_i \cap S} g_{ji} + \sum_{\substack{i \in S: \\ T_i \subseteq F(\sigma_0,i)}} \sum_{j \in T_i \cap S} g_{ij} \\
&= \sum_{i \in S} \sum_{j \in T_i \cap S \cap P(\sigma_0,i)} g_{ji} + \sum_{i \in S} \sum_{j \in T_i \cap S \cap F(\sigma_0,i)} g_{ij} \\
&= \sum_{j \in S} \sum_{\substack{i \in S \cap F(\sigma_0,j): \\ i \notin T_j}} g_{ji} + \sum_{i \in S} \sum_{j \in T_i \cap S \cap F(\sigma_0,i)} g_{ij} \\
&= \sum_{i \in S} \sum_{\substack{j \in S \cap F(\sigma_0,i): \\ j \notin T_i}} g_{ij} + \sum_{i \in S} \sum_{j \in T_i \cap S \cap F(\sigma_0,i)} g_{ij} \\
&= \sum_{i \in S} \sum_{j \in S \cap F(\sigma_0,i)} g_{ij} \\
&= \sum_{i,j \in S : \sigma_0(i) < \sigma_0(j)} g_{ij} \\
&\geq v(S).
\end{aligned}$$

In the above derivation, the equations follow from

- interchanging the summations of the first term and the fact that $i \in T_j$ if and only if $j \notin T_i$ for all $i, j \in N$ with $i \neq j$ (third equality),

- interchanging the indices of the summations of the first term (fourth equality).

Note that if $S = N$ the inequalities become equalities. This proves that $m^\pi(v) \in C(v)$. □

To show that every SoSi sequencing game has a non-empty core, it remains to be proven that SoSi sequencing games satisfy condition (6.3) in Theorem 6.3.1. We start with some basic definitions and notations regarding components, modified components and urgency respecting orders. For $S \in 2^N \backslash \{\emptyset\}$, $\sigma \in \Pi(N)$ and $s, t \in N$ with $\sigma(s) < \sigma(t)$, define

$$\begin{aligned} S^\sigma(s,t) &= \{i \in S \mid \sigma(s) < \sigma(i) < \sigma(t)\}, \\ \bar{S}^\sigma(s,t) &= \{i \in N\backslash S \mid \sigma(s) < \sigma(i) < \sigma(t)\}, \\ S^\sigma[s,t] &= \{i \in S \mid \sigma(s) \leq \sigma(i) \leq \sigma(t)\}, \\ \bar{S}^\sigma[s,t] &= \{i \in N\backslash S \mid \sigma(s) \leq \sigma(i) \leq \sigma(t)\}. \end{aligned}$$

The sets of players $S^\sigma[s,t)$, $\bar{S}^\sigma[s,t)$, $S^\sigma(s,t]$ and $\bar{S}^\sigma(s,t]$ are defined in a similar way.

A connected coalition $U \subseteq S$ with respect to σ is called a *component* of S with respect to σ if $U \subseteq U' \subseteq S$ and U' connected with respect to σ implies that $U' = U$. Let $h(\sigma, S) \geq 1$ denote the number of components of S with respect to σ. The partition of S into components with respect to σ is denoted by

$$S\backslash\sigma = \left\{S_1^\sigma, S_2^\sigma, \ldots, S_{h(\sigma,S)}^\sigma\right\}.$$

with for each $k \in \{1, \ldots, h(\sigma, S) - 1\}$, $i \in S_k^\sigma$ and $j \in S_{k+1}^\sigma$ we have $\sigma(i) < \sigma(j)$. In the same way processing order σ divides $N\backslash S$ into subgroups. For this, define

$$\begin{aligned} \bar{S}_0^\sigma &= \{i \in N\backslash S \mid \sigma(i) < \sigma(j) \text{ for all } j \in S_1^\sigma\}, \\ \bar{S}_{h(\sigma,S)}^\sigma &= \{i \in N\backslash S \mid \sigma(i) > \sigma(j) \text{ for all } j \in S_{h(\sigma,S)}^\sigma\}, \\ \bar{S}_k^\sigma &= \{i \in N\backslash S \mid \sigma(j) < \sigma(i) < \sigma(l) \text{ for all } j \in S_k^\sigma, \text{ for all } l \in S_{k+1}^\sigma\}, \end{aligned}$$

for all $k \in \{1, \ldots, h(\sigma, S) - 1\}$. Notice that $\bar{S}_0^\sigma$ and $\bar{S}_{h(\sigma,S)}^\sigma$ might be empty sets, but $\bar{S}_k^\sigma \neq \emptyset$ for all $k \in \{1, \ldots, h(\sigma, S) - 1\}$. See Figure 6.2 for an illustration of the subdivision of S and $N\backslash S$ into subgroups by means of processing order σ.

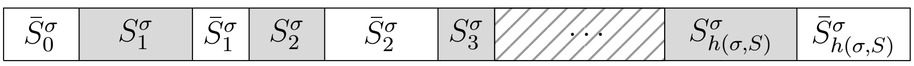


Figure 6.2: Partition of the players in S and $N\backslash S$ with respect to an order σ

Note that for given $S \subseteq N$ it is possible that a processing order $\sigma \in \mathcal{A}(\sigma_0, S)$ contains less components than σ_0, because all players of a certain component with respect to S may step out from this component and join other components. For $\sigma \in \mathcal{A}(\sigma_0, S)$ with $\sigma_0 \in \Pi(N)$, define *modified components* $S_1^{\sigma_0,\sigma}, \ldots, S_{h(\sigma_0,S)}^{\sigma_0,\sigma}$ by

$$S_k^{\sigma_0,\sigma} = \left\{ i \in S \mid \sigma(j) < \sigma(i) < \sigma(l) \text{ for all } j \in \overline{S}_{k-1}^{\sigma_0}, \text{ for all } l \in \overline{S}_k^{\sigma_0} \right\},$$

for all $k \in \{1, \ldots, h(\sigma_0, S)\}$. Hence, $S_k^{\sigma_0,\sigma}$ consists of the group of players that are positioned in processing order σ in between the subgroups $\overline{S}_{k-1}^{\sigma_0}$ and $\overline{S}_k^{\sigma_0}$.

Note that $S_k^{\sigma_0,\sigma}$ might be empty for some k while

$$\bigcup_{k=1}^{h(\sigma_0,S)} S_k^{\sigma_0,\sigma} = S.$$

Moreover, recall that a player is not allowed to move to an earlier component (condition (i) of admissibility) but he is allowed to move to any position later in the processing order and thus we have

$$\bigcup_{k=1}^{l} S_k^{\sigma_0,\sigma} \subseteq \bigcup_{k=1}^{l} S_k^{\sigma_0},$$

for all $l \in \{1, \ldots, h(\sigma_0, S)\}$. Furthermore, denote the index of the corresponding modified component of player $i \in S$ in processing order σ with respect to initial processing order σ_0 by $c(i, S, \sigma)$, where

$$c(i, S, \sigma) = k \text{ if and only if } i \in S_k^{\sigma_0,\sigma}.$$

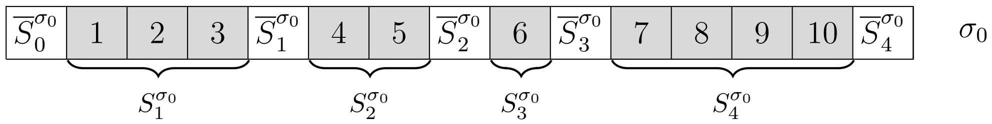
Since the component index of player $i \in S$ with respect to σ can only be increased (due to condition (i) of admissibility), we have

$$c(i, S, \sigma) \geq c(i, S, \sigma_0).$$

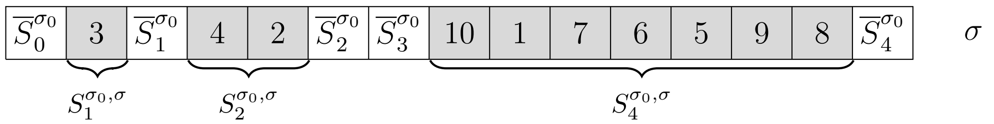
An illustration of the definitions of components, modified components and the index $c(i, S, \sigma)$ can be found in the following example.

Example 6.3.1. Consider a one-machine sequencing situation (N, σ_0, p, α) with $S \subseteq N$ such that $S = \{1, 2, \ldots, 10\}$. In Figure 6.3(a) an illustration can be found of initial processing order σ_0 and the partition of S into components. Next, consider processing order σ as illustrated in Figure 6.3(b) that is admissible for S. Note that σ contains less components than σ_0. Figure 6.3(b) also illustrates the definition of modified components. Note that there is one modified component that is empty, namely

$S_3^{\sigma_0,\sigma}$. Since player 3 belongs to the first modified component, we have $c(3,S,\sigma) = 1$. Moreover, since player 3 is the only player who belongs to the first modified component, we have $S_1^{\sigma_0,\sigma} = \{3\}$. Similarly, we have $c(4,S,\sigma) = c(2,S,\sigma) = 2$, and $c(i,S,\sigma) = 4$, for all $i \in S\backslash\{2,3,4\}$. $\triangle$



(a) Illustration of the components of S with respect to σ_0



(b) Illustration of the modified components of S with respect to σ and initial order σ_0

Figure 6.3: Illustration of components and modified components

A processing order $\sigma \in \mathcal{A}(\sigma_0, S)$ is called *urgency respecting* with respect to S if

(i) (σ is *componentwise optimal*) for all $i, j \in S$ with $c(i,S,\sigma) = c(j,S,\sigma)$:

$$\sigma(i) < \sigma(j) \Rightarrow u_i \geq u_j.$$

(ii) (σ satisfies *partial tiebreaking*) for all $i, j \in S$ with $c(i,S,\sigma_0) = c(j,S,\sigma_0)$:

$$u_i = u_j, \sigma_0(i) < \sigma_0(j) \Rightarrow \sigma(i) < \sigma(j).$$

Componentwise optimality states that the players within a component of S are in non-increasing order with respect to their urgency. The partial tiebreaking condition ensures that if there are two players with the same urgency in the same component of S with respect to σ_0, then the player who was first in σ_0 is earlier in processing order σ. Note that the partial tiebreaking condition does not imply anything about the relative order of two players with the same urgency who are in the same component of S with respect to σ but who were in different components of S with respect to σ_0. Therefore, an urgency respecting order does not need to be unique. Clearly, there always exists an optimal order for S that is urgency respecting. The partial tiebreaking condition is required to make sure that there is a well-defined procedure of consecutive movements to go from the initial order σ_0 to the urgency respecting order σ, as is explained later in this section.

Next, define $\sigma_0^S \in \mathcal{A}(\sigma_0, S)$ to be the unique urgency respecting processing order such that for all $i \in S$

$$c(i, S, \sigma_0^S) = c(i, S, \sigma_0),$$

Hence, all players in S stay in their component as they are in the initial order σ_0, i.e., the partition of S into components stays the same.

For a processing order $\sigma \in \Pi(N)$ and $i, j \in N$, with $\sigma(i) < \sigma(j)$, we define $[i, j]\sigma$ to be the processing order that is obtained from σ by moving player i to the position directly behind player j, i.e.,

$$([i,j]\sigma)(s) = \begin{cases} \sigma(s) & \text{if } s \notin N^\sigma[i,j] \\ \sigma(s) - 1 & \text{if } s \in N^\sigma(i,j] \\ \sigma(j) & \text{if } s = i, \end{cases}$$

for every $s \in N$ (see Figure 6.4).

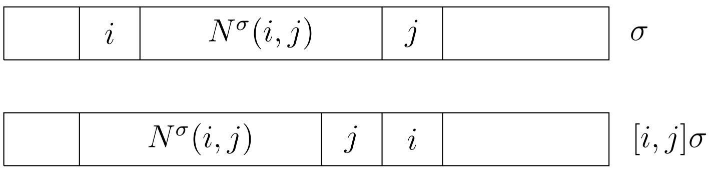


Figure 6.4: Illustration of $[i, j]\sigma$

In a SoSi sequencing game there are two types of operations allowed for a coalition S given the initial order σ_0. A *type I operation* is a swap of adjacent pairs in the same component of S. A *type II operation* is a move from a player in S to the position directly behind another player of S within one of the subsequent components.

An urgency respecting processing order $\sigma \in \mathcal{A}(\sigma_0, S)$ can always be obtained with the operations described above from the initial order σ_0 via the processing order σ_0^S. In order to obtain the processing order σ_0^S from the initial order σ_0 only type I operations are performed, while for obtaining the processing order σ from σ_0^S only type II operations need to be performed. These type II operations can be chosen in such a way that the moved player is already on the correct urgency respecting position in his new component, as demonstrated below.

Let $\sigma \in \mathcal{A}(\sigma_0, S)$ be urgency respecting. Define $R(\sigma)$ as the set of players who switch component, i.e.,

$$R(\sigma) = \{i \in S \mid c(i, S, \sigma) > c(i, S, \sigma_0)\}. \tag{6.4}$$

If $\sigma \neq \sigma_0^S$, then $|R(\sigma)| \geq 1$. Next, define $r_1(\sigma) \in R(\sigma)$ such that

$$\sigma_0^S(r_1(\sigma)) \geq \sigma_0^S(r),$$

for all $r \in R(\sigma)$ and $m_1(\sigma) \in S$ with $m_1(\sigma) \notin R(\sigma)$, such that

$$\sigma(m_1(\sigma)) < \sigma(r_1(\sigma))$$

and

$$\sigma(m_1(\sigma)) \geq \sigma(j),$$

for all $j \in S$ with $j \notin R(\sigma)$ and $\sigma(j) < \sigma(r_1(\sigma))$. Note that $m_1(\sigma)$ is well-defined because $\sigma_0^S(m_1(\sigma)) > \sigma_0^S(r_1(\sigma))$ due to condition (ii) of admissibility. Defining

$$\tau^{\sigma,S,1} = [r_1(\sigma), m_1(\sigma)]\sigma_0^S,$$

$\tau^{\sigma,S,1}$ is an urgency respecting and admissible order for S. Intuitively, player $r_1(\sigma)$ is the first player who is moved in order to go from the order σ_0^S to the order σ. Moreover, $m_1(\sigma)$ is the player where player $r_1(\sigma)$ must be positioned behind.

For $k \in \{2, \ldots, |R(\sigma)|\}$, recursively, define $r_k(\sigma) \in R(\sigma)\backslash\{r_1(\sigma), \ldots, r_{k-1}(\sigma)\}$ such that

$$\sigma_0^S(r_k(\sigma)) \geq \sigma_0^S(r),$$

for all $r \in R(\sigma)\backslash\{r_1(\sigma), \ldots, r_{k-1}(\sigma)\}$. Moreover, define $m_k(\sigma) \in S$ with $m_k(\sigma) \notin R(\sigma)\backslash\{r_1(\sigma), \ldots, r_{k-1}(\sigma)\}$ such that

$$\sigma(m_k(\sigma)) < \sigma(r_k(\sigma))$$

and

$$\sigma(m_k(\sigma)) \geq \sigma(j),$$

for all $j \in S$ with $j \notin R(\sigma)\backslash\{r_1(\sigma), \ldots, r_{k-1}(\sigma)\}$ and $\sigma(j) < \sigma(r_k(\sigma))$, and, finally, set

$$\tau^{\sigma,S,k} = [r_k(\sigma), m_k(\sigma)]\tau^{\sigma,S,k-1}. \tag{6.5}$$

Note that $m_k(\sigma)$ is well-defined because $\tau^{\sigma,S,k-1}(m_k(\sigma)) > \tau^{\sigma,S,k-1}(r_k(\sigma))$ due to condition (ii) of admissibility. Moreover, $\tau^{\sigma,S,k}$ is an admissible urgency respecting order for S (because σ is urgency respecting) and

$$\tau^{\sigma,S,|R(\sigma)|} = \sigma.$$

For notational convenience we define $\tau^{\sigma,S,0}$ to be σ_0^S.

An illustration of the procedure described above can be found in the following example.

Example 6.3.2. Consider a one-machine sequencing situation (N, σ_0, p, α) with $S \subseteq N$ such that $S = \{1, 2, \ldots, 10\}$, and $\sigma_0(k) < \sigma_0(l)$ if and only if $k < l$, for $k, l \in S$. Moreover, assume that the components of coalition S with respect to σ_0 are given by

$$S_1^{\sigma_0} = \{1, 2, 3\}, S_2^{\sigma_0} = \{4, 5\}, S_3^{\sigma_0} = \{6\} \text{ and } S_4^{\sigma_0} = \{7, 8, 9, 10\},$$

with the urgencies of the players in S specified in Table 6.2.

Player i	1	2	3	4	5	6	7	8	9	10
u_i	$\frac{4}{9}$	$\frac{4}{9}$	4	$1\frac{1}{2}$	$\frac{1}{3}$	$\frac{3}{8}$	$\frac{4}{9}$	$\frac{1}{5}$	$\frac{1}{4}$	1

Table 6.2: Urgencies of the players in coalition S in Example 6.3.2

In Figure 6.5 the orders σ_0 and σ_0^S (the first two processing orders) are illustrated, together with an urgency respecting order $\sigma \in \mathcal{A}(\sigma_0, S)$ (the last processing order) for which

$$S_1^{\sigma_0,\sigma} = \{3\}, S_2^{\sigma_0,\sigma} = \{1, 4\}, S_3^{\sigma_0,\sigma} = \emptyset \text{ and } S_4^{\sigma_0,\sigma} = \{2, 5, 6, 7, 8, 9, 10\}.$$

Hence, from (6.4) it follows that

$$R(\sigma) = \{1, 2, 5, 6\},$$

and

$$r_1(\sigma) = 6, r_2(\sigma) = 5, r_3(\sigma) = 2 \text{ and } r_4(\sigma) = 1.$$

Moreover,

$$m_1(\sigma) = 7, m_2(\sigma) = 6, m_3(\sigma) = 7 \text{ and } m_4(\sigma) = 4.$$

$\bar{S}_0^{\sigma_0}$	1	2	3	$\bar{S}_1^{\sigma_0}$	4	5	$\bar{S}_2^{\sigma_0}$	6	$\bar{S}_3^{\sigma_0}$	7	8	9	10	$\bar{S}_4^{\sigma_0}$	σ_0
$\bar{S}_0^{\sigma_0}$	3	1	2	$\bar{S}_1^{\sigma_0}$	4	5	$\bar{S}_2^{\sigma_0}$	6	$\bar{S}_3^{\sigma_0}$	10	7	9	8	$\bar{S}_4^{\sigma_0}$	σ_0^S
$\bar{S}_0^{\sigma_0}$	3	1	2	$\bar{S}_1^{\sigma_0}$	4	5	$\bar{S}_2^{\sigma_0}$	$\bar{S}_3^{\sigma_0}$	10	7	6	9	8	$\bar{S}_4^{\sigma_0}$	$\tau^{\sigma,S,1}$
$\bar{S}_0^{\sigma_0}$	3	1	2	$\bar{S}_1^{\sigma_0}$	4	$\bar{S}_2^{\sigma_0}$	$\bar{S}_3^{\sigma_0}$	10	7	6	5	9	8	$\bar{S}_4^{\sigma_0}$	$\tau^{\sigma,S,2}$
$\bar{S}_0^{\sigma_0}$	3	1	$\bar{S}_1^{\sigma_0}$	4	$\bar{S}_2^{\sigma_0}$	$\bar{S}_3^{\sigma_0}$	10	7	2	6	5	9	8	$\bar{S}_4^{\sigma_0}$	$\tau^{\sigma,S,3}$
$\bar{S}_0^{\sigma_0}$	3	$\bar{S}_1^{\sigma_0}$	4	1	$\bar{S}_2^{\sigma_0}$	$\bar{S}_3^{\sigma_0}$	10	7	2	6	5	9	8	$\bar{S}_4^{\sigma_0}$	$\tau^{\sigma,S,4} = \sigma$

Figure 6.5: The processing orders corresponding to Example 6.3.2

The orders $\tau^{\sigma,S,1}, \tau^{\sigma,S,2}, \tau^{\sigma,S,3}$ and $\tau^{\sigma,S,4}$ are depicted in Figure 6.5 as well. Since $\tau^{\sigma,S,4} = \sigma$ we find that

$$\sigma = [1,4][2,7][5,6][6,7]\sigma_0^S. \qquad \triangle$$

Now we are ready to prove non-emptiness of the core of SoSi sequencing games.

Theorem 6.3.2. *Every SoSi sequencing game has a non-empty core.* [3]

Proof. Clearly, it suffices to show that SoSi sequencing games satisfy condition (6.3) in Theorem 6.3.1. Let (N, σ_0, p, α) be a one-machine sequencing situation and let (N, v) be the corresponding SoSi sequencing game. Next, let $S \subseteq N$ and let σ^* be an urgency respecting optimal order for coalition S. Because of (6.2) we can assume without loss of generality that S is not connected. We define a special processing order θ_0 such that the coalition S is connected with respect to θ_0. Thereafter, all type I and type II operations that are used to obtain σ^* from σ_0 via σ_0^S are performed on the processing order θ_0.

Denote by Q the set of players outside S and positioned between two components

[3] This result can also be obtained using Theorem 5.1 in Slikker (2006b). However, our proof is more context specific as Theorem 5.1 in Slikker (2006b) holds for a more general class of relaxed sequencing games.

of S according to σ_0, i.e.,

$$Q = \bigcup_{k=1}^{h(\sigma_0,S)-1} \bar{S}_k^{\sigma_0}.$$

Consider a one-machine sequencing situation (N, θ_0, p, α) with N, p and α as defined above and θ_0 an initial order such that

1. $\theta_0(i) = \sigma_0(i)$ for all $i \in N \backslash (S \cup Q)$,

2. $\sigma_0(i) < \sigma_0(j) \Rightarrow \theta_0(i) < \theta_0(j)$ for all $i, j \in S$,

3. $\min\{\theta_0(i) \mid i \in S\} = \min\{\sigma_0(i) \mid i \in S\}$,

4. $\max\{\theta_0(i) \mid i \in S\} = \min\{\sigma_0(i) \mid i \in S\} + |S| - 1$.

Hence, the order θ_0 is derived from σ_0 in such a way that

- the position of the players outside $S \cup Q$ has not been changed,

- the relative order between the players in S remains the same,

- the players in S are moved forward as far as possible.

As a consequence, all players in Q are positioned in an arbitrary way between $S_{h(\sigma_0,S)}^{\sigma_0}$ and $\bar{S}_{h(\sigma_0,S)}^{\sigma_0}$ according to θ_0. In particular, S is a connected coalition with respect to θ_0 (cf. Figure 6.6).

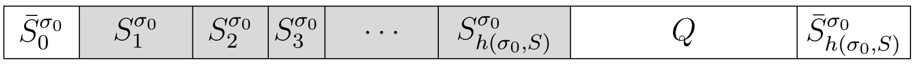


Figure 6.6: Illustration of the order θ_0

Denote processing order $\hat{\theta}_0 \in \mathcal{A}(\theta_0, S)$ such that

1. $\hat{\theta}_0(i) = \theta_0(i)$ for all $i \notin S$,

2. $\sigma_0^S(i) < \sigma_0^S(j) \Rightarrow \hat{\theta}_0(i) < \hat{\theta}_0(j)$ for all $i, j \in S$.

Hence, the order $\hat{\theta}_0$ is obtained from θ_0 in such a way that

- the position of the players outside S has not been changed,

- such that the relative order between the players in S is the same as their relative order in σ_0^S.

Obviously $\hat{\theta}_0$ can be obtained from θ_0 in exactly the same way as σ_0^S from σ_0 using the same operations of type I, conducted in the same order. Note that each operation results in the same cost difference. Therefore,

$$\sum_{i \in S} \alpha_i (C_i(\sigma_0) - C_i(\sigma_0^S)) = \sum_{i \in S} \alpha_i (C_i(\theta_0) - C_i(\hat{\theta}_0)). \tag{6.6}$$

Observe that if the order σ_0 is already urgency respecting, i.e., if $\sigma_0^S = \sigma_0$, then also $\hat{\theta}_0 = \theta_0$ and thus (6.6) still holds.

Next, let $R = R(\sigma^*)$, $r_k = r_k(\sigma^*)$ and $m_k = m_k(\sigma^*)$ for all $k \in \{1, \ldots, |R|\}$ as defined in (6.4) such that

$$\sigma^* = [r_{|R|}, m_{|R|}] \ldots [r_2, m_2][r_1, m_1]\sigma_0^S.$$

Define the processing order θ^* by

$$\theta^* = [r_{|R|}, m_{|R|}] \ldots [r_2, m_2][r_1, m_1]\hat{\theta}_0.$$

Remember, if $|R| = 0$, then $\sigma^* = \sigma_0^S$ and thus also $\theta^* = \hat{\theta}_0$. The processing order θ^* is obtained from $\hat{\theta}_0$ in the same way as σ^* is obtained from σ_0^S using the same type II operations, conducted in the same order. Note that these operations are indeed valid due to the definition of $\hat{\theta}_0$. Obviously, θ^* is an admissible order for S in the sequencing situation (N, θ_0, p, α). Abbreviate $\tau^{\sigma^*, S, k}$ for $k \in \{0, 1, \ldots, |R|\}$ as defined in (6.5) by $\hat{\sigma}_k$. Moreover, set

$$\hat{\theta}_k = [r_k, m_k] \ldots [r_2, m_2][r_1, m_1]\hat{\theta}_0,$$

for all $k \in \{0, 1, \ldots, |R|\}$. Notice that $\hat{\theta}_{|R|} = \theta^*$.

Let $k \in \{1, \ldots, |R|\}$ and consider the operation $[r_k, m_k]$ performed on $\hat{\sigma}_{k-1}$ and $\hat{\theta}_{k-1}$. Then,

$$\begin{aligned} \sum_{i \in S} \alpha_i (C_i(\hat{\sigma}_{k-1}) - C_i(\hat{\sigma}_k)) &= \left(\sum_{i \in S^{\hat{\sigma}_{k-1}}(r_k, m_k]} \alpha_i \right) p_{r_k} - \alpha_{r_k} \left(\sum_{i \in N^{\hat{\sigma}_{k-1}}(r_k, m_k]} p_i \right) \\ &\leq \left(\sum_{i \in S^{\hat{\sigma}_{k-1}}(r_k, m_k]} \alpha_i \right) p_{r_k} - \alpha_{r_k} \left(\sum_{i \in S^{\hat{\sigma}_{k-1}}(r_k, m_k]} p_i \right) \end{aligned}$$

$$
\begin{aligned}
&= \left(\sum_{i \in S^{\hat{\theta}_{k-1}}(r_k, m_k]} \alpha_i \right) p_{r_k} - \alpha_{r_k} \left(\sum_{i \in S^{\hat{\theta}_{k-1}}(r_k, m_k]} p_i \right) \\
&= \sum_{i \in S} \alpha_i (C_i(\hat{\theta}_{k-1}) - C_i(\hat{\theta}_k)).
\end{aligned}
$$

Hence,

$$
\begin{aligned}
v(S) &= \sum_{i \in S} \alpha_i (C_i(\sigma_0) - C_i(\sigma^*)) \\
&= \sum_{i \in S} \alpha_i (C_i(\sigma_0) - C_i(\sigma_0^S)) + \sum_{k=1}^{|R|} \sum_{i \in S} \alpha_i (C_i(\hat{\sigma}_{k-1}) - C_i(\hat{\sigma}_k)) \\
&\leq \sum_{i \in S} \alpha_i (C_i(\theta_0) - C_i(\hat{\theta}_0)) + \sum_{k=1}^{|R|} \sum_{i \in S} \alpha_i (C_i(\hat{\theta}_{k-1}) - C_i(\hat{\theta}_k)) \\
&= \sum_{i \in S} \alpha_i (C_i(\theta_0) - C_i(\theta^*)) \\
&\leq \max_{\theta \in \mathcal{A}(\theta_0, S)} \sum_{i \in S} \alpha_i (C_i(\theta_0) - C_i(\theta)) \\
&= \sum_{i,j \in S: \theta_0(i) < \theta_0(j)} g_{ij} \\
&= \sum_{i,j \in S: \sigma_0(i) < \sigma_0(j)} g_{ij},
\end{aligned}
$$

also if $|R| = 0$. In the above derivation, the last two equalities follow from

- the fact that coalition S is connected with respect to θ_0,
- the fact that the relative order of the players in S with respect to θ_0 is the same as the relative order of the players in S with respect to σ_0. □

6.4 A polynomial time algorithm for each coalition

This section provides a polynomial time algorithm determining an optimal order for every possible coalition and, consequently, the values of the coalitions. First, some new notions as composed costs per time unit, composed processing times and composed urgencies are introduced. Using these notions we can exclude some admissible orders from being optimal.

The *composed costs per time unit* α_S and the *composed processing time* p_S for a coalition $S \in 2^N$ are defined by

$$\alpha_S = \sum_{i \in S} \alpha_i$$

and

$$p_S = \sum_{i \in S} p_i,$$

respectively. Consequently, the *composed urgency* u_S of a non-empty coalition $S \in 2^N \backslash \{\emptyset\}$ is defined by

$$u_S = \frac{\alpha_S}{p_S}.$$

Next we explain how the concept of composed urgency helps to decide which of two related processing orders is less costly for a certain coalition. For the moment we do not worry about admissibility of these orders. Consider a non-empty coalition S and a processing order $\sigma \in \Pi(N)$. First take $i \in S$ and $j \in N$ such that $\sigma(i) < \sigma(j)$. Let $\hat{\sigma} \in \Pi(N)$ be the processing order obtained from σ by moving player i to the position directly behind player j. Then the difference between the costs for coalition S with respect to the processing order σ and the processing order $\hat{\sigma}$ can be calculated as follows:

$$\begin{aligned} & \sum_{s \in S} \alpha_s C_s(\sigma) - \sum_{s \in S} \alpha_s C_s(\hat{\sigma}) \\ = & \sum_{s \in S^\sigma(i,j]} \alpha_s p_i - \alpha_i \sum_{s \in N^\sigma(i,j]} p_s \\ = & \ \alpha_{S^\sigma(i,j]} p_i - \alpha_i p_{N^\sigma(i,j]}. \end{aligned}$$

Similarly, take $i \in S$ and $j \in N$ such that $\sigma(i) > \sigma(j)$ and let $\tilde{\sigma} \in \Pi(N)$ be the processing order obtained from σ by moving player i to the position directly in front of player j. Then the difference between the costs for coalition S with respect to processing order σ and processing order $\tilde{\sigma}$ is equal to

$$\alpha_i p_{N^\sigma[j,i)} - \alpha_{S^\sigma[j,i)} p_i.$$

Hence, with $\delta_i^\sigma(j, S)$ representing the cost difference for coalition S made by moving player i directly behind player j in case $\sigma(i) < \sigma(j)$ and the cost difference made by moving player i directly in front of player j in case $\sigma(i) > \sigma(j)$, we have

$$\delta_i^\sigma(j,S) = \begin{cases} \alpha_{S^\sigma(i,j]}p_i - \alpha_i p_{N^\sigma(i,j]} & \text{if } \sigma(i) < \sigma(j) \\ \alpha_i p_{N^\sigma[j,i)} - \alpha_{S^\sigma[j,i)}p_i & \text{if } \sigma(i) > \sigma(j), \end{cases} \tag{6.7}$$

for all $i \in S$ and $j \in N$.

The above cost differences have an additive structure in the following sense. Take $i \in S$ and $j,k \in N$ such that $\sigma(i) < \sigma(j) < \sigma(k)$. The cost difference $\delta_i^\sigma(k,S)$ can be split up in two parts by first moving player i to the position directly behind player j and thereafter moving player i to the position directly behind player k. Hence, we can write

$$\delta_i^\sigma(k,S) = \delta_i^\sigma(j,S) + \delta_i^{\hat{\sigma}}(k,S),$$

where $\hat{\sigma} \in \Pi(N)$ is the processing order obtained from σ by moving player i to the position directly behind player j.

Using the above notation of a cost difference it is easily checked whether having a certain player $i \in S$ in a later position (behind player j) in the processing order is beneficial for S or not because

$$\begin{aligned} \delta_i^\sigma(j,S) > 0 \quad &\Leftrightarrow \quad \frac{\alpha_i}{p_i} < \frac{\alpha_{S^\sigma(i,j]}}{p_{N^\sigma(i,j]}}, \\ \delta_i^\sigma(j,S) = 0 \quad &\Leftrightarrow \quad \frac{\alpha_i}{p_i} = \frac{\alpha_{S^\sigma(i,j]}}{p_{N^\sigma(i,j]}}, \\ \delta_i^\sigma(j,S) < 0 \quad &\Leftrightarrow \quad \frac{\alpha_i}{p_i} > \frac{\alpha_{S^\sigma(i,j]}}{p_{N^\sigma(i,j]}}, \end{aligned}$$

with $\sigma(i) < \sigma(j)$. Similarly, it is also easily checked whether having a certain player $i \in S$ on a position earlier (in front of player j) in the processing order is beneficial for S or not because

$$\begin{aligned} \delta_i^\sigma(j,S) > 0 \quad &\Leftrightarrow \quad \frac{\alpha_i}{p_i} > \frac{\alpha_{S^\sigma[j,i)}}{p_{N^\sigma[j,i)}}, \\ \delta_i^\sigma(j,S) = 0 \quad &\Leftrightarrow \quad \frac{\alpha_i}{p_i} = \frac{\alpha_{S^\sigma[j,i)}}{p_{N^\sigma[j,i)}}, \\ \delta_i^\sigma(j,S) < 0 \quad &\Leftrightarrow \quad \frac{\alpha_i}{p_i} < \frac{\alpha_{S^\sigma[j,i)}}{p_{N^\sigma[j,i)}}, \end{aligned}$$

with $\sigma(i) > \sigma(j)$.

Using these criteria one can exclude some admissible orders from being optimal, as illustrated in the following lemma. To be specific, Lemma 6.4.1 states that if it is admissible that two players switch position, then the player with the highest urgency should be positioned first. As a consequence, if there is a player $t \in S$ positioned according to $\sigma \in \mathcal{A}(\sigma_0, S)$ behind a player $s \in S$ with lower urgency and it is admissible that these players swap their position then σ cannot be an optimal order, i.e., σ can be improved.

Lemma 6.4.1. *Let (N, σ_0, p, α) be a one-machine sequencing situation, let $S \in 2^N \backslash \{\emptyset\}$ and let $\sigma \in \mathcal{A}(\sigma_0, S)$ be an optimal order for S. Let $s, t \in S$ with $\sigma(s) < \sigma(t)$ and $c(t, \sigma_0, S) \leq c(s, \sigma, S)$. Then, $u_s \geq u_t$.*

Proof. Suppose, for sake of contradiction, that $u_s < u_t$. If $N^\sigma(s,t) = \emptyset$, then players s and t are neighbors with respect to σ, so interchanging the positions of players s and t results in cost savings of $\alpha_t p_s - \alpha_s p_t > 0$ because $u_s < u_t$. Hence, we have a contradiction with the optimality of σ.

Therefore, $N^\sigma(s,t) \neq \emptyset$ and denote the first player of $N^\sigma(s,t)$ with respect to σ by k. We distinguish between three different cases by considering $\frac{\alpha_t}{p_t}$ and $\frac{\alpha_{S^\sigma(s,t)}}{p_{N^\sigma(s,t)}}$, and show that for every case we can find an admissible order for which the total costs for coalition S is less than the total costs with respect to processing order σ which would contradict the optimality of σ.

Case 1: $\frac{\alpha_t}{p_t} > \frac{\alpha_{S^\sigma(s,t)}}{p_{N^\sigma(s,t)}}$. Consider the processing order σ_1 obtained from σ by moving player t to the position directly behind player s (which is the position directly in front of player k, cf. Figure 6.7). Notice that the processing order σ_1 fulfills condition (i) of admissibility because all players in $\bar{S}^\sigma(s,t)$ are positioned behind player t with respect to σ_0 while condition (ii) is satisfied trivially. According to (6.7) the resulting cost difference for coalition S with respect to σ and σ_1 is

$$\begin{aligned} \delta_t^\sigma(k,S) &= \alpha_t p_{N^\sigma[k,t)} - \alpha_{S^\sigma[k,t)} p_t \\ &= \alpha_t p_{N^\sigma(s,t)} - \alpha_{S^\sigma(s,t)} p_t > 0, \end{aligned}$$

which is a contradiction with the optimality of σ.

Case 2: $\frac{\alpha_t}{p_t} = \frac{\alpha_{S^\sigma(s,t)}}{p_{N^\sigma(s,t)}}$. Consider again the admissible processing order σ_1 defined in case 1. In this case the resulting cost difference for coalition S with respect to σ and σ_1 is

$$\begin{aligned} \delta_t^\sigma(k,S) &= \alpha_t p_{N^\sigma[k,t)} - \alpha_{S^\sigma[k,t)} p_t \\ &= \alpha_t p_{N^\sigma(s,t)} - \alpha_{S^\sigma(s,t)} p_t = 0. \end{aligned}$$

Consider processing order σ_2 obtained from σ_1 by interchanging the positions of players s and t (cf. Figure 6.7). This order is also admissible and the resulting cost difference for coalition S with respect to σ_1 and σ_2 is $\alpha_t p_s - \alpha_s p_t > 0$ since $u_s < u_t$. Therefore,

$$\begin{aligned}\sum_{i \in S} \alpha_i \left(C_i(\sigma) - C_i(\sigma_2)\right) &= \sum_{i \in S} \alpha_i \left(C_i(\sigma) - C_i(\sigma_1)\right) + \sum_{i \in S} \alpha_i \left(C_i(\sigma_1) - C_i(\sigma_2)\right) \\ &= 0 + \alpha_t p_s - \alpha_s p_t > 0,\end{aligned}$$

which is a contradiction with the optimality of σ.

Case 3: $\frac{\alpha_t}{p_t} < \frac{\alpha_{S^\sigma(s,t)}}{p_{N^\sigma(s,t)}}$. In this case $S^\sigma(s,t) \neq \emptyset$. Let l be the last player in S between s and t. Consider the processing order σ_3 obtained from σ by moving player s to the position directly behind player l (cf. Figure 6.7). Then, σ_3 is admissible for S and by (6.7) the resulting cost difference for coalition S with respect to σ and σ_3 is

$$\begin{aligned}\delta_s^\sigma(l,S) &= \alpha_{S^\sigma(s,l]} p_s - \alpha_s p_{N^\sigma(s,l]} \\ &\geq \alpha_{S^\sigma(s,t)} p_s - \alpha_s p_{N^\sigma(s,t)} > 0,\end{aligned}$$

where the last inequality follows from $u_s < u_t$ and thus $\frac{\alpha_s}{p_s} < \frac{\alpha_{S^\sigma(s,t)}}{p_{N^\sigma(s,t)}}$. Hence, we have a contradiction with the optimality of σ. □

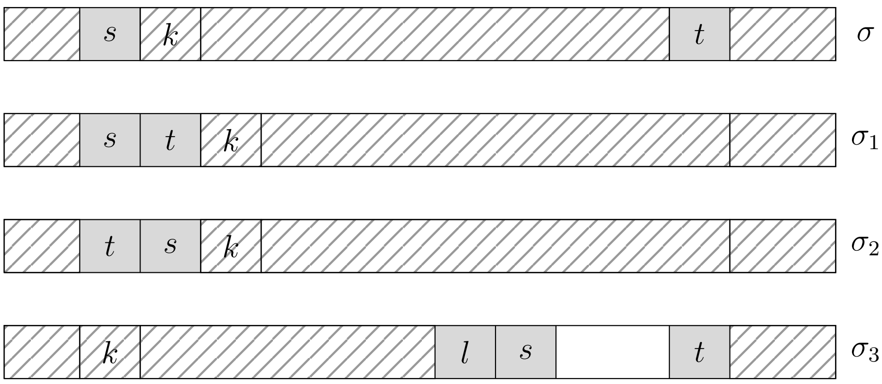


Figure 6.7: The various processing orders used in the proof of Lemma 6.4.1

Given a coalition S, Algorithm 1 determines an urgency respecting optimal order and the corresponding cost savings with respect to σ_0. The algorithm starts in step 0 with reordering the players within components of S with respect to σ_0. This is done by setting the current processing order σ equal to the urgency respecting processing order σ_0^S. The cost savings resulting from these rearrangements are equal to the maximum gain made by all possible neighbor switches within the components as given by

$$\sum_{j=1}^{h(\sigma_0,S)} \sum_{s,t\in S_j^{\sigma_0}:\sigma_0(s)<\sigma_0(t)} g_{st}.$$

Hence, $v(S)$ can be initialized by this value, as is done in step 1.

Subsequently, the players are considered in reverse order with respect to σ_0^S. Note that it is not admissible for a player from the last component with respect to σ_0 to step out from his component. Hence, the algorithm does not consider the players in the last component and the first player to be considered is the last player of the penultimate component. This initialization is done in step 1 where k represents the current modified component and i represents the current player.

In step 2 it is checked whether moving the current player i to a later modified component is beneficial. When one wants to move player i from the modified component k to a later modified component l, the position of this player in modified component l is fixed because the processing order must remain urgency respecting. This is guaranteed by considering the unique option to move the player directly behind the last player in modified component l with strictly higher urgency than player i. In this way, the partial tiebreaking condition is satisfied if later in the course of algorithm's run another player is moved to the same modified component. So, player i is never moved to a subsequent component having no players with higher urgency than player i. Let $A \subseteq \{k+1,\ldots,h(\sigma_0,S)\}$ correspond to all later modified components that contain at least one player with higher urgency than player i. If $A=\emptyset$, then moving player i cannot be beneficial and the processing order is not adapted. With $l \in A$, define t_l as the last player in $S_l^{\sigma_0,\sigma}$ with strictly higher urgency than player i. According to (6.7), for every $l \in A$ the resulting cost savings due to moving player i to $S_l^{\sigma_0,\sigma}$ are equal to $\delta_i^{\sigma}(t_l,S)$. If these costs savings are non-positive for all $l \in A$, then moving player i is not beneficial for S and the processing order is not adapted. On the other hand, if $\delta_i^{\sigma}(t_l,S)$ is positive for some $l \in A$, then player i is moved to the modified component giving the highest cost savings. Moreover, when there is a tie then the component with the smallest index is chosen (such that the partial tiebreaking condition is still satisfied at a later stage if another player joins the same component). With l^* representing the index of the new modified component of player i, processing order σ is adapted by moving player i to the position directly behind player t_{l^*} and the current value of the coalition $v(S)$ is increased by $\delta_i^{\sigma}(t_{l^*},S)$.

In step 3 the settings of k and i are updated. If player i was not the first player of his component, then his old predecessor is the new considered player while going back to step 2. Otherwise, the last player of the component preceding the old component of player i is considered while going back to step 2. Moreover, if player i was the first player of coalition S with respect to processing order σ_0^S, then all players have been considered and the algorithm stops. Note that the algorithm is polynomial and thus terminates in a finite number of steps.

Algorithm 1
Input: a one-machine sequencing situation (N, σ_0, p, α), a coalition $S \in 2^N \backslash \{\emptyset\}$
Output: an urgency respecting optimal order $\sigma \in \mathcal{A}(\sigma_0, S)$, the value $v(S)$

Step 0 (Preprocessing step)
$\sigma := \sigma_0^S$ ▷ Order the players within the components

Step 1 (Initialization)
$v(S) := \sum_{j=1}^{h(\sigma_0,S)} \sum_{s,t \in S_j^{\sigma_0} : \sigma_0(s) < \sigma_0(t)} g_{st}$ ▷ Initialize $v(S)$ with the cost savings from step 0
$k := h(\sigma_0, S) - 1$ ▷ Begin with the penultimate component
$i := \arg\max_{j \in S_k^{\sigma_0}} \sigma(j)$ ▷ Begin with the last player of this component

Step 2 (Improve solution)
$A := \{l \in \{k+1, \ldots, h(\sigma_0, S)\} \mid \exists j \in S_l^{\sigma_0,\sigma} : u_j > u_i\}$
$t_l := \arg\max\{\sigma(j) \mid u_j > u_i, j \in S_l^{\sigma_0,\sigma}\}$ for all $l \in A$
if $A \neq \emptyset$ **and** $\max_{l \in A} \delta_i^\sigma(t_l, S) > 0$ **then** ▷ Is an improvement possible?
  $l^* := \min\{l \in A \mid \delta_i^\sigma(t_l, S) = \max_{m \in A} \delta_i^\sigma(t_m, S)\}$
  $\sigma_{\text{old}} := \sigma$
  $\sigma := [i, t_{l^*}]\sigma$ ▷ Revise the processing order
  $v(S) := v(S) + \delta_i^\sigma(t_{l^*}, S)$ ▷ Revise the value of the coalition
end if

Step 3 (Update settings)
if $\sigma_{\text{old}}^{-1}(\sigma_{\text{old}}(i) - 1) \in S$ **then** ▷ Was player i the first player of his component?
  $i := \sigma_{\text{old}}^{-1}(\sigma_{\text{old}}(i) - 1)$ ▷ If not, consider the old predecessor of i
  Go to step 2
else

```
        k := k − 1                          ▷ If yes, go to the previous component
        if k > 0 then
            i := arg max{σ(j) | j ∈ S_k^{σ0,σ}}    ▷ Take the last player of this component
            Go to step 2
        else
            STOP                            ▷ All players are considered
        end if
    end if
```

Let σ_S denote the urgency respecting processing order obtained from Algorithm 1 with respect to coalition S. Note that $R(\sigma_S) \subset S$, as defined in (6.4) in Section 6.3, exactly consists of those players for whom an improvement was found in step 2. Moreover, the players $r_1(\sigma_S), \ldots, r_{|R(\sigma_S)|}(\sigma_S)$ are exactly the players considered by the algorithm, conducted in the same order. Furthermore, $m_k(\sigma_S)$ with $k \in \{1, \ldots, |R(\sigma_S)|\}$ corresponds to the player where player $r_k(\sigma_S)$ is positioned behind (the corresponding t_{l^*}) according to the algorithm. Note that processing order $\tau^{\sigma_S,S,k-1}$ is the processing order obtained by algorithm when player $r_k(\sigma_S)$ is considered and therefore the total cost savings obtained by the algorithm are equal to

$$\sum_{j=1}^{h(\sigma_0,S)} \sum_{s,t \in S_j^{\sigma_0}:\sigma_0(s)<\sigma_0(t)} g_{st} + \sum_{k=1}^{|R(\sigma_S)|} \delta_{r_k(\sigma_S)}^{\tau^{\sigma_S,S,k-1}}(m_k(\sigma_S), S).$$

In the example below Algorithm 1 is explained step by step.

Example 6.4.1. Consider a one-machine sequencing situation (N, σ_0, p, α) with $S \subseteq N$ such that $S = \{1, \ldots, 10\}$. In Figure 6.8 an illustration can be found of initial order σ_0 together with all relevant data on the cost coefficients and processing times (the numbers above and below the players, respectively). The completion times of the players with respect to this initial order are also indicated in the figure (bottom line in bold).

α_i		4	3	4		6	3		3		4	2	2	7	
σ_0	$\bar{S}_0^{\sigma_0}$	1	2	3	$\bar{S}_1^{\sigma_0}$	4	5	$\bar{S}_2^{\sigma_0}$	6	$\bar{S}_3^{\sigma_0}$	7	8	9	10	$\bar{S}_4^{\sigma_0}$
p_i	1	9	6	1	5	4	9	5	8	3	9	10	8	7	2
$C_i(\sigma_0)$	**1**	**10**	**16**	**17**	**22**	**26**	**35**	**40**	**48**	**51**	**60**	**70**	**78**	**85**	**87**

Figure 6.8: Initial order σ_0 in Example 6.4.1

In step 0 of the algorithm σ is set to processing order σ_0^S, see Figure 6.9. Note that $h(\sigma_0, S) = 4$. The resulting gains yielded by these switches within the components are

$$\sum_{j=1}^{4} \sum_{s,t \in S_j^{\sigma_0} : \sigma_0(s) < \sigma_0(t)} g_{st} = 193.$$

We initialize $v(S) := 193$. In step 1, the component to be considered first is the penultimate component, i.e., $k := 3$, and the first player to be considered is the last player of this component, which is player 6, so $i := 6$.

α_i		4	3	4		6	3		3		7	4	2	2	
σ_0^S	$\bar{S}_0^{\sigma_0}$	3	2	1	$\bar{S}_1^{\sigma_0}$	4	5	$\bar{S}_2^{\sigma_0}$	6	$\bar{S}_3^{\sigma_0}$	10	7	9	8	$\bar{S}_4^{\sigma_0}$
p_i	1	1	6	9	5	4	9	5	8	3	7	9	8	10	2
$C_i(\sigma_0^S)$	**1**	**2**	**8**	**17**	**22**	**26**	**35**	**40**	**48**	**51**	**58**	**67**	**75**	**85**	**87**

Figure 6.9: The processing order σ after step 0 in Example 6.4.1

<u>$k = 3$, $i = 6$:</u> Since there exists a player in $S_4^{\sigma_0,\sigma}$ who is more urgent than player 6, for example player 10, we have $A := \{4\}$. This means that it might be beneficial to move player 6 to component $S_4^{\sigma_0,\sigma}$. Since player 7 is the last player in component $S_4^{\sigma_0,\sigma}$ with higher urgency than player 6, player 6 should be moved to the position directly behind player 7 if he is moved to component $S_4^{\sigma_0,\sigma}$, so $t_4 := 7$. Moving player 6 to this component results in a cost difference of

$$\begin{aligned}\delta_6^\sigma(7, S) &= \alpha_{S^\sigma(6,7]} p_6 - \alpha_6 p_{N^\sigma(6,7]} \\ &= (\alpha_{10} + \alpha_7) p_6 - \alpha_6 (p_{\bar{S}_3^{\sigma_0}} + p_{10} + p_7) \\ &= 11 \cdot 8 - 3 \cdot 19 = 31.\end{aligned}$$

Since $\delta_6^\sigma(7, S) > 0$, it is beneficial to move player 6 and $l^* = 4$. Hence, we update the processing order σ by moving player 6 to the position directly behind player 7 and we set $v(S) := 193 + \delta_6^\sigma(7, S) = 224$. The updated processing order σ can be found in Figure 6.10.

$\bar{S}_0^{\sigma_0}$	3	2	1	$\bar{S}_1^{\sigma_0}$	4	5	$\bar{S}_2^{\sigma_0}$	$\bar{S}_3^{\sigma_0}$	10	7	6	9	8	$\bar{S}_4^{\sigma_0}$

Figure 6.10: The processing order σ after player 6 is considered in Example 6.4.1

In step 3 we have to update k and i. Since player 6 was the only player in his component, and thus his previous predecessor is not a member of coalition S, we consider next the component that was in front of player 6 and thus $k := 2$. The considered player is the last player of this component, i.e., $i := 5$.

$k = 2$, $i = 5$: Since $S_3^{\sigma_0,\sigma} = \emptyset$ we know $3 \notin A$. Moreover, as $S_4^{\sigma_0,\sigma}$ does contain a player who is more urgent than player 5, we have $A := \{4\}$. According to the given urgencies, player 5 should be moved to the position directly behind player 6 if he is moved to component $S_4^{\sigma_0,\sigma}$ ($t_4 := 6$). The resulting cost savings are

$$\begin{aligned} \delta_5^{\sigma}(6, S) &= \alpha_{S^{\sigma}(5,6]} p_5 - \alpha_5 p_{N^{\sigma}(5,6]} \\ &= (\alpha_{10} + \alpha_7 + \alpha_6) p_5 - \alpha_5 (p_{\bar{S}_2^{\sigma_0}} + p_{\bar{S}_3^{\sigma_0}} + p_{10} + p_7 + p_6) \\ &= 14 \cdot 9 - 3 \cdot 32 = 30. \end{aligned}$$

Since $\delta_5^{\sigma}(6, S) > 0$ we have $l^* = 4$ and thus the processing order σ is updated as illustrated in Figure 6.11. Moreover, $v(S)$ is increased by $\delta_5^{\sigma}(6, S) = 30$, so $v(S) := 254$.

$\bar{S}_0^{\sigma_0}$	3	2	1	$\bar{S}_1^{\sigma_0}$	4	$\bar{S}_2^{\sigma_0}$	$\bar{S}_3^{\sigma_0}$	10	7	6	5	9	8	$\bar{S}_4^{\sigma_0}$

Figure 6.11: The processing order σ after player 5 is considered in Example 6.4.1

Since the previous predecessor of player 5, player 4, is a member of coalition S, he becomes the new current player (so $i := 4$ and $k := 2$).

$k = 2$, $i = 4$: Since component $S_3^{\sigma_0,\sigma}$ is empty and all players in component $S_4^{\sigma_0,\sigma}$ have urgencies smaller than u_4, we have $A := \emptyset$. This means that it is not possible to reduce the total costs by moving player 4 to a different component. Hence, σ and $v(S)$ are not changed.

Since the predecessor of player 4 is outside S, we next consider the first component ($k := 1$) and the last player of the first component ($i := 1$).

$k = 1$, $i = 1$: Here $A := \{2, 4\}$. According to the urgencies of the players, if player 1 is moved to a different component then the position of player 1 should be either directly behind player 4 ($t_2 := 4$) or directly behind player 10 ($t_4 := 10$). The resulting cost savings are

$$\begin{aligned} \delta_1^{\sigma}(4, S) &= \alpha_{S^{\sigma}(1,4]} p_1 - \alpha_1 p_{N^{\sigma}(1,4]} \\ &= (\alpha_4) p_1 - \alpha_1 (p_{\bar{S}_1^{\sigma_0}} + p_4) \end{aligned}$$

$$= 6 \cdot 9 - 4 \cdot 9 = 18$$

and

$$\begin{aligned}\delta_1^\sigma(10,S) &= \alpha_{S^\sigma(1,10]}p_1 - \alpha_1 p_{N^\sigma(1,10]} \\ &= (\alpha_4+\alpha_{10})p_1 - \alpha_1(p_{\bar{S}_1^{\sigma_0}} + p_4 + p_{\bar{S}_2^{\sigma_0}} + p_{\bar{S}_3^{\sigma_0}} + p_{10}) \\ &= 13 \cdot 9 - 4 \cdot 24 = 21.\end{aligned}$$

Since moving player 1 to component $S_4^{\sigma_0,\sigma}$ results in larger cost savings than moving player 1 to component $S_2^{\sigma_0,\sigma}$, we have $l^* := 4$. Therefore, processing order σ is updated as illustrated in Figure 6.12 and $v(S)$ is increased by $\delta_1^\sigma(10,S) = 21$, so $v(S) := 275$.

$\bar{S}_0^{\sigma_0}$	3	2	$\bar{S}_1^{\sigma_0}$	4	$\bar{S}_2^{\sigma_0}$	$\bar{S}_3^{\sigma_0}$	10	1	7	6	5	9	8	$\bar{S}_4^{\sigma_0}$

Figure 6.12: The processing order σ after player 1 is considered in Example 6.4.1

Since the previous predecessor of player 1, player 2, is also a member of coalition S, he becomes the new current player ($i := 2$ and $k := 1$).

$k = 1$, $i = 2$: Like in the previous step it can be concluded that $A := \{2, 4\}$, $t_2 := 4$ and $t_4 := 10$, so the potential cost savings are

$$\begin{aligned}\delta_2^\sigma(4,S) &= \alpha_{S^\sigma(2,4]}p_2 - \alpha_2 p_{N^\sigma(2,4]} \\ &= (\alpha_4)p_2 - \alpha_2(p_{\bar{S}_1^{\sigma_0}} + p_4) \\ &= 6 \cdot 6 - 3 \cdot 9 = 9\end{aligned}$$

and

$$\begin{aligned}\delta_2^\sigma(10,S) &= \alpha_{S^\sigma(2,10]}p_2 - \alpha_2 p_{N^\sigma(2,10]} \\ &= (\alpha_4+\alpha_{10})p_2 - \alpha_2(p_{\bar{S}_1^{\sigma_0}} + p_4 + p_{\bar{S}_2^{\sigma_0}} + p_{\bar{S}_3^{\sigma_0}} + p_{10}) \\ &= 13 \cdot 6 - 3 \cdot 24 = 6.\end{aligned}$$

Since $\delta_2^\sigma(4,S) > \delta_2^\sigma(10,S)$ we have $l^* := 2$ and thus the processing order σ is modified by moving player 2 to the position directly behind player 4 (cf. Figure 6.13). Moreover, $v(S)$ is increased by $\delta_2^\sigma(4,S) = 9$, so $v(S) := 284$.

The next player to be considered is player 3 ($i := 3$ and $k := 1$).

$\bar{S}_0^{\sigma_0}$	3	$\bar{S}_1^{\sigma_0}$	4	2	$\bar{S}_2^{\sigma_0}$	$\bar{S}_3^{\sigma_0}$	10	1	7	6	5	9	8	$\bar{S}_4^{\sigma_0}$

Figure 6.13: The processing order σ after player 2 is considered in Example 6.4.1

$k = 1$, $i = 3$: Since all players in the components $S_2^{\sigma_0,\sigma}$ and $S_4^{\sigma_0,\sigma}$ have urgency smaller than u_3, we have $A := \emptyset$. Hence, σ and $v(S)$ are not changed.

Next $k := 0$. According to step 4, the algorithm terminates and an optimal order for coalition S is found, namely the processing order in Figure 6.13 which can be summarized as

$$\sigma_S = [2,4][1,10][5,6][6,7]\sigma_0^S.$$

Moreover, the total cost savings obtained are 284. [4] △

The following lemma shows that Algorithm 1 always constructs an urgency respecting admissible order for a certain coalition S. This lemma is used to prove that the processing order found by the algorithm is also optimal with respect to coalition S (see Theorem 6.4.3).

Lemma 6.4.2. *Let (N, σ_0, p, α) be a one-machine sequencing situation and let $S \in 2^N \backslash \{\emptyset\}$. Then Algorithm 1 constructs an urgency respecting admissible order σ_S for coalition S.*

Proof. We only have to prove that σ_S satisfies the partial tiebreaking condition, because the other conditions of admissibility and componentwise optimality are satisfied trivially by construction. Suppose, for sake of contradiction, that σ_S does not satisfy the partial tiebreaking condition. Then there exist $i, j \in S$ with $c(i, S, \sigma_0) = c(j, S, \sigma_0)$, $u_i = u_j$, $\sigma_0(i) < \sigma_0(j)$, while $\sigma_S(i) > \sigma_S(j)$. Without loss of generality we can assume $\sigma_0^S(j) = \sigma_0^S(i) + 1$. Let σ^j be the order obtained during the run of the algorithm just before player j is considered and σ^i be the order obtained during the run of the algorithm just before player i is considered, i.e., immediately after player j is considered. We distinguish between two cases: $\sigma^j = \sigma^i$ and $\sigma^j \neq \sigma^i$.

Case 1: $\sigma^j = \sigma^i$, i.e., player j has not been moved by the algorithm. Then, due to the structure of the algorithm, there is a player $t \in S$ with $c(t, S, \sigma^i) > c(i, S, \sigma_0)$ and

[4] Note that the only feature of the algorithm that has not been illustrated in this example, is the tiebreaking rule in step 2 for choosing l^*.

$u_t > u_i$ such that

$$\delta_i^{\sigma^i}(t,S) > 0,$$

while

$$\delta_j^{\sigma^j}(t,S) \leq 0.$$

However,

$$\begin{aligned}\delta_i^{\sigma^i}(t,S) &= \alpha_{S^{\sigma^i}(i,t]}p_i - \alpha_i p_{N^{\sigma^i}(i,t]}\\ &= \alpha_{S^{\sigma^j}(j,t]}p_i + \alpha_j p_i - \alpha_i p_{N^{\sigma^j}(j,t]} - \alpha_i p_j\\ &= \alpha_{S^{\sigma^j}(j,t]}p_i - \alpha_i p_{N^{\sigma^j}(j,t]} > 0,\end{aligned}$$

and thus

$$\frac{\alpha_j}{p_j} = \frac{\alpha_i}{p_i} < \frac{\alpha_{S^{\sigma^j}(j,t]}}{p_{N^{\sigma^j}(j,t]}}.$$

Therefore, $\delta_j^{\sigma^j}(t,S) = \alpha_{S^{\sigma^j}(j,t]}p_i - \alpha_i p_{N^{\sigma^j}(j,t]} > 0$, which is a contradiction.

Case 2: $\sigma^j \neq \sigma^i$, i.e., player j has been moved by the algorithm. Let $t_j \in S$ be such that $\sigma^i = [j,t_j]\sigma^j$. Due to the structure of the algorithm there is a player $t_i \in S$ with $c(t_i,S,\sigma^i) > c(t_j,S,\sigma^j)$ such that

$$\delta_i^{\sigma^i}(t_i,S) > \delta_i^{\sigma^i}(t_j,S),$$

while

$$\delta_j^{\sigma^j}(t_i,S) \leq \delta_j^{\sigma^j}(t_j,S).$$

However,

$$\begin{aligned}\delta_i^{\sigma^i}(t_i,S) - \delta_i^{\sigma^i}(t_j,S) &= \alpha_{S^{\sigma^i}(i,t_i]}p_i - \alpha_i p_{N^{\sigma^i}(i,t_i]} - \alpha_{S^{\sigma^i}(i,t_j]}p_i + \alpha_i p_{N^{\sigma^i}(i,t_j]}\\ &= \alpha_{S^{\sigma^i}(t_j,t_i]}p_i - \alpha_i p_{N^{\sigma^i}(t_j,t_i]}\\ &= \alpha_{S^{\sigma^j}(t_j,t_i]}p_i + \alpha_j p_i - \alpha_i p_{N^{\sigma^j}(t_j,t_i]} - \alpha_i p_j\\ &= \alpha_{S^{\sigma^j}(t_j,t_i]}p_i - \alpha_i p_{N^{\sigma^j}(t_j,t_i]} > 0,\end{aligned}$$

and thus

$$\frac{\alpha_j}{p_j} = \frac{\alpha_i}{p_i} < \frac{\alpha_{S^{\sigma^j}(t_j,t_i]}}{p_{N^{\sigma^j}(t_j,t_i]}}.$$

Therefore,

$$\begin{aligned}\delta_j^{\sigma^j}(t_i,S)-\delta_j^{\sigma^j}(t_j,S) &= \alpha_{S^{\sigma^j}(j,t_i]}p_j-\alpha_j p_{N^{\sigma^j}(j,t_i]}-\alpha_{S^{\sigma^j}(j,t_j]}p_j+\alpha_j p_{N^{\sigma^j}(j,t_j]} \\ &= \alpha_{S^{\sigma^j}(t_j,t_i]}p_j-\alpha_j p_{N^{\sigma^j}(t_j,t_i]}>0,\end{aligned}$$

which is a contradiction. □

The class of urgency respecting orders for S can be divided into groups with identical costs for S. Within these groups one can select a unique representative σ by imposing that

(i) for all $i,j\in S$ with $c(i,S,\sigma)=c(j,S,\sigma)$:

$$u_i=u_j,\sigma_0(i)<\sigma_0(j)\Rightarrow\sigma(i)<\sigma(j),$$

(ii) there does not exist a different urgency respecting order $\hat{\sigma}$ with $\sum_{i\in S}\alpha_iC_i(\hat{\sigma})=\sum_{i\in S}\alpha_iC_i(\sigma)$ and $i\in S$ such that

$$c(i,S,\hat{\sigma})<c(i,S,\sigma),$$

and

$$c(j,S,\hat{\sigma})=c(j,S,\sigma),$$

for all $j\in N\backslash\{i\}$.

Thus, condition (i) determines the relative order for players with the same urgency and from the same modified component of S with respect to σ. Likewise, condition (ii) ensures that no player in S can be moved to an earlier modified component of σ while the total costs remain the same. In particular, for coalition S there is a unique representative of the urgency respecting optimal orders and denote this order by σ_S^*.

Theorem 6.4.3. *Let (N,σ_0,p,α) be a one-machine sequencing situation and let $S\in 2^N\backslash\{\emptyset\}$. Then $\sigma_S=\sigma_S^*$.*

Proof. Suppose for sake of contradiction $\sigma_S\neq\sigma_S^*$. For simplicity we denote in this proof σ_S^* by σ^*. From Lemma 6.4.2 it follows that σ_S is an urgency respecting admissible order for coalition S and thus there is a well-defined procedure of consecutive movements to go from initial order σ_0 to the urgency respecting order σ_S, as is also the case for order σ^*. Therefore, we can distinguish between the following three cases:

- case 1: $|R(\sigma_S)| \geq 1$ and $|R(\sigma^*)| = 0$, i.e., at least one player switched component in σ_S, but no player switched component in σ^*,
- case 2: $|R(\sigma_S)| = 0$ and $|R(\sigma^*)| \geq 1$, i.e., no player switched component in σ_S, but at least one player switched component in σ^*,
- case 3: $|R(\sigma_S)| \geq 1$ and $|R(\sigma^*)| \geq 1$, i.e., both in σ_S and σ^* there is at least one player who switched component.

Note that the case where no player switched component in both σ_S and σ^*, i.e., $|R(\sigma_S)| = |R(\sigma^*)| = 0$, is not possible because then it would hold that $\sigma_S = \sigma^*(= \sigma_0^S)$.

Case 1: Denote $\tau = \sigma^*(= \tau^{\sigma_S,S,0})$, $r = r_1(\sigma_S)$ and $m = m_1(\sigma_S)$. The algorithm moved player r to the position behind player m, therefore $\delta_r^{\tau}(m,S) > 0$. Consequently, $\delta_r^{\sigma^*}(m,S) > 0$ which is a contradiction with the optimality of σ^*.

Case 2: Denote $\tau = \sigma_S(= \tau^{\sigma^*,S,0})$, $\tau^* = \tau^{\sigma^*,S,1}$, $r = r_1(\sigma^*)$ and $m = m_1(\sigma^*)$. The algorithm did not move player r to the position behind player m, therefore $\delta_r^{\tau}(m,S) \leq 0$, i.e.,

$$\frac{\alpha_r}{p_r} \geq \frac{\alpha_{S^{\tau}(r,m]}}{p_{N^{\tau}(r,m]}} = \frac{\alpha_{S^{\tau^*}[k,r)}}{p_{N^{\tau^*}[k,r)}},$$

where player $k \in N$ is the direct follower of player r with respect to τ. Note that $\bar{S}^{\tau^*}[k,r) = \bar{S}^{\sigma^*}[k,r)$ and $S^{\tau^*}[k,r) \subseteq S^{\sigma^*}[k,r)$ because every follower of player k with respect to τ^* will not be moved anymore. If $S^{\tau^*}[k,r) = S^{\sigma^*}[k,r)$, then

$$\frac{\alpha_r}{p_r} \geq \frac{\alpha_{S^{\sigma^*}[k,r)}}{p_{N^{\sigma^*}[k,r)}},$$

i.e., $\delta_r^{\sigma^*}(k,S) \geq 0$, which is either a contradiction with the optimality of σ^* or with condition (ii) of σ^*. Hence, $S^{\tau^*}[k,r) \subset S^{\sigma^*}[k,r)$. Note that $S^{\sigma^*}[k,r) \backslash S^{\tau^*}[k,r)$ consists of players positioned between players k and r with respect to σ^*, but who are in front of player r with respect to τ^*. Let us call these players the "new players". Define $q \in S^{\sigma^*}[k,r) \backslash S^{\tau^*}[k,r)$ such that

$$\sigma^*(q) \leq \sigma^*(i),$$

for all $i \in S^{\sigma^*}[k,r) \backslash S^{\tau^*}[k,r)$. This means that player q is among the "new players" the player positioned first with respect to σ^* and thus $S^{\sigma^*}[k,q)$ does not contain any “new players". We show that moving player q in front of player k with respect to σ^* does not result in worse costs for coalition S which is a contradiction. As $\tau(q) < \tau(r) < \tau(k)$ and $\sigma^*(k) < \sigma^*(q) < \sigma^*(r)$, we know that the swap of players q and r with respect to σ^* is admissible. Consequently, from Lemma 6.4.1 it follows that $u_q \geq u_r$. If $N^{\tau^*}[k,r) = N^{\sigma^*}[k,q)$, then

$$\frac{\alpha_q}{p_q} \geq \frac{\alpha_r}{p_r} \geq \frac{\alpha_{S^{\tau^*}[k,r)}}{p_{N^{\tau^*}[k,r)}} = \frac{\alpha_{S^{\sigma^*}[k,q)}}{p_{N^{\sigma^*}[k,q)}},$$

i.e., $\delta_q^{\sigma^*}(k,S) \geq 0$, which is either a contradiction with the optimality of σ^* or with condition (ii) of σ^*. Hence, $N^{\tau^*}[k,r) \neq N^{\sigma^*}[k,q)$, i.e., $N^{\tau^*}[k,r) \cap F(\sigma^*,q) \neq \emptyset$. Define $t \in N^{\tau^*}[k,r) \cap F(\sigma^*,q)$ such that

$$\sigma^*(t) \leq \sigma^*(i),$$

for all $i \in N^{\tau^*}[k,r) \cap F(\sigma^*,q)$. Hence, player $t \in N^{\tau^*}[k,r)$ is defined such that $N^{\tau^*}[k,t) = N^{\sigma^*}[k,q)$. This property of player t is used in order to show the contradiction. Denote by $\hat{\tau}$ the processing order that is obtained from τ^* by moving player r to the position directly in front of player t, which is the same as $[r,l]\tau$ where player $l \in N$ is the direct predecessor of player t with respect to τ. Note that since $q \in R(\sigma^*)$ and $\sigma^* \in \mathcal{A}(\sigma_0,S)$, we know from condition (ii) of admissibility that $l \in S$ and thus $\hat{\tau} \in \mathcal{A}(\sigma_0,S)$. Moreover, as the algorithm did not move player r we know $\delta_r^{\tau}(l,S) \leq 0$, and thus

$$\frac{\alpha_q}{p_q} \geq \frac{\alpha_r}{p_r} \geq \frac{\alpha_{S^{\tau}(r,l]}}{p_{N^{\tau}(r,l]}} = \frac{\alpha_{S^{\hat{\tau}}[k,r)}}{p_{N^{\hat{\tau}}[k,r)}} = \frac{\alpha_{S^{\sigma^*}[k,q)}}{p_{N^{\sigma^*}[k,q)}},$$

where the last equality follows from $N^{\hat{\tau}}[k,r) = N^{\tau^*}[k,t) = N^{\sigma^*}[k,q)$. Consequently, $\delta_q^{\sigma^*}(k,S) \geq 0$ which is a contradiction with the optimality of σ^*, or with condition (i) or (ii) of σ^*.

Case 3: We make a further distinction, namely the following three cases:

– case 3.1: $|R(\sigma_S)| > |R(\sigma^*)|$ and

$$[r_k(\sigma_S), m_k(\sigma_S)] = [r_k(\sigma^*), m_k(\sigma^*)],$$

for all $k \in \{1,\ldots,|R(\sigma^*)|\}$. Hence, there are more players in σ_S switching components than in σ^*. Moreover, σ^* is an intermediate processing order of σ_S in the procedure of consecutive movements to go from σ_0 to σ_S.

– case 3.2: $|R(\sigma_S)| < |R(\sigma^*)|$ and

$$[r_k(\sigma_S), m_k(\sigma_S)] = [r_k(\sigma^*), m_k(\sigma^*)],$$

for all $k \in \{1, \ldots, |R(\sigma_S)|\}$. Hence, there are more players in σ^* switching components than in σ_S. Moreover, σ_S is an intermediate processing order of σ^* in the procedure of consecutive movements to go from σ_0 to σ^*.

– case 3.3: There exists a $c \leq \min\{|R(\sigma_S)|, |R(\sigma^*)|\}$ such that

$$[r_c(\sigma_S), m_c(\sigma_S)] \neq [r_c(\sigma^*), m_c(\sigma^*)]$$

and

$$[r_k(\sigma_S), m_k(\sigma_S)] = [r_k(\sigma^*), m_k(\sigma^*)],$$

for all $k < c$. Hence, neither σ_S is an intermediate processing order of σ^* nor σ^* is an intermediate processing order of σ_S in the procedure of consecutive movements to go from σ_0 to σ^* and σ_S.

Note that the case where $|R(\sigma_S)| = |R(\sigma^*)|$ and

$$[r_k(\sigma_S), m_k(\sigma_S)] = [r_k(\sigma^*), m_k(\sigma^*)],$$

for all $k \in \{1, \ldots, |R(\sigma_S)|\}$ is not possible because then it would hold that $\sigma_S = \sigma^*$.

Case 3.1: Denote $\tau = \sigma^*(= \tau^{\sigma_S, S, |R(\sigma^*)|})$, $r = r_{|R(\sigma^*)|+1}(\sigma_S)$ and $m = m_{|R(\sigma^*)|+1}(\sigma_S)$. Similar to case 1 we have $\delta_r^{\sigma^*}(m, S) = \delta_r^{\tau}(m, S) > 0$ which is a contradiction with the optimality of σ^*.

Case 3.2: Denote $\tau = \sigma_S(= \tau^{\sigma^*, S, |R(\sigma_S)|})$, $\tau^* = \tau^{\sigma^*, S, |R(\sigma_S)|+1}$, $r = r_{|R(\sigma_S)|+1}(\sigma^*)$ and $m = m_{|R(\sigma_S)|+1}(\sigma^*)$. Then, using the same arguments as in case 2, we have a contradiction.

Case 3.3: We make a further distinction, namely the following three cases:

– case 3.3.1: $\sigma_0^S(r_c(\sigma_S)) > \sigma_0^S(r_c(\sigma^*))$, i.e., player $r_c(\sigma_S)$ switched component in σ_S but did not switch component in σ^*.

– case 3.3.2: $\sigma_0^S(r_c(\sigma_S)) < \sigma_0^S(r_c(\sigma^*))$, i.e., player $r_c(\sigma^*)$ switched component in σ^* but did not switch component in σ_S.

- case 3.3.3: $r_c(\sigma_S) = r_c(\sigma^*)$, $m_c(\sigma_S) \neq m_c(\sigma^*)$, i.e., player $r_c(\sigma_S)(= r_c(\sigma^*))$ switched component in both σ_S and σ^*, but he is positioned behind two different players.

Case 3.3.1: Denote $\tau^* = \tau^{\sigma^*,S,c-1}(= \tau^{\sigma_S,S,c-1})$, $r = r_c(\sigma_S)$ and $m = m_c(\sigma_S)$. The algorithm moved player r to the position behind player m, therefore $\delta_r^{\tau^*}(m, S) > 0$, i.e.,

$$\frac{\alpha_r}{p_r} < \frac{\alpha_{S^{\tau^*}(r,m]}}{p_{N^{\tau^*}(r,m]}}.$$

Note that $\bar{S}^{\tau^*}(r, m] = \bar{S}^{\sigma^*}(r, m]$ and $S^{\tau^*}(r, m] \subseteq S^{\sigma^*}(r, m]$ because every follower of player r with respect to τ^* will not be moved anymore. If $S^{\tau^*}(r, m] = S^{\sigma^*}(r, m]$, then

$$\frac{\alpha_r}{p_r} < \frac{\alpha_{S^{\sigma^*}(r,m]}}{p_{N^{\sigma^*}(r,m]}},$$

i.e., $\delta_r^{\sigma^*}(m, S) > 0$, which is a contradiction with the optimality of σ^*. Hence, $S^{\tau^*}(r, m] \subset S^{\sigma^*}(r, m]$. Note that $S^{\sigma^*}(r, m]\backslash S^{\tau^*}(r, m]$ consists of players positioned between players r and m with respect to σ^*, but who are in front of player r with respect to τ^*. Let us call these players the "new players". Define $q \in S^{\sigma^*}(r, m]\backslash S^{\tau^*}(r, m]$ such that

$$\sigma^*(q) \geq \sigma^*(i),$$

for all $i \in S^{\sigma^*}(r, m]\backslash S^{\tau^*}(r, m]$. This means that player q is among the "new players" the player positioned last with respect to σ^* and thus $S^{\sigma^*}(q, m]$ does not contain any "new players". We show that moving player q behind player m with respect to σ^* results in cost savings for coalition S which is a contradiction. As $\sigma_0(q) < \sigma_0(r)$ and $\sigma^*(q) > \sigma^*(r)$, we know that the swap of players q and r with respect to σ^* is admissible. Consequently, from Lemma 6.4.1 it follows that $u_q \leq u_r$. If $N^{\tau^*}(r, m] = N^{\sigma^*}(q, m]$, then

$$\frac{\alpha_q}{p_q} \leq \frac{\alpha_r}{p_r} < \frac{\alpha_{S^{\tau^*}(r,m]}}{p_{N^{\tau^*}(r,m]}} = \frac{\alpha_{S^{\sigma^*}(q,m]}}{p_{N^{\sigma^*}(q,m]}},$$

i.e., $\delta_q^{\sigma^*}(m, S) > 0$, which is a contradiction with the optimality of σ^*. Hence, $N^{\tau^*}(r, m] \neq N^{\sigma^*}(q, m]$, i.e., $N^{\tau^*}(r, m] \cap P(\sigma^*, q) \neq \emptyset$. Define $t \in N^{\tau^*}(r, m] \cap P(\sigma^*, q)$ such that

$$\sigma^*(t) \geq \sigma^*(i),$$

for all $i \in N^{\tau^*}(r,m] \cap P(\sigma^*, q)$. Hence, player $t \in N^{\tau^*}(r,m]$ is defined such that $N^{\tau^*}(t,m] = N^{\sigma^*}(q,m]$. This property of player t is used in order to show the contradiction. Observe that

$$\delta_r^{\tau^*}(m,S) = \delta_r^{\tau^*}(t,S) + \delta_r^{\hat{\tau}}(m,S),$$

where $\hat{\tau}$ is the processing order that is obtained from τ^* by moving player r to the position directly behind player t. Note that since $q \in R(\sigma^*)$ and $\sigma^* \in \mathcal{A}(\sigma_0, S)$, we know from condition (ii) of admissibility that $t \in S$ and thus $\hat{\tau} \in \mathcal{A}(\sigma_0, S)$. Since the algorithm moved player r behind player m and because $\tau(t) < \tau(m)$, we know from the greedy aspect of the algorithm that $\delta_r^{\tau^*}(m,S) > \delta_r^{\tau^*}(t,S)$, i.e., $\delta_r^{\hat{\tau}}(m,S) > 0$, and thus

$$\frac{\alpha_q}{p_q} \leq \frac{\alpha_r}{p_r} < \frac{\alpha_{S^{\hat{\tau}}(r,m]}}{p_{N^{\hat{\tau}}(r,m]}} = \frac{\alpha_{S^{\sigma^*}(q,m]}}{p_{N^{\sigma^*}(q,m]}},$$

where the last equality follows from $N^{\hat{\tau}}(r,m] = N^{\tau^*}(t,m] = N^{\sigma^*}(q,m]$. Consequently, $\delta_q^{\sigma^*}(m,S) > 0$ which is a contradiction with the optimality of σ^*.

Case 3.3.2: Denote $\tau = \tau^{\sigma_S,S,c-1} (= \tau^{\sigma^*,S,c-1})$, $\tau^* = \tau^{\sigma^*,S,c}$, $r = r_c(\sigma^*)$ and $m = m_c(\sigma^*)$. Then, using the same arguments as in case 2, we have a contradiction.

Case 3.3.3: We make a further distinction, namely the following three cases:

- case 3.3.3.1: $c(m_c(\sigma_S), S, \sigma_S) > c(m_c(\sigma^*), S, \sigma^*)$, i.e., player $r_c(\sigma_S)(= r_c(\sigma^*))$ is moved with respect to σ_S to a modified component further than with respect to σ^*.
- case 3.3.3.2: $c(m_c(\sigma_S), S, \sigma_S) < c(m_c(\sigma^*), S, \sigma^*)$, i.e., player $r_c(\sigma_S)(= r_c(\sigma^*))$ is moved with respect to σ^* to a modified component further than with respect to σ_S.
- case 3.3.3.3: $c(m_c(\sigma_S), S, \sigma_S) = c(m_c(\sigma^*), S, \sigma^*)$, i.e., player $r_c(\sigma_S)(= r_c(\sigma^*))$ is moved both with respect to σ_S and σ^* to the same modified component.

Case 3.3.3.1: Denote $\bar{\tau} = \tau^{\sigma^*,S,c-1} (= \tau^{\sigma_S,S,c-1})$, $\tau^* = \tau^{\sigma^*,S,c}$, $r = r_c(\sigma_S)(= r_c(\sigma^*))$ and $m = m_c(\sigma_S)$. Observe that

$$\delta_r^{\bar{\tau}}(m,S) = \delta_r^{\bar{\tau}}(m_c(\sigma^*),S) + \delta_r^{\tau^*}(m,S),$$

due to the fact that $\tau^* = [r, m_c(\sigma^*)]\bar{\tau}$. Since the algorithm moved player r behind player m and because $\bar{\tau}(m_c(\sigma^*)) < \bar{\tau}(m)$, we know from the greedy aspect of the algorithm that $\delta_r^{\bar{\tau}}(m, S) > \delta_r^{\bar{\tau}}(m_c(\sigma^*), S)$, i.e., $\delta_r^{\tau^*}(m, S) > 0$. Then, using the same arguments as in case 3.3.1, we have a contradiction.

Case 3.3.3.2: Denote $\bar{\tau} = \tau^{\sigma_S, S, c-1}(= \tau^{\sigma^*, S, c-1})$, $\tau^* = \tau^{\sigma^*, S, c}$, $\tau = \tau^{\sigma_S, S, c}$, $r = r_c(\sigma^*)(= r_c(\sigma_S))$ and $m = m_c(\sigma^*)$. Observe that

$$\delta_r^{\bar{\tau}}(m, S) = \delta_r^{\bar{\tau}}(m_c(\sigma_S), S) + \delta_r^{\tau}(m, S),$$

because $\tau = [r, m_c(\sigma_S)]\bar{\tau}$. Since the algorithm moved player r behind player $m_c(\sigma_S)$, we know from the greedy aspect of the algorithm that $\delta_r^{\bar{\tau}}(m_c(\sigma_S), S) \geq \delta_r^{\bar{\tau}}(m, S)$, i.e., $\delta_r^{\tau}(m, S) \leq 0$. Then, using the same arguments as in case 2, we have a contradiction.

Case 3.3.3.3: Denote $r = r_c(\sigma_S)(= r_c(\sigma^*))$. Since $\tau^{\sigma_S, S, c}$ and $\tau^{\sigma^*, S, c}$ are both urgency respecting, we know $u_r \leq u_{m_c(\sigma_S)}$ and $u_r \leq u_{m_c(\sigma^*)}$. Moreover, by the definition of the algorithm we know $u_r < u_{m_c(\sigma_S)}$. In addition, since $m_c(\sigma_S) \neq m_c(\sigma^*)$, it must hold that $u_r = u_{m_c(\sigma^*)}$. Consequently, since $\sigma_0^S(r) < \sigma_0^S(m_c(\sigma^*))$ (and thus also $\sigma_0(r) < \sigma_0(m_c(\sigma^*))$ because σ_0^S is urgency respecting) and $\sigma^*(r) > \sigma_0^S(m_c(\sigma^*))$, we have a contradiction with condition (i) of σ^*.

As one can see, for every (sub) case there is a contradiction and thus $\sigma_S = \sigma^*$. □

6.5 Key features of the algorithm

In order to show that SoSi sequencing games are convex (Theorem 6.6.1 in Section 6.6), we use specific key features of the previously provided algorithm. Note that for this reason the algorithm cannot be used as a black box in the proof as we use intermediate steps, outcomes and properties of the algorithm. In this section we will identify and summarize these specific features. For example, in Theorem 6.5.3, we will show that in determining an optimal processing order of a coalition $S \cup \{i\}$ in a SoSi sequencing game, the algorithm can start from the optimal processing order found for coalition S.

The following properties follow directly from the definition and the characteristics of the algorithm to find the optimal processing order σ_S for coalition S and will be used in this chapter in order to show that SoSi sequencing games are convex.

– Property (i): After every step during the run of the algorithm, we have a processing order that is urgency respecting with respect to S.

– Property (ii): If during the run of the algorithm a player is moved to a position later in the processing order, then this results in strictly positive cost savings which corresponds to the, at that instance, highest possible cost savings. In case of multiple options, we choose the component that is most to the left and, in that component, we choose the position that is most to the left.

– Property (iii): The mutual order between players who have already been considered will stay the same during the rest of the run of the algorithm.

– Property (iv): The processing order σ_S is the unique optimal processing order such that no player can be moved to an earlier component while the total costs remain the same. Also, if there are two players with the same urgency in the same component, then the player who was first in σ_0 is earlier in processing order σ_S.

– Property (v): If it is admissible with respect to σ_0 to move a player to a component more to the left with respect to order σ_S, then moving this player to this component will lead to higher total costs.

Note that using Lemma 6.4.1 together with the fact that the algorithm moves a player to the left as far as possible (see property (ii) of the algorithm), we have that if the algorithm moves player k to a later component, then the players from coalition S that player k jumps over all have a strict higher urgency than player k.

The following proposition, which will frequently be used later on, provides a basic property of composed costs per time unit and composed processing times. Namely, if every player in a set of players U are individually more urgent than a specific player i, then also the composed job U as a whole is more urgent than player i.

Proposition 6.5.1. *Let $U \subset N$ with $U \neq \emptyset$ and let $i \in N \backslash U$. If $u_i < u_j$ for all $j \in U$, then*

$$\frac{\alpha_i}{p_i} < \frac{\alpha_U}{p_U},$$

or equivalently,

$$\alpha_i p_U - \alpha_U p_i < 0.$$[5]

Proof. Assume $u_i < u_j$ for all $j \in U$, i.e., $\alpha_i p_j < \alpha_j p_i$, for all $j \in U$. By adding these $|U|$ equations we get $\alpha_i \sum_{j \in U} p_j < p_i \sum_{j \in U} \alpha_j$, i.e.,

$$\frac{\alpha_i}{p_i} < \frac{\sum_{j \in U} \alpha_j}{\sum_{j \in U} p_j} = \frac{\alpha_U}{p_U}.$$ □

The following lemma compares the processing orders that are obtained from the algorithm with respect to coalition S and coalition $S \cup \{i\}$, in case player $i \in N \backslash S$ is the only player in the component of $S \cup \{i\}$ with respect to σ_0. This lemma will be the driving force behind Theorem 6.5.3, which in turn is the crux for proving convexity of SoSi sequencing games.

Lemma 6.5.2. *Let (N, σ_0, p, α) be a one-machine sequencing situation, let $S \subset N$ with $S \neq \emptyset$ and let $i \in N \backslash S$ be such that $(S \cup \{i\})^{\sigma_0}_{c(i, S \cup \{i\}, \sigma_0)} = \{i\}$. Then, for all $k \in S$ we have*

$$c(k, S \cup \{i\}, \sigma_{S \cup \{i\}}) \geq c(k, S \cup \{i\}, \sigma_S).$$

Proof. See the appendix in Section 6.8. □

From the previous lemma it follows that if one wants to determine an optimal processing order of a coalition in a SoSi sequencing game, then the information of optimal processing orders of specific subcoalitions can be used. More precisely, if one wants to know the optimal processing order $\sigma_{S \cup \{i\}}$ derived by the algorithm for a coalition $S \cup \{i\}$ with $i \notin S$ and i being the only player in its component in σ_0, then it does not matter whether you take σ_0 or σ_S as initial processing order, as is stated in the following theorem.

Since the initial order will be varied we need some additional notation. We denote the obtained processing order after the complete run of the algorithm for one-machine sequencing situation (N, σ, p, α) with initial order σ and coalition S by $\text{Alg}((N, \sigma, p, \alpha), S)$. Hence, $\text{Alg}((N, \sigma_0, p, \alpha), S) = \sigma_S$.

Theorem 6.5.3. *Let (N, σ_0, p, α) be a one-machine sequencing situation, let $S \subset N$ with $S \neq \emptyset$ and let $i \in N \backslash S$ be such that $(S \cup \{i\})^{\sigma_0}_{c(i, S \cup \{i\}, \sigma_0)} = \{i\}$. Then,*

$$\sigma_{S \cup \{i\}} = Alg((N, \sigma_S, p, \alpha), S \cup \{i\}).$$

[5]Note that this proposition also holds if every $<$ sign is replaced by a $>$, $\leq$ or $\geq$ sign.

Proof. We start with proving that the minimum costs for coalition $S \cup \{i\}$ in the sequencing situation (N, σ_0, p, α) is equal to the minimum costs for coalition $S \cup \{i\}$ in the sequencing situation (N, σ_S, p, α). Then, we show that the two corresponding sets of optimal processing orders are equal. Finally, the fact that the algorithm always selects a unique processing order among the set of all optimal processing orders (property (iv)) completes the proof.

Note that $\mathcal{A}(\sigma_S, S \cup \{i\}) \subseteq \mathcal{A}(\sigma_0, S \cup \{i\})$ and thus

$$\min_{\sigma \in \mathcal{A}(\sigma_0, S\cup\{i\})} \sum_{j \in S\cup\{i\}} \alpha_j C_j(\sigma)) \leq \min_{\sigma \in \mathcal{A}(\sigma_S, S\cup\{i\})} \sum_{j \in S\cup\{i\}} \alpha_j C_j(\sigma)). \tag{6.8}$$

Moreover, from Lemma 6.5.2 we know that for all $k \in S \cup \{i\}$ we have $c(k, S \cup \{i\}, \sigma_{S\cup\{i\}}) \geq c(k, S \cup \{i\}, \sigma_S)$ and thus $\sigma_{S\cup\{i\}} \in \mathcal{A}(\sigma_S, S \cup \{i\})$. As a consequence, since

$$\sum_{j \in S\cup\{i\}} \alpha_j C_j(\sigma_{S\cup\{i\}}) = \min_{\sigma \in \mathcal{A}(\sigma_0, S\cup\{i\})} \sum_{j \in S\cup\{i\}} \alpha_j C_j(\sigma)),$$

we have together with (6.8) that

$$\min_{\sigma \in \mathcal{A}(\sigma_0, S\cup\{i\})} \sum_{j \in S\cup\{i\}} \alpha_j C_j(\sigma)) = \min_{\sigma \in \mathcal{A}(\sigma_S, S\cup\{i\})} \sum_{j \in S\cup\{i\}} \alpha_j C_j(\sigma)). \tag{6.9}$$

Let $\mathcal{O}(\sigma_0, S \cup \{i\})$ and $\mathcal{O}(\sigma_S, S \cup \{i\})$ denote the set of optimal processing orders for coalition $S \cup \{i\}$ in sequencing situation (N, σ_0, p, α) and (N, σ_S, p, α) respectively. We will show $\mathcal{O}(\sigma_0, S \cup \{i\}) = \mathcal{O}(\sigma_S, S \cup \{i\})$.

First, take $\sigma^* \in \mathcal{O}(\sigma_S, S \cup \{i\})$. Since $\mathcal{A}(\sigma_S, S \cup \{i\}) \subseteq \mathcal{A}(\sigma_0, S \cup \{i\})$, we have $\sigma^* \in \mathcal{A}(\sigma_0, S \cup \{i\})$. Moreover, due to (6.9), we also have $\sigma^* \in \mathcal{O}(\sigma_0, S \cup \{i\})$.

Second, take $\sigma^* \in \mathcal{O}(\sigma_0, S \cup \{i\})$. From property (iv) of the algorithm we know that for all $k \in S \cup \{i\}$ we have $c(k, S \cup \{i\}, \sigma^*) \geq c(k, S \cup \{i\}, \sigma_{S\cup\{i\}})$. Therefore, together with $c(k, S\cup\{i\}, \sigma_{S\cup\{i\}}) \geq c(k, S\cup\{i\}, \sigma_S)$ from Lemma 6.5.2, we know $\sigma^* \in \mathcal{A}(\sigma_S, S \cup \{i\})$. Consequently, together with (6.9), we can conclude $\sigma^* \in \mathcal{O}(\sigma_S, S \cup \{i\})$. Hence, we have

$$\mathcal{O}(\sigma_0, S \cup \{i\}) = \mathcal{O}(\sigma_S, S \cup \{i\}).$$

Finally, since the algorithm chooses among all optimal processing orders the order in which the players are in a component to the left as far as possible and because the algorithm chooses a fixed order within the components (property (iv)), we have $\sigma_{S\cup\{i\}} = \mathrm{Alg}((N, \sigma_S, p, \alpha), S \cup \{i\})$. □

It readily follows from the previous theorem that all players in a component to the right of player i with respect to σ_S are not moved to a different component when applying the algorithm to one-machine sequencing situation (N, σ_S, p, α) and coalition $S \cup \{i\}$. This is stated in the following proposition.

Proposition 6.5.4. *Let (N, σ_0, p, α) be a one-machine sequencing situation, let $S \subset N$ with $S \neq \emptyset$ and let $i \in N \backslash S$ be such that $(S \cup \{i\})^{\sigma_0}_{c(i, S\cup\{i\}, \sigma_0)} = \{i\}$. Then for all $k \in S \cap F(\sigma_S, i)$ we have*

$$c(k, S \cup \{i\}, \sigma_{S\cup\{i\}}) = c(k, S \cup \{i\}, \sigma_S),$$

The next proposition states that all players in a component to the left of player i with respect to σ_S are, if they are moved by the algorithm, moved componentwise at least as far as the original component of player i in σ_0. As a consequence, all players that are in $\sigma_{S\cup\{i\}}$ to the left of the original component of player i in σ_0, are not moved by the algorithm when going from σ_S to $\sigma_{S\cup\{i\}}$.

Proposition 6.5.5. *Let (N, σ_0, p, α) be a one-machine sequencing situation, let $S \subset N$ with $S \neq \emptyset$ and let $i \in N \backslash S$ be such that $(S \cup \{i\})^{\sigma_0}_{c(i, S\cup\{i\}, \sigma_0)} = \{i\}$.*

(i) For all $k \in S$ with $c(k, S \cup \{i\}, \sigma_{S\cup\{i\}}) > c(k, S \cup \{i\}, \sigma_S)$ we have

$$c(k, S \cup \{i\}, \sigma_{S\cup\{i\}}) \geq c(i, S \cup \{i\}, \sigma_0),$$

(ii) For all $k \in S$ with $c(k, S \cup \{i\}, \sigma_{S\cup\{i\}}) < c(i, S \cup \{i\}, \sigma_0)$ we have

$$c(k, S \cup \{i\}, \sigma_{S\cup\{i\}}) = c(k, S \cup \{i\}, \sigma_S).$$

The previous proposition follows directly from the following, more technical, lemma. This lemma shows that, when applying the algorithm to one-machine sequencing situation (N, σ_S, p, α) and coalition $S \cup \{i\}$, once a predecessor of player i with respect to σ_S is considered by the algorithm, moving this player to a position that is to the left of the original component of player i in σ_0 is never beneficial.

Lemma 6.5.6. *Let (N, σ_0, p, α) be a one-machine sequencing situation, let $S \subset N$ with $S \neq \emptyset$ and let $i \in N \backslash S$ be such that $(S \cup \{i\})^{\sigma_0}_{c(i, S\cup\{i\}, \sigma_0)} = \{i\}$. Let $m \in S \cap P(\sigma_S, i)$ and $l \in S \cap F(\tau_m, m)$ with $c(l, S \cup \{i\}, \tau_m) < c(i, S \cup \{i\}, \sigma_0)$. Then*

$$\alpha_{(S\cup\{i\})^{\tau_m}(m,l]} p_m - \alpha_m p_{N^{\tau_m}(m,l]} \leq 0, \tag{6.10}$$

where τ_m denotes the processing order during the run of the algorithm for one-machine sequencing situation (N, σ_S, p, α) and coalition $S \cup \{i\}$ just before player m is considered.

Proof. See the appendix in Section 6.8. □

6.6 On the convexity of SoSi sequencing games

In this section we prove that SoSi sequencing games are convex.

Theorem 6.6.1. *Let (N, σ_0, p, α) be a one-machine sequencing situation and let (N, v) be the corresponding SoSi sequencing game. Then, (N, v) is convex.*

Before presenting the formal proof of our main result, we highlight some of its important aspects beforehand. Using (2.2), let $S \in 2^N \backslash \{\emptyset\}$, $i, j \in N$ and let $i \neq j$ be such that $S \subseteq N \backslash \{i, j\}$.

Note that without loss of generality we can assume

- Assumption 1: $\sigma_0(j) < \sigma_0(i)$.
- Assumption 2: $(S \cup \{j\} \cup \{i\})^{\sigma_0}_{c(j, S \cup \{j\} \cup \{i\}, \sigma_0)} = \{j\}$ and $(S \cup \{j\} \cup \{i\})^{\sigma_0}_{c(i, S \cup \{j\} \cup \{i\}, \sigma_0)} = \{i\}$.

The first assumption is harmless because of the symmetric role of i and j in (2.2). The second assumption states that player i and j both are the only player in the component of $S \cup \{j\} \cup \{i\}$ with respect to σ_0. In theory this is no restriction since it is always possible to add dummy players with zero processing times and zero costs per time unit (a more formal explanation can be found in the appendix in Section 6.8). This assumption facilitates the comparison of the marginal contribution of player i to coalition S and the marginal contribution of player i to coalition $S \cup \{j\}$. For example, if one determines the optimal processing order for coalition $S \cup \{i\}$ via initial processing order σ_S and player i is the only player in its component of $S \cup \{j\} \cup \{i\}$ with respect to σ_0 (and thus also with respect to σ_S), then the players of coalition $S \cup \{i\}$ are in every component already ordered with respect to their urgency and thus the preprocessing step of the algorithm can be skipped. As a consequence, the marginal contribution of player i to coalition S can be written as the sum of the positive cost difference of the players who are moved by the algorithm to a different component.

In order to denote the different types of players that are moved, we introduce the following notation. For $U \in 2^N \backslash \{\emptyset\}$ and $k \in N$ such that $U \subseteq N \backslash \{k\}$ and $(U \cup \{k\})^{\sigma_0}_{c(k, U \cup \{k\}, \sigma_0)} = \{k\}$, let $M^k(U)$ denote the set of players who are moved to a different component during the run of the algorithm with respect to one-machine sequencing situation (N, σ_U, p, α) and coalition $U \cup \{k\}$. Since the algorithm only moves players to components that are to the right of its original component in σ_U, we have

$$M^k(U) = \left\{ l \in N \mid c(l, \sigma_{U \cup \{k\}}) > c(l, \sigma_U) \right\}.$$

As the algorithm only moves the players of the coalition $U \cup \{k\}$ and all players outside this coalition are not moved, we have

$$M^k(U) \subseteq (U \cup \{k\}). \tag{6.11}$$

Moreover, from Proposition 6.5.4 and 6.5.5 it follows respectively that

$$M^k(U) \subseteq (P(\sigma_U, k) \cup \{k\}), \tag{6.12}$$

and

$$c(l, \sigma_{U \cup \{k\}}) \geq c(k, \sigma_U), \tag{6.13}$$

for all $l \in M^k(U)$.

In order to prove Theorem 6.6.1 we need to compare the marginal contribution of player i to coalition S and the marginal contribution of player i to coalition $S \cup \{j\}$. As argued above, both marginal contributions can be written as the sum of the positive cost differences of the players who are moved by the algorithm to a different component. In order to compare those cost differences more easily, we first partition the players in $M^i(S)$, based on their position in the processing orders σ_S and $\sigma_{S \cup \{j\}}$, in four subsets. Second, we derive from σ_S a special processing order $\overline{\sigma}$ in such a way that all players from $M^i(S)$ are in $\overline{\sigma}$ and $\sigma_{S \cup \{j\}}$ in the same component. The convexity proof is finished by means of adequately comparing all positive cost differences.

Proof of Theorem 6.6.1. Let $S \in 2^N \backslash \{\emptyset\}$, $i, j \in N$ and $i \neq j$ such that $S \subseteq N \backslash \{i, j\}$. We will prove

$$v(S \cup \{i\}) - v(S) \leq v(S \cup \{j\} \cup \{i\}) - v(S \cup \{j\}). \tag{6.14}$$

We partition the players in $M^i(S)$, based on their position in the processing orders σ_S and $\sigma_{S\cup\{j\}}$, in four subsets. First, note that from (6.11) it follows that $M^i(S) \subseteq S \cup \{i\}$ and thus $j \notin M^i(S)$. From (6.12) it follows that all players in $M^i(S)$ are in σ_S to the left of player i, or player i himself. By assumption 1 we have that player j is to the left of player i in σ_0 (and thus also in σ_S). So, we can split $M^i(S)$ into the following two disjoint sets:

- $M^i_1(S)$: the set of players in $M^i(S)$ who are in σ_S to the left of player j,
- $M^i_2(S)$: the set of players in $M^i(S)$ who are in σ_S between player j and player i, or player i himself.

Based on the position in $\sigma_{S\cup\{j\}}$, we can split $M^i_1(S)$ into another three disjoint subsets:

- $M^i_{1a}(S)$: the set of players in $M^i_1(S)$ who are in $\sigma_{S\cup\{j\}}$ to the left of the original component of player j,
- $M^i_{1b}(S)$: the set of players in $M^i_1(S)$ who are in $\sigma_{S\cup\{j\}}$ between the original components of player j and player i, or in the original component of player j,
- $M^i_{1c}(S)$: the set of players in $M^i_1(S)$ who are in $\sigma_{S\cup\{j\}}$ to the right of the original component of player i.

From Proposition 6.5.4 it follows that all players in $M^i_2(S)$ are in $\sigma_{S\cup\{j\}}$ between the original components of player j and player i, so we do not further split $M^i_2(S)$ into subsets. We have now a partition of $M^i(S)$ in four subsets, namely $\{M^i_{1a}(S), M^i_{1b}(S), M^i_{1c}(S), M^i_2(S)\}$. Moreover, if $i \in M^i(S)$ then $i \in M^i_2(S)$.

The definition of the partition of $M^i(S)$ in four subsets explains the position of the corresponding players in the processing orders σ_S and $\sigma_{S\cup\{j\}}$. The following four claims indicate how the partition also determines the position in the two other processing orders $\sigma_{S\cup\{i\}}$ and $\sigma_{S\cup\{j\}\cup\{i\}}$. For notational convenience, we denote $c(k, S\cup \{i\} \cup \{j\}, \sigma)$ by $c(k,\sigma)$ for every $k \in S \cup \{i\} \cup \{j\}$ and $\sigma \in \Pi(N)$.

- **Claim 1:** $c(k, \sigma_{S\cup\{j\}\cup\{i\}}) = c(k, \sigma_{S\cup\{i\}}) \geq c(i, \sigma_S)$ for all $k \in M^i(S)$.
- **Claim 2:** $c(k, \sigma_{S\cup\{j\}\cup\{i\}}) = c(k, \sigma_{S\cup\{j\}})$ for all $k \in M^i_{1c}(S)$.
- **Claim 3:** $c(k, \sigma_S) = c(k, \sigma_{S\cup\{j\}})$ for all $k \in M^i_2(S)$.
- **Claim 4:** $c(k, \sigma_S) = c(k, \sigma_{S\cup\{j\}})$ for all $k \in M^i_{1a}(S)$.

The proofs of these four claims can be found in the appendix in Section 6.8. Figure 6.14 illustrates for all four partition elements of $M^i(S)$ its position with respect to the original components of player i and player j in the four different processing orders. The solid arrows give the original components and/or the actual positions of player i and j. The dotted arrows give possible positions of player i or j.

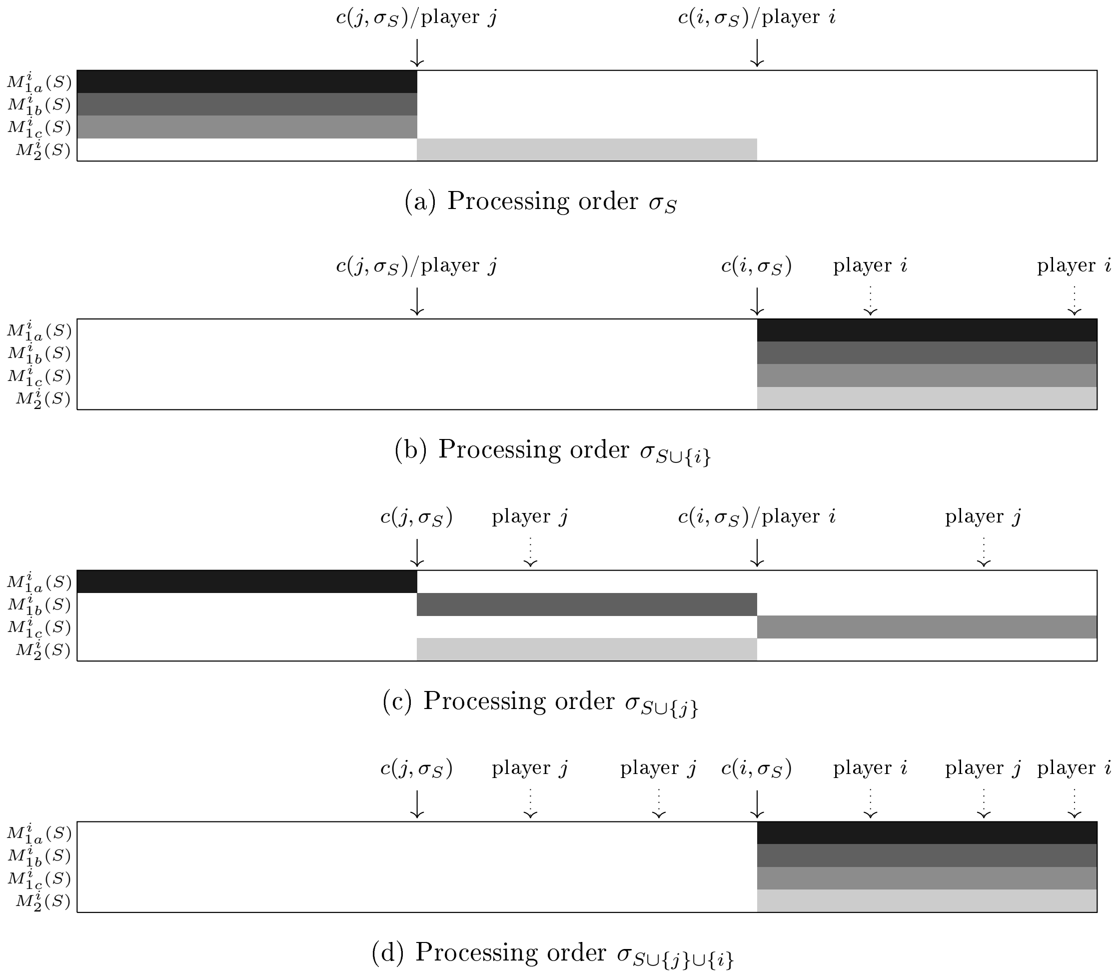


Figure 6.14: The position of the players in $M^i(S)$ in the four different processing orders

We define $\overline{\sigma} \in \Pi(N)$ as the unique urgency respecting processing order that satisfies

(i) for all $k \in M^i(S)$:

$$c(k, \overline{\sigma}) = c(k, \sigma_{S\cup\{j\}}), \tag{6.15}$$

(ii) for all $k \in S \backslash M^i(S)$:

$$c(k, \overline{\sigma}) = c(k, \sigma_S),$$

(iii) for all $k, l \in S$ with $c(k, \overline{\sigma}) = c(l, \overline{\sigma})$:

$$u_k = u_l, \sigma_0(k) < \sigma_0(l) \Rightarrow \overline{\sigma}(k) < \overline{\sigma}(l).$$

Note that conditions (i) and (ii) determine the components for the players in S. Next, the urgency respecting requirement determines the order within the components for the players with different urgencies. Finally, in case there is a tie for the urgency of two players in the same component, item (iii) states a tiebreaking rule. As a consequence, due to this tiebreaking rule, we have that $\overline{\sigma}$ is unique.

Note that $\overline{\sigma}$ can be considered as a temporary processing order when going from σ_S to $\sigma_{S \cup \{i\}}$ (cf. Figure 6.15). The processing order $\overline{\sigma}$ is derived from processing order σ_S in such a way that all players from $M^i(S)$ are in $\overline{\sigma}$ and $\sigma_{S \cup \{j\}}$ in the same component. From claim 3 and 4 it follows that the players in $M^i_{1a}(S)$ and $M^i_2(S)$ are in σ_S and $\sigma_{S \cup \{j\}}$ in the same component and thus those players do not need to be moved. Hence, only the players in $M^i_{1b}(S)$ and $M^i_{1c}(S)$ need to be moved. Hence, we start from σ_S and we move all players in $M^i_{1b}(S)$ and $M^i_{1c}(S)$ to the components they are in in $\sigma_{S \cup \{j\}}$. Note that since the tiebreaking rule mentioned in condition (iii) is the same tiebreaking rule as in property (iv) of the algorithm, the mutual order of the players in $M^i(S)$ is in $\overline{\sigma}$ the same as in $\sigma_{S \cup \{j\}}$.

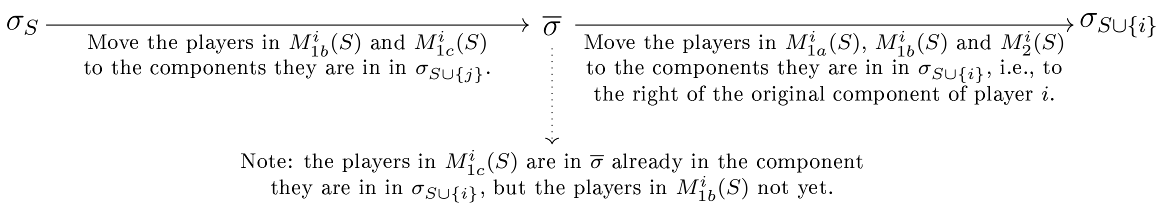


Figure 6.15: Overview how to obtain $\sigma_{S \cup \{i\}}$ from σ_S via $\overline{\sigma}$

An illustration of the position of the players in $M^i(S)$ in $\overline{\sigma}$ can be found in Figure 6.16. Note that since $i \notin S \cup \{j\}$ it follows that $c(i, \sigma_S) = c(i, \sigma_{S \cup \{j\}})$. Moreover, we note that $\overline{\sigma}$ and $\sigma_{S \cup \{j\}}$ are not necessarily equal to each other as the players in $M^j(S) \backslash M^i(S)$ are in $\overline{\sigma}$ and $\sigma_{S \cup \{j\}}$ in different components. However, as the players in $M^j(S)$ will be moved to a component to the right when going from σ_S to $\sigma_{S\cup\{j\}}$, we have

$$c(k, \sigma_{S\cup\{j\}}) \geq c(k, \overline{\sigma}), \tag{6.16}$$

for all $k \in S \cup \{j\} \cup \{i\}$.

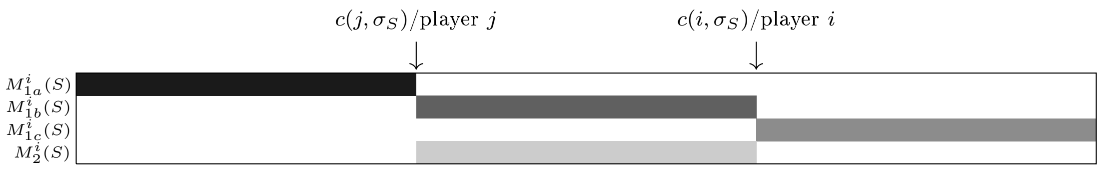


Figure 6.16: The position of the players in $M^i(S)$ in $\overline{\sigma}$

Now we consider the transition from $\overline{\sigma}$ to $\sigma_{S\cup\{i\}}$ and its corresponding cost differences. Since the players in $M^i_{1c}(S)$ are in $\overline{\sigma}$ already in the component they are in in $\sigma_{S\cup\{i\}}$, only all players in $M^i_{1a}(S)$, $M^i_{1b}(S)$ and $M^i_2(S)$ need to be moved to a component to the right when going from $\overline{\sigma}$ to $\sigma_{S\cup\{i\}}$ (see also Figure 6.15). We go from $\overline{\sigma}$ to $\sigma_{S\cup\{i\}}$ by considering the players in $M^i_{1a}(S)$, $M^i_{1b}(S)$ and $M^i_2(S)$ in an order reverse to the order they are in $\overline{\sigma}$, i.e., the players are considered from the right to the left. For $k \in M^i_{1a}(S) \cup M^i_{1b}(S) \cup M^i_2(S)$, denote the processing order just before player k is moved by $\overline{\tau}_k$ and let $\overline{r}_k$ denote the player where player k will be moved behind. The cost difference for coalition $S \cup \{i\}$ due to moving this player, when going from $\overline{\sigma}$ to $\sigma_{S\cup\{i\}}$, is denoted by $\overline{\delta}_k$, i.e.,

$$\overline{\delta}_k = \alpha_{(S\cup\{i\})^{\overline{\tau}_k}(k,\overline{r}_k]} p_k - \alpha_k p_{N^{\overline{\tau}_k}(k,\overline{r}_k]}.$$

Similarly, we can write the marginal contribution of player i to coalition $S \cup \{j\}$ as the sum of positive cost differences of the players in $M^i(S \cup \{j\})$. We go from $\sigma_{S\cup\{j\}}$ to $\sigma_{S\cup\{j\}\cup\{i\}}$ by considering the players in $M^i(S \cup \{j\})$ in an order reverse to the order they are in $\sigma_{S\cup\{j\}}$, i.e., the players are considered from the right to the left. We note that since the mutual order of the players in $M^i(S)$ is the same in $\overline{\sigma}$ and $\sigma_{S\cup\{j\}}$, the order in which the players in $M^i_{1a}(S) \cup M^i_{1b}(S) \cup M^i_2(S)$ are considered when going from $\overline{\sigma}$ to $\sigma_{S\cup\{i\}}$ is the same as the order in which they are considered when going from $\sigma_{S\cup\{j\}}$ to $\sigma_{S\cup\{j\}\cup\{i\}}$. For $k \in M^i(S \cup \{j\})$, denote the processing order just before player k is moved by τ_k and let r_k denote the player where player k

will be moved behind. The cost difference for coalition $S \cup \{j\} \cup \{i\}$ due to moving this player, when going from $\sigma_{S\cup\{j\}}$ to $\sigma_{S\cup\{j\}\cup\{i\}}$, is denoted by δ_k, i.e.,

$$\delta_k = \alpha_{(S\cup\{j\}\cup\{i\})^{\tau_k}(k,r_k]}p_k - \alpha_k p_{N^{\tau_k}(k,r_k]}.$$

From (6.16) together with the fact that the players in $M^i_{1a}(S) \cup M^i_{1b}(S) \cup M^i_2(S)$ are moved to the same component in $\sigma_{S\cup\{i\}}$ and $\sigma_{S\cup\{j\}\cup\{i\}}$, and the fact that the players in $M^j(S)\backslash M^i(S)$ are moved to a component to the right, we have

$$c(l, \tau_k) \geq c(l, \overline{\tau}_k), \tag{6.17}$$

for all $k \in M^i_{1a}(S) \cup M^i_{1b}(S) \cup M^i_2(S)$ and $l \in S \cup \{j\} \cup \{i\}$.

The following claim states that the cost savings when moving a player in $M^i_{1a}(S) \cup M^i_{1b}(S) \cup M^i_2(S)$ when going from $\overline{\sigma}$ to $\sigma_{S\cup\{i\}}$ is at most the cost savings when moving the same player when going from $\sigma_{S\cup\{j\}}$ to $\sigma_{S\cup\{j\}\cup\{i\}}$.

Claim 5: $\overline{\delta}_k \leq \delta_k$ for all $k \in M^i_{1a}(S) \cup M^i_{1b}(S) \cup M^i_2(S)$.
Proof. The proof can be found in the appendix in Section 6.8.

We are now ready to prove (6.14). Note that a detailed explanation of the subsequent equalities and inequalities can be found after the equations.

$$\begin{aligned}
& v(S\cup\{i\}) - v(S) \\
\stackrel{\text{(i)}}{=} & \sum_{k\in P(\sigma_0,i)\cap F(\sigma_S,i)} \alpha_i p_k + C(\sigma_S, S\cup\{i\}) - C(\sigma_{S\cup\{i\}}, S\cup\{i\}) \\
\stackrel{\text{(ii)}}{=} & \sum_{k\in P(\sigma_0,i)\cap F(\sigma_S,i)} \alpha_i p_k + C(\sigma_S, S\cup\{i\}) - C(\overline{\sigma}, S\cup\{i\}) \\
& + C(\overline{\sigma}, S\cup\{i\}) - C(\sigma_{S\cup\{i\}}, S\cup\{i\}) \\
\stackrel{\text{(iii)}}{=} & \sum_{k\in P(\sigma_0,i)\cap F(\sigma_S,i)} \alpha_i p_k + C(\sigma_S, S) - C(\overline{\sigma}, S) + \sum_{k\in M^i_{1c}(S)} \alpha_i p_k \\
& + C(\overline{\sigma}, S\cup\{i\}) - C(\sigma_{S\cup\{i\}}, S\cup\{i\}) \\
\stackrel{\text{(iv)}}{\leq} & \sum_{k\in P(\sigma_0,i)\cap F(\sigma_S,i)} \alpha_i p_k + \sum_{k\in M^i_{1c}(S)} \alpha_i p_k + C(\overline{\sigma}, S\cup\{i\}) - C(\sigma_{S\cup\{i\}}, S\cup\{i\}) \\
\stackrel{\text{(v)}}{=} & \sum_{k\in P(\sigma_0,i)\cap F(\sigma_S,i)} \alpha_i p_k + \sum_{k\in M^i_{1c}(S)} \alpha_i p_k + \sum_{k\in M^i_{1a}(S)\cup M^i_{1b}(S)\cup M^i_2(S)} \overline{\delta}_k
\end{aligned}$$

$$\begin{aligned}
&\overset{\text{(vi)}}{\leq} \sum_{k \in P(\sigma_0,i) \cap F(\sigma_S,i)} \alpha_i p_k + \sum_{k \in M^i_{1c}(S)} \alpha_i p_k + \sum_{k \in M^i_{1a}(S) \cup M^i_{1b}(S) \cup M^i_2(S)} \delta_k \\
&\overset{\text{(vii)}}{\leq} \sum_{k \in P(\sigma_0,i) \cap F(\sigma_S,i)} \alpha_i p_k + \sum_{k \in M^i_{1c}(S)} \alpha_i p_k + \sum_{k \in M^i(S \cup \{j\})} \delta_k \\
&\overset{\text{(viii)}}{=} \sum_{k \in P(\sigma_0,i) \cap F(\sigma_S,i)} \alpha_i p_k + \sum_{k \in M^i_{1c}(S)} \alpha_i p_k \\
&\qquad + C(\sigma_{S \cup \{j\}}, S \cup \{j\} \cup \{i\}) - C(\sigma_{S \cup \{j\} \cup \{i\}}, S \cup \{j\} \cup \{i\}) \\
&\overset{\text{(ix)}}{\leq} \sum_{k \in P(\sigma_0,i) \cap F(\sigma_S,i)} \alpha_i p_k + \sum_{k \in P(\sigma_S,i) \cap F(\sigma_{S \cup \{j\}},i)} \alpha_i p_k \\
&\qquad + C(\sigma_{S \cup \{j\}}, S \cup \{j\} \cup \{i\}) - C(\sigma_{S \cup \{j\} \cup \{i\}}, S \cup \{j\} \cup \{i\}) \\
&\overset{\text{(x)}}{=} \sum_{k \in P(\sigma_0,i) \cap F(\sigma_{S \cup \{j\}},i)} \alpha_i p_k + C(\sigma_{S \cup \{j\}}, S \cup \{j\} \cup \{i\}) - C(\sigma_{S \cup \{j\} \cup \{i\}}, S \cup \{j\} \cup \{i\}) \\
&\overset{\text{(xi)}}{=} v(S \cup \{j\} \cup \{i\}) - v(S \cup \{j\}),
\end{aligned}$$

which proves (6.14).

Explanations:

(i) The extra worth that is obtained by adding player i to coalition S can be split into two parts. The first part is due to the fact that player i joins the coalition and it represents the cost savings for player i in processing order σ_S compared to σ_0. The completion time of player i is reduced by the sum of the processing times of the players that jumped over player i when going from σ_0 to σ_S without moving any players. The second part represents the cost savings for coalition $S \cup \{i\}$ by additionally moving players when going from σ_S to the optimal processing order $\sigma_{S \cup \{i\}}$.

(ii) The optimal processing order $\sigma_{S \cup \{i\}}$ can be obtained from σ_S via $\overline{\sigma}$ where some players are already (partially) moved to the right.

(iii) The cost difference for coalition $S \cup \{i\}$ when going from σ_S to $\overline{\sigma}$ can be split into two parts: the cost difference for coalition S and the cost difference for player i. By the definition of $\overline{\sigma}$ and since $i \notin S \cup \{j\}$, player i is not moved when going from σ_S to $\overline{\sigma}$ and the completion time of player i is reduced by the sum of the processing times of the players that jumped over player i when going from σ_S to $\overline{\sigma}$, i.e., the sum of the processing times of the players in $M^i_{1c}(S)$.

(iv) Processing order σ_S is optimal for coalition S and thus $C(\sigma_S, S) - C(\overline{\sigma}, S) \leq 0$.

(v) This follows from the definition of $\overline{\delta}_k$.

(vi) This follows from claim 5.

(vii) This follows from $(M^i_{1a}(S) \cup M^i_{1b}(S) \cup M^i_2(S)) \subseteq M^i(S \cup \{j\})$ (cf. Figure 6.14) and $\delta_k > 0$ for all $k \in M^i(S \cup \{j\})$ due to property (ii) of the algorithm.

(viii) This follows from the definition of δ_k.

(ix) This follows from $M^i_{1c}(S) \subseteq (P(\sigma_S, i) \cap F(\sigma_{S\cup\{j\}}, i))$ (cf. Figure 6.14).

(x) The group of players that jump over player i when going from σ_0 to $\sigma_{S\cup\{j\}}$ can be split into two groups: the group of players that jumped over player i when going from σ_0 to σ_S and the group of players that were positioned in front of player i in σ_S but jumped over player i when going from σ_S to $\sigma_{S\cup\{j\}}$. Hence, $\{P(\sigma_0, i)\cap F(\sigma_S, i), P(\sigma_S, i)\cap F(\sigma_{S\cup\{j\}}, i)\}$ is a partition of $P(\sigma_0, i)\cap F(\sigma_{S\cup\{j\}}, i)$.

(xi) Similar to the explanation in (i).

To conclude, we have shown $v(S \cup \{i\}) - v(S) \leq v(S \cup \{j\} \cup \{i\}) - v(S \cup \{j\})$ which proves the convexity of SoSi sequencing games. □

6.7 Concluding remarks

Closely related to the class of SoSi sequencing games is the class of so-called *Step out sequencing games*. In a Step out sequencing game a member of a coalition S can swap with another member of the coalition S who is in the same component of S with respect to σ_0, while he is also allowed to step out from his initial position in the processing order and enter at the rear of the processing order. Hence, in contrast to a SoSi sequencing game, a player in S cannot join any position later in the processing order but only the rear of the processing order. Example 6.7.1 illustrates the differences between admissible orders in a Step out sequencing game and a SoSi sequencing game.

Example 6.7.1. Consider a one-machine sequencing situation (N, σ_0, p, α) with $S \subseteq N$ such that $S = \{1, 2, 3\}$. Assume that the components of coalition S with respect to σ_0 are given by

$$S_1^{\sigma_0} = \{1\} \text{ and } S_2^{\sigma_0} = \{2, 3\}.$$

Moreover, assume that $\bar{S}_2^{\sigma_0} \neq \emptyset$ (cf. Figure 6.17(a)). In a SoSi sequencing game it is allowed for player 1 to step out from his position in the processing order and to step in in the second component of S, for example in between players 2 and 3. However, this is not allowed in a Step out sequencing game, because in a Step out sequencing game player 1 can only enter at the rear of the processing order. This means that processing order $\hat{\sigma}$ in Figure 6.17(b) is admissible in a SoSi sequencing game but not in a Step out sequencing game. △

(a) (b)

Figure 6.17: The processing orders used in Example 6.7.1

Using similar arguments as in the proof of Theorem 6.3.2, it can be shown that Step out sequencing games also belong to the class of relaxed sequencing games as discussed in Theorem 6.3.1, which implies that every Step out sequencing game has a non-empty core.

Finding a polynomial time algorithm determining an optimal order for every possible coalition for a Step out sequencing game turns out to be different than for a SoSi sequencing game. Example 6.7.2 gives an example of a Step out sequencing game where moving some players individually is not beneficial, but moving these players simultaneously is. This illustrates that the type of algorithm that works for SoSi sequencing games cannot be applied to Step out sequencing games because the algorithm in Section 6.4 moves players individually. It would be interesting to try and develop a polynomial time algorithm for determining an optimal order for every possible coalition in a Step out sequencing game, if possible.

Example 6.7.2. Reconsider the one-machine sequencing situation (N, σ_0, p, α) of Example 6.7.1 with $S = \{1, 2, 3\}$. In Figure 6.18 an illustration can be found of an initial order σ_0 together with the cost coefficients and processing times (the numbers above and below the players, respectively). The completion times of the players with respect to this initial order are also indicated in the figure (bottom line in bold).

Some alternative admissible orders are given in Figure 6.18. The corresponding cost savings for coalition S with respect to these admissible orders are as follows

$$\begin{aligned}\sum_{i\in\{1,2,3\}} \alpha_i(C_i(\sigma_0)-C_i(\sigma_1)) &= 0+0-1=-1,\\ \sum_{i\in\{1,2,3\}} \alpha_i(C_i(\sigma_0)-C_i(\sigma_2)) &= -105+72+8=-25,\\ \sum_{i\in\{1,2,3\}} \alpha_i(C_i(\sigma_0)-C_i(\sigma_3)) &= -49+72-1=22.\end{aligned}$$

Hence, both moving player 1 and player 3 individually to the rear of the processing order is not beneficial (cf. σ_1 and σ_2, respectively). However, if you move players 1 and 3 simultaneously to the rear of the processing order, then some cost savings are obtained (cf. σ_3). △

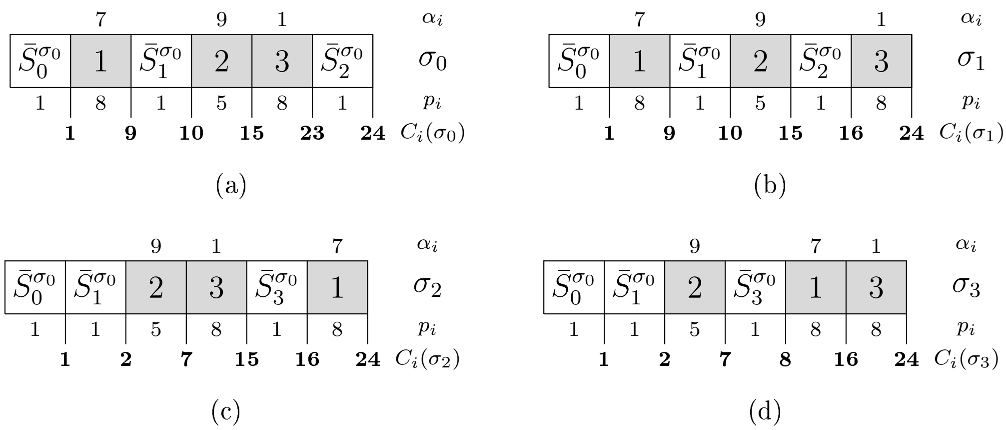


Figure 6.18: Various processing orders used in Example 6.7.2

Note that we applied our SoSi relaxation only to the classical sequencing setting considered by Curiel et al. (1989). Another direction of future research is to apply the SoSi relaxation to other types of sequencing models as well.

6.8 Appendix

Proof of Lemma 6.5.2

In this proof we denote $c(k, S\cup\{i\}, \sigma)$ by $c(k,\sigma)$ for every $k \in S\cup\{i\}$ and every $\sigma \in \Pi(N)$. We prove the lemma with the help of the algorithm. First, note that because player i is the only player in its component in σ_0, we have $\sigma_0^S = \sigma_0^{S\cup\{i\}}$, i.e., the processing orders are the same after the preprocessing step of the algorithm. Therefore, if we go from σ_0 to the optimal processing orders σ_S and $\sigma_{S\cup\{i\}}$, then the steps performed by the algorithm are the same up to the moment that player i is considered. Moreover, since the players are considered in reverse order with respect to σ_0^S, we have

$$c(k, \sigma_{S\cup\{i\}}) = c(k, \sigma_S), \tag{6.18}$$

for all $k \in S \cap F(\sigma_0^S, i)$. Hence, it remains to be proven that also for the players in $S \cap P(\sigma_0^S, i)$ the lemma is true.

Let player $m \in S$ be the closest predecessor of player i with respect to σ_0^S for which the lemma is not true, i.e.,

$$c(m, \sigma_{S\cup\{i\}}) < c(m, \sigma_S),$$

and

$$c(k, \sigma_{S\cup\{i\}}) \geq c(k, \sigma_S),$$

for all $k \in S \cap F(\sigma_0^S, m) \cap P(\sigma_0^S, i)$. We will derive a contradiction. We continue the proof as follows. We look to which component player m will be moved by the algorithm with respect to coalition S. Then, there is a specific player who is in a component with index at least as high as the component that player m is moved by the algorithm with respect to coalition S. We show that moving player m behind this specific player is actually more beneficial with respect to coalition $S \cup \{i\}$, which contradicts the optimality of the algorithm.

Denote the processing order when player m is considered by the algorithm with respect to coalition S by τ^S and with respect to coalition $S \cup \{i\}$ by $\tau^{S\cup\{i\}}$. Let r^S denote the player where player m will be moved behind with respect to coalition S according to the algorithm. Similarly, let $r^{S\cup\{i\}}$ denote the player where player m will be moved behind with respect to coalition $S \cup \{i\}$ according to the algorithm. Note that in case player m is not moved by the algorithm with respect to coalition $S\cup\{i\}$, then we define player $r^{S\cup\{i\}}$ as player m. Since $c(m, \sigma_{S\cup\{i\}}) = c(r^{S\cup\{i\}}, \tau^{S\cup\{i\}})$ and $c(m, \sigma_S) = c(r^S, \tau^S)$, we have

$$c(r^{S\cup\{i\}}, \tau^{S\cup\{i\}}) < c(r^S, \tau^S). \tag{6.19}$$

As we will see later $(S\cup\{i\})^{\sigma_0,\tau^S}_{c(r^{S\cup\{i\}},\tau^{S\cup\{i\}})}\neq\emptyset$, let $\tilde{r}^S\in(S\cup\{i\})^{\sigma_0,\tau^S}_{c(r^{S\cup\{i\}},\tau^{S\cup\{i\}})}$ be such that player m would be moved behind this player in case player m is moved to component $(S\cup\{i\})^{\sigma_0,\tau^S}_{c(r^{S\cup\{i\}},\tau^{S\cup\{i\}})}$ according to the algorithm with respect to coalition S . Note that player $\tilde{r}^S$ is unique because the algorithm always selects a unique player per component. Note that player $\tilde{r}^S$ might also be player i as in this way we make sure that player $\tilde{r}^S$ also exists if $c(r^{S\cup\{i\}},\tau^{S\cup\{i\}})=c(i,\sigma_0)$.[6] Note that in case player m and $r^{S\cup\{i\}}$ coincide (that means that player m is not moved by the algorithm with respect to coalition $S\cup\{i\}$), then we define player $\tilde{r}^S$ as player m.

Since player m is moved behind player r^S and not behind player $\tilde{r}^S$, we have due to property (v) of the algorithm that

$$\alpha_{S^{\tau^S}(\tilde{r}^S,r^S]}p_m-\alpha_m p_{N^{\tau^S}(\tilde{r}^S,r^S]}>0. \tag{6.20}$$

We distinguish between two cases:

- case A: $\{k\in S\mid\sigma_0^S(m)<\sigma_0^S(k)<\sigma_0^S(i)\}=\emptyset$,
- case B: $\{k\in S\mid\sigma_0^S(m)<\sigma_0^S(k)<\sigma_0^S(i)\}\neq\emptyset$.

<u>Case A:</u> $[\{k\in S\mid\sigma_0^S(m)<\sigma_0^S(k)<\sigma_0^S(i)\}\neq\emptyset]$
Hence, there are no players of coalition S in between player m and player i in σ_0^S. Then, from (6.18) it follows that for every $k\in S\cap F(\tau^S,m)$ we have

$$c(k,\tau^S)=c(k,\tau^{S\cup\{i\}}). \tag{6.21}$$

Note that this implies that $(S\cup\{i\})^{\sigma_0,\tau^S}_{c(r^{S\cup\{i\}},\tau^{S\cup\{i\}})}$ is non-empty.

We will prove that moving player m behind player r^S is more beneficial than moving player m behind player $r^{S\cup\{i\}}$, i.e., we will prove that

$$\alpha_{(S\cup\{i\})^{\tau^{S\cup\{i\}}}(r^{S\cup\{i\}},r^S]}p_m-\alpha_m p_{N^{\tau^{S\cup\{i\}}}(r^{S\cup\{i\}},r^S]}>0. \tag{6.22}$$

This would imply that the step made by the algorithm for player m when applied on coalition $S\cup\{i\}$ is not optimal, which contradicts the optimality of the algorithm. Hence, for case A, it remains to prove (6.22).

We distinguish from now on between the following four cases:

[6] Note that in case $c(r^{S\cup\{i\}},\tau^{S\cup\{i\}})=c(i,\sigma_0)$ it is not admissible for the algorithm to move player m to component $(S\cup\{i\})^{\sigma_0,\tau^S}_{c(r^{S\cup\{i\}},\tau^{S\cup\{i\}})}$ due to requirement (ii) of admissibility, but this is no problem as also in this case the upcoming arguments are still valid.

- case A.1: $i \notin N^{\tau^S}(\tilde{r}^S, r^S]$ and $i \notin N^{\tau^{S\cup\{i\}}}(r^{S\cup\{i\}}, r^S]$,
- case A.2: $i \in N^{\tau^S}(\tilde{r}^S, r^S]$ and $i \notin N^{\tau^{S\cup\{i\}}}(r^{S\cup\{i\}}, r^S]$,
- case A.3: $i \in N^{\tau^S}(\tilde{r}^S, r^S]$ and $i \in N^{\tau^{S\cup\{i\}}}(r^{S\cup\{i\}}, r^S]$,
- case A.4: $i \notin N^{\tau^S}(\tilde{r}^S, r^S]$ and $i \in N^{\tau^{S\cup\{i\}}}(r^{S\cup\{i\}}, r^S]$.

<u>Case A.1:</u> $[i \notin N^{\tau^S}(\tilde{r}^S, r^S]$ and $i \notin N^{\tau^{S\cup\{i\}}}(r^{S\cup\{i\}}, r^S]]$
Note that this case occurs if $r^{S\cup\{i\}} \neq m$ and player i is not necessarily moved by the algorithm with respect to coalition $S\cup\{i\}$. Then, it follows from (6.21) together with the fact $c(\tilde{r}^S, \tau^S) = c(r^{S\cup\{i\}}, \tau^{S\cup\{i\}})$ that

$$\begin{aligned} & \alpha_{(S\cup\{i\})^{\tau^{S\cup\{i\}}}(r^{S\cup\{i\}}, r^S]} p_m - \alpha_m p_{N^{\tau^{S\cup\{i\}}}(r^{S\cup\{i\}}, r^S]} \\ = \; & \alpha_{S^{\tau^S}(\tilde{r}^S, r^S]} p_m - \alpha_m p_{N^{\tau^S}(\tilde{r}^S, r^S]} \overset{(6.20)}{>} 0. \end{aligned}$$

<u>Case A.2:</u> $[i \in N^{\tau^S}(\tilde{r}^S, r^S]$ and $i \notin N^{\tau^{S\cup\{i\}}}(r^{S\cup\{i\}}, r^S]]$
Note that this case occurs if $r^{S\cup\{i\}} = m$ and player i has been moved by the algorithm with respect to coalition $S \cup \{i\}$ such that $\tau^{S\cup\{i\}}(i) > \tau^{S\cup\{i\}}(r^S)$. Then, using the same arguments as in case A.1, we have

$$\begin{aligned} & \alpha_{(S\cup\{i\})^{\tau^{S\cup\{i\}}}(r^{S\cup\{i\}}, r^S]} p_m - \alpha_m p_{N^{\tau^{S\cup\{i\}}}(r^{S\cup\{i\}}, r^S]} \\ = \; & \alpha_{S^{\tau^S}(\tilde{r}^S, r^S]} p_m - \alpha_m p_{N^{\tau^S}(\tilde{r}^S, r^S]} + \alpha_m p_i \overset{(6.20)}{>} 0. \end{aligned}$$

<u>Case A.3:</u> $[i \in N^{\tau^S}(\tilde{r}^S, r^S]$ and $i \in N^{\tau^{S\cup\{i\}}}(r^{S\cup\{i\}}, r^S]]$
Note that this case occurs if $r^{S\cup\{i\}} = m$ and player i has either not been moved by the algorithm with respect to coalition $S \cup \{i\}$ or it has been moved such that $\tau^{S\cup\{i\}}(i) < \tau^{S\cup\{i\}}(r^S)$. Then, using the same arguments as in case A.1, we have

$$\begin{aligned} & \alpha_{(S\cup\{i\})^{\tau^{S\cup\{i\}}}(r^{S\cup\{i\}}, r^S]} p_m - \alpha_m p_{N^{\tau^{S\cup\{i\}}}(r^{S\cup\{i\}}, r^S]} \\ = \; & \alpha_{S^{\tau^S}(\tilde{r}^S, r^S]} p_m - \alpha_m p_{N^{\tau^S}(\tilde{r}^S, r^S]} + \alpha_i p_m \overset{(6.20)}{>} 0. \end{aligned}$$

<u>Case A.4:</u> $[i \notin N^{\tau^S}(\tilde{r}^S, r^S]$ and $i \in N^{\tau^{S\cup\{i\}}}(r^{S\cup\{i\}}, r^S]]$
Note that this case occurs if $r^{S\cup\{i\}} \neq m$ and player i has been moved by the algorithm

with respect to coalition $S \cup \{i\}$ such that $\tau^{S\cup\{i\}}(r^{S\cup\{i\}}) < \tau^{S\cup\{i\}}(i) < \tau^{S\cup\{i\}}(r^S)$. Then, using the same arguments as in case A.1, we have

$$\begin{aligned}&\alpha_{(S\cup\{i\})^{\tau^{S\cup\{i\}}}(r^{S\cup\{i\}},r^S]}p_m - \alpha_m p_{N^{\tau^{S\cup\{i\}}}(r^{S\cup\{i\}},r^S]} \\ = \ &\alpha_{S^{\tau^S}(\tilde{r}^S,r^S]}p_m - \alpha_m p_{N^{\tau^S}(\tilde{r}^S,r^S]} + \alpha_i p_m - \alpha_m p_i.\end{aligned}$$

If we can show that $\alpha_i p_m - \alpha_m p_i > 0$, (6.22) follows.

Let $\hat{r}^S$ be the direct predecessor of player i in $\tau^{S\cup\{i\}}$. Since player i is moved behind player $\hat{r}^S$ and not behind player r^S, we have due to the optimality of the algorithm that

$$\alpha_{(S\cup\{i\})^{\tau^{S\cup\{i\}}}(i,r^S]}p_i - \alpha_i p_{N^{\tau^{S\cup\{i\}}}(i,r^S]} \leq 0.$$

Consequently, it follows from (6.21) together with the fact $c(\hat{r}^S,\tau^S) = c(i,\tau^{S\cup\{i\}})$ that we also have

$$\alpha_{S^{\tau^S}(\hat{r}^S,r^S]}p_i - \alpha_i p_{N^{\tau^S}(\hat{r}^S,r^S]} \leq 0.$$

Therefore, together with (6.20), we can conclude $\frac{\alpha_i}{p_i} > \frac{\alpha_m}{p_m}$, i.e., $\alpha_i p_m - \alpha_m p_i > 0$.

<u>Case B:</u> $[\{k \in S \mid \sigma_0^S(m) < \sigma_0^S(k) < \sigma_0^S(i)\} \neq \emptyset]$
Hence, there are players of coalition S in between player m and player i in σ_0^S. Therefore, due to the definition of player m, it follows that for every $k \in S$ with $\sigma_0^S(m) < \sigma_0^S(k) < \sigma_0^S(i)$ we have

$$c(k,\sigma_{S\cup\{i\}}) \geq c(k,\sigma_S), \tag{6.23}$$

i.e., the statement in the lemma holds for all followers of player m with respect to σ_0^S intersected with $S \cap P(\sigma_0^S,i)$. First, note that $(S\cup\{i\})^{\sigma_0,\tau^S}_{c(r^{S\cup\{i\}},\tau^{S\cup\{i\}})}$ is non-empty because of the following. Due to requirement (i) and (ii) of admissibility we know that the first player in $(S\cup\{i\})^{\sigma_0,\tau^{S\cup\{i\}}}_{c(r^{S\cup\{i\}},\tau^{S\cup\{i\}})}$ with respect to $\tau^{S\cup\{i\}}$ is also the first player in $(S\cup\{i\})^{\sigma_0,\sigma_0^S}_{c(r^{S\cup\{i\}},\tau^{S\cup\{i\}})}$ with respect to σ_0^S. Therefore, using (6.23) we know that this player also belongs to $(S\cup\{i\})^{\sigma_0,\tau^S}_{c(r^{S\cup\{i\}},\tau^{S\cup\{i\}})}$. Hence, $(S\cup\{i\})^{\sigma_0,\tau^S}_{c(r^{S\cup\{i\}},\tau^{S\cup\{i\}})}$ is non-empty.

Next, define player $l \in S^{\tau^S}(\tilde{r}^S,r^S]$ as the player in $S^{\tau^S}(\tilde{r}^S,r^S]$ who is positioned last with respect to $\tau^{S\cup\{i\}}$, i.e., $\tau^{S\cup\{i\}}(l) \geq \tau^{S\cup\{i\}}(k)$ for all $k \in S^{\tau^S}(\tilde{r}^S,r^S]$. Note that player l was actually player r^S in case A because of (6.21). From the assumptions in (6.23) and (6.19) it follows that

$$c(l,\tau^{S\cup\{i\}}) \geq c(r^S,\tau^{S\cup\{i\}}) \geq c(r^S,\tau^S) > c(r^{S\cup\{i\}},\tau^{S\cup\{i\}}), \tag{6.24}$$

i.e., player l is to the right of player $r^{S\cup\{i\}}$ in $\tau^{S\cup\{i\}}$. We will prove that moving player m behind player l is more beneficial than moving player m behind player $r^{S\cup\{i\}}$, i.e., we will prove that

$$\alpha_{(S\cup\{i\})^{\tau^{S\cup\{i\}}}(r^{S\cup\{i\}},l]}p_m - \alpha_m p_{N^{\tau^{S\cup\{i\}}}(r^{S\cup\{i\}},l]} > 0. \tag{6.25}$$

This implies that the step made by the algorithm for player m when applied on coalition $S \cup \{i\}$ is not optimal, which contradicts the optimality of the algorithm. Hence, for case B, it remains to prove (6.25).

From the definitions of players $\tilde{r}^S$ and l, together with (6.23), it follows that

$$S^{\tau^S}(\tilde{r}^S, r^S] \subseteq S^{\tau^{S\cup\{i\}}}(r^{S\cup\{i\}}, l] \subseteq (S\cup\{i\})^{\tau^{S\cup\{i\}}}(r^{S\cup\{i\}}, l]. \tag{6.26}$$

Moreover, since during the run of the algorithm player m jumps over all players in $S^{\tau^S}(m, r^S]$, Lemma 6.4.1 implies that

$$u_k > u_m, \tag{6.27}$$

for all $k \in S^{\tau^S}(m, r^S]$.

Below we distinguish between the following four cases:

- case B.1: $i \notin N^{\tau^S}(\tilde{r}^S, r^S]$ and $i \notin N^{\tau^{S\cup\{i\}}}(r^{S\cup\{i\}}, l]$,
- case B.2: $i \in N^{\tau^S}(\tilde{r}^S, r^S]$ and $i \notin N^{\tau^{S\cup\{i\}}}(r^{S\cup\{i\}}, l]$,
- case B.3: $i \in N^{\tau^S}(\tilde{r}^S, r^S]$ and $i \in N^{\tau^{S\cup\{i\}}}(r^{S\cup\{i\}}, l]$,
- case B.4: $i \notin N^{\tau^S}(\tilde{r}^S, r^S]$ and $i \in N^{\tau^{S\cup\{i\}}}(r^{S\cup\{i\}}, l]$.

Case B.1: [$i \notin N^{\tau^S}(\tilde{r}^S, r^S]$ and $i \notin N^{\tau^{S\cup\{i\}}}(r^{S\cup\{i\}}, l]$]
Using $c(l, \tau^{S\cup\{i\}}) \geq c(r^S, \tau^S)$ from (6.24), we distinguish between another two cases:

- case B.1(i): $c(l, \tau^{S\cup\{i\}}) = c(r^S, \tau^S)$,
- case B.1(ii): $c(l, \tau^{S\cup\{i\}}) > c(r^S, \tau^S)$.

Case B.1(i): [$c(l, \tau^{S\cup\{i\}}) = c(r^S, \tau^S)$]

Since $\tilde{r}^S \in (S \cup \{i\})^{\sigma_0,\tau^S}_{c(r^{S\cup\{i\}},\tau^{S\cup\{i\}})}$ and thus $c(\tilde{r}^S,\tau^S) = c(r^{S\cup\{i\}},\tau^{S\cup\{i\}})$, the assumptions $i \notin N^{\tau^S}(\tilde{r}^S, r^S]$ and $c(l,\tau^{S\cup\{i\}}) = c(r^S,\tau^S)$ imply that

$$\overline{S}^{\tau^S}(\tilde{r}^S, r^S] = \overline{S}^{\tau^{S\cup\{i\}}}(r^{S\cup\{i\}}, l] = (\overline{S \cup \{i\}})^{\tau^{S\cup\{i\}}}(r^{S\cup\{i\}}, l]. \tag{6.28}$$

For every $k \in S^{\tau^{S\cup\{i\}}}(r^{S\cup\{i\}}, l]\backslash S^{\tau^S}(\tilde{r}^S, r^S]$ we know that $k \in S^{\tau^S}(m, \tilde{r}^S]$, because $c(k,\tau^S) \leq c(k,\tau^{S\cup\{i\}})$ by (6.23) and $P(\tau^S,m) = P(\tau^{S\cup\{i\}},m)$. Hence, also $k \in S^{\tau^S}(m, r^S]$ and thus from (6.27) it follows that $u_k > u_m$. Together with Proposition 6.5.1 applied on the set $S^{\tau^{S\cup\{i\}}}(r^{S\cup\{i\}}, l]\backslash S^{\tau^S}(\tilde{r}^S, r^S]$ and player m we have

$$\begin{aligned} & \alpha_{(S\cup\{i\})^{\tau^{S\cup\{i\}}}(r^{S\cup\{i\}},l]\backslash S^{\tau^S}(\tilde{r}^S,r^S]}p_m - \alpha_m p_{(S\cup\{i\})^{\tau^{S\cup\{i\}}}(r^{S\cup\{i\}},l]\backslash S^{\tau^S}(\tilde{r}^S,r^S]} \\ = \; & \alpha_{S^{\tau^{S\cup\{i\}}}(r^{S\cup\{i\}},l]\backslash S^{\tau^S}(\tilde{r}^S,r^S]}p_m - \alpha_m p_{S^{\tau^{S\cup\{i\}}}(r^{S\cup\{i\}},l]\backslash S^{\tau^S}(\tilde{r}^S,r^S]} > 0, \end{aligned} \tag{6.29}$$

where the equality follows from the assumption $i \notin N^{\tau^{S\cup\{i\}}}(r^{S\cup\{i\}}, l]$. As a consequence,

$$\begin{aligned} & \alpha_{(S\cup\{i\})^{\tau^{S\cup\{i\}}}(r^{S\cup\{i\}},l]}p_m - \alpha_m p_{N^{\tau^{S\cup\{i\}}}(r^{S\cup\{i\}},l]} \\ = \; & \alpha_{(S\cup\{i\})^{\tau^{S\cup\{i\}}}(r^{S\cup\{i\}},l]}p_m - \alpha_m \left(p_{(S\cup\{i\})^{\tau^{S\cup\{i\}}}(r^{S\cup\{i\}},l]} + p_{(\overline{S\cup\{i\}})^{\tau^{S\cup\{i\}}}(r^{S\cup\{i\}},l]}\right) \\ \overset{(6.28)}{=} \; & \alpha_{(S\cup\{i\})^{\tau^{S\cup\{i\}}}(r^{S\cup\{i\}},l]}p_m - \alpha_m \left(p_{(S\cup\{i\})^{\tau^{S\cup\{i\}}}(r^{S\cup\{i\}},l]} + p_{\overline{S}^{\tau^S}(\tilde{r}^S,r^S]}\right) \\ \overset{(6.26)}{=} \; & \alpha_{S^{\tau^S}(\tilde{r}^S,r^S]}p_m - \alpha_m \left(p_{S^{\tau^S}(\tilde{r}^S,r^S]} + p_{\overline{S}^{\tau^S}(\tilde{r}^S,r^S]}\right) \\ & + \alpha_{(S\cup\{i\})^{\tau^{S\cup\{i\}}}(r^{S\cup\{i\}},l]\backslash S^{\tau^S}(\tilde{r}^S,r^S]}p_m - \alpha_m p_{(S\cup\{i\})^{\tau^{S\cup\{i\}}}(r^{S\cup\{i\}},l]\backslash S^{\tau^S}(\tilde{r}^S,r^S]} \\ \overset{(6.29)}{\geq} \; & \alpha_{S^{\tau^S}(\tilde{r}^S,r^S]}p_m - \alpha_m \left(p_{S^{\tau^S}(\tilde{r}^S,r^S]} + p_{\overline{S}^{\tau^S}(\tilde{r}^S,r^S]}\right) \\ \overset{(6.20)}{>} \; & 0, \end{aligned}$$

proving (6.25).

<u>Case B.1(ii):</u> $[c(l,\tau^{S\cup\{i\}}) > c(r^S,\tau^S)]$
Define Q as the set of players from $S^{\tau^S}(\tilde{r}^S, r^S]$ who are positioned in $\tau^{S\cup\{i\}}$ in a component to the right of component $(S \cup \{i\})^{\sigma_0,\tau^{S\cup\{i\}}}_{c(r^S,\tau^S)}$. Since $c(l,\tau^{S\cup\{i\}}) > c(r^S,\tau^S)$ and $l \in S^{\tau^S}(\tilde{r}^S, r^S]$, we have $l \in Q$ and thus $Q \neq \emptyset$. Select from Q only the players who have in their corresponding component in $\tau^{S\cup\{i\}}$ no players from $S^{\tau^S}(\tilde{r}^S, r^S]$ in front of him and denote this set of players by $\mathcal{Q}$, i.e., we select in each component a player of Q (if possible) that is most to the left in $\tau^{S\cup\{i\}}$.

Set $\mathcal{Q} = \{q_1, q_2, \ldots, q_{|\mathcal{Q}|}\}$ such that $c(q_k,\tau^{S\cup\{i\}}) < c(q_{k+1},\tau^{S\cup\{i\}})$ for all $k \in \{1,\ldots,|\mathcal{Q}|-1\}$. Define w_1 as the first player in $\overline{(S \cup \{i\})}^{\sigma_0}_{c(r^S,\tau^S)}$, i.e., the direct follower in $\tau^{S\cup\{i\}}$ of component $(S\cup\{i\})^{\sigma_0,\tau^{S\cup\{i\}}}_{c(r^S,\tau^S)}$. For $k\in\{2,\dots,|\mathcal{Q}|\}$, define w_k as the first player in $\overline{(S\cup\{i\})}^{\sigma_0}_{c(q_{k-1},\tau^{S\cup\{i\}})}$, i.e., the direct follower in $\tau^{S\cup\{i\}}$ of component $(S\cup\{i\})^{\sigma_0,\tau^{S\cup\{i\}}}_{c(q_{k-1},\tau^{S\cup\{i\}})}$. Note that because of the definition of player l we have $c(q_{|\mathcal{Q}|},\tau^{S\cup\{i\}})=c(l,\tau^{S\cup\{i\}})$.

The collection of sets $\{N^{\tau^{S\cup\{i\}}}[w_k,q_k)\mid k\in\{1,\dots,|\mathcal{Q}|\}\}$ are by definition mutually disjoint. Moreover, for $k\in\{1,\dots,|\mathcal{Q}|\}$, we have $N^{\tau^{S\cup\{i\}}}[w_k,q_k)\cap S^{\tau^S}(\tilde{r}^S,r^S]=\emptyset$ and $(S\cup\{i\})^{\tau^{S\cup\{i\}}}[w_k,q_k)\subseteq(S\cup\{i\})^{\tau^{S\cup\{i\}}}(r^{S\cup\{i\}},l]$. If we set $R=S^{\tau^S}(\tilde{r}^S,r^S]\cup\bigcup_{k=1}^{|\mathcal{Q}|}(S\cup\{i\})^{\tau^{S\cup\{i\}}}[w_k,q_k)$, then using (6.26) we have

$$R\subseteq(S\cup\{i\})^{\tau^{S\cup\{i\}}}(r^{S\cup\{i\}},l],\tag{6.30}$$

and

$$\alpha_R=\alpha_{S^{\tau^S}(\tilde{r}^S,r^S]}+\sum_{k=1}^{|\mathcal{Q}|}\alpha_{(S\cup\{i\})^{\tau^{S\cup\{i\}}}[w_k,q_k)},\tag{6.31}$$

and

$$p_R=p_{S^{\tau^S}(\tilde{r}^S,r^S]}+\sum_{k=1}^{|\mathcal{Q}|}p_{(S\cup\{i\})^{\tau^{S\cup\{i\}}}[w_k,q_k)}.\tag{6.32}$$

Note, for $k\in\{1,\dots,|\mathcal{Q}|\}$, we have $c(q_k,\sigma_0)\leq c(q_k,\tau^S)\leq c(r^S,\tau^S)$. Hence, as w_k is in $\tau^{S\cup\{i\}}$ to the right of component $(S\cup\{i\})^{\sigma_0,\tau^{S\cup\{i\}}}_{c(r^S,\tau^S)}$, we have that it is admissible to more player q_k in front of w_k with respect to $\tau^{S\cup\{i\}}$. In other words, it is admissible to move player q_k to the tail of component $(S\cup\{i\})^{\sigma_0,\tau^{S\cup\{i\}}}_{c(q_{k-1},\tau^{S\cup\{i\}})}$ (and the tail of component $(S\cup\{i\})^{\sigma_0,\tau^{S\cup\{i\}}}_{c(r^S,\tau^S)}$ in case $k=1$). Since player q_k is not moved behind player w_k, we have due to property (v) of the algorithm that

$$\alpha_{(S\cup\{i\})^{\tau^{S\cup\{i\}}}[w_k,q_k)}p_{q_k}-\alpha_{q_k}p_{N^{\tau^{S\cup\{i\}}}[w_k,q_k)}>0.$$

Moreover, since $q_k\in S^{\tau^S}(m,r^S]$ it follows from (6.27) that $u_m<u_{q_k}$. As a consequence,

$$\alpha_{(S\cup\{i\})^{\tau^{S\cup\{i\}}}[w_k,q_k)}p_m-\alpha_m p_{N^{\tau^{S\cup\{i\}}}[w_k,q_k)}>0.\tag{6.33}$$

We have

$$
\begin{aligned}
&\alpha_R p_m - \alpha_m \left(p_R + p_{\overline{(S\cup\{i\})}^{\tau^{S\cup\{i\}}}(r^{S\cup\{i\}},l]} \right) \\
\overset{(6.31),(6.32)}{=} \;& \left(\alpha_{S^{\tau^S}(\tilde{r}^S, r^S]} + \sum_{k=1}^{|\mathcal{Q}|} \alpha_{(S\cup\{i\})^{\tau^{S\cup\{i\}}}[w_k,q_k)} \right) p_m \\
& - \alpha_m \left(p_{S^{\tau^S}(\tilde{r}^S, r^S]} + \sum_{k=1}^{|\mathcal{Q}|} p_{(S\cup\{i\})^{\tau^{S\cup\{i\}}}[w_k,q_k)} \right. \\
& \left. + p_{\overline{(S\cup\{i\})}^{\tau^{S\cup\{i\}}}(r^{S\cup\{i\}},w_1)} + p_{\overline{(S\cup\{i\})}^{\tau^{S\cup\{i\}}}[w_1,l]} \right) \\
\overset{\text{see below}}{=} \;& \left(\alpha_{S^{\tau^S}(\tilde{r}^S, r^S]} + \sum_{k=1}^{|\mathcal{Q}|} \alpha_{(S\cup\{i\})^{\tau^{S\cup\{i\}}}[w_k,q_k)} \right) p_m \\
& - \alpha_m \left(p_{S^{\tau^S}(\tilde{r}^S, r^S]} + \sum_{k=1}^{|\mathcal{Q}|} p_{(S\cup\{i\})^{\tau^{S\cup\{i\}}}[w_k,q_k)} \right. \\
& \left. + p_{\overline{S}^{\tau^S}(\tilde{r}^S, r^S]} + \sum_{k=1}^{|\mathcal{Q}|} p_{\overline{(S\cup\{i\})}^{\tau^{S\cup\{i\}}}[w_k,q_k)} \right) \\
= \;& \left(\alpha_{S^{\tau^S}(\tilde{r}^S, r^S]} + \sum_{k=1}^{|\mathcal{Q}|} \alpha_{(S\cup\{i\})^{\tau^{S\cup\{i\}}}[w_k,q_k)} \right) p_m \\
& - \alpha_m \left(p_{N^{\tau^S}(\tilde{r}^S, r^S]} + \sum_{k=1}^{|\mathcal{Q}|} p_{N^{\tau^{S\cup\{i\}}}[w_k,q_k)} \right) \\
\overset{(6.20),(6.33)}{>} \;& 0. \qquad (6.34)
\end{aligned}
$$

Note that the second equality follows from $c(\tilde{r}^S, \tau^S) = c(r^{S\cup\{i\}}, \tau^{S\cup\{i\}})$, the fact that w_1 is the direct follower in $\tau^{S\cup\{i\}}$ of component $(S \cup \{i\})^{\sigma_0, \tau^{S\cup\{i\}}}_{c(r^S, \tau^S)}$, and the assumption $i \notin N^{\tau^S}(\tilde{r}^S, r^S]$. Moreover, the collection of sets $\{\overline{(S \cup \{i\})}^{\tau^{S\cup\{i\}}}[w_k, q_k) \mid k \in \{1, \ldots, |\mathcal{Q}|\}\}$ forms a partition of the set $\overline{(S \cup \{i\})}^{\tau^{S\cup\{i\}}}[w_1, l]$.

For every $k \in S^{\tau^{S\cup\{i\}}}(r^{S\cup\{i\}}, l]\backslash R$, we know that either $k \in S^{\tau^S}(m, \tilde{r}^S]$ or $c(k, \tau^S) \geq c(r^S, \tau^S)$. If $k \in S^{\tau^S}(m, \tilde{r}^S]$, then also $k \in S^{\tau^S}(m, r^S]$ and thus from (6.27) we have $u_k > u_m$. Next, if $c(k, \tau^S) \geq c(r^S, \tau^S)$, then because $c(l, \tau^S) \leq c(r^S, \tau^S)$ we know that the swap of players k and l with respect to $\tau^{S\cup\{i\}}$ is admissible. Therefore, according to Lemma 6.4.1, we know $u_k \geq u_l$. Moreover, since $l \in S^{\tau^S}(m, r^S]$, it follows from (6.27) that $u_m < u_l$. As a consequence, $u_m < u_k$. Together with

Proposition 6.5.1 applied on the set $S^{\tau^{S\cup\{i\}}}(r^{S\cup\{i\}},l]\backslash R$ and player m we have

$$\begin{aligned} & \alpha_{(S\cup\{i\})^{\tau^{S\cup\{i\}}}(r^{S\cup\{i\}},l]\backslash R}p_m - \alpha_m p_{(S\cup\{i\})^{\tau^{S\cup\{i\}}}(r^{S\cup\{i\}},l]\backslash R} \\ = \; & \alpha_{S^{\tau^{S\cup\{i\}}}(r^{S\cup\{i\}},l]\backslash R}p_m - \alpha_m p_{S^{\tau^{S\cup\{i\}}}(r^{S\cup\{i\}},l]\backslash R} > 0, \end{aligned} \tag{6.35}$$

where the equality follows from the assumption $i \notin N^{\tau^{S\cup\{i\}}}(r^{S\cup\{i\}},l]$. As a consequence,

$$\begin{aligned} & \alpha_{(S\cup\{i\})^{\tau^{S\cup\{i\}}}(r^{S\cup\{i\}},l]}p_m - \alpha_m p_{N^{\tau^{S\cup\{i\}}}(r^{S\cup\{i\}},l]} \\ = \; & \alpha_{(S\cup\{i\})^{\tau^{S\cup\{i\}}}(r^{S\cup\{i\}},l]}p_m - \alpha_m \left(p_{(S\cup\{i\})^{\tau^{S\cup\{i\}}}(r^{S\cup\{i\}},l]} + p_{\overline{(S\cup\{i\})}^{\tau^{S\cup\{i\}}}(r^{S\cup\{i\}},l]}\right) \\ \overset{(6.30)}{=} \; & \alpha_R p_m - \alpha_m \left(p_R + p_{\overline{(S\cup\{i\})}^{\tau^{S\cup\{i\}}}(r^{S\cup\{i\}},l]}\right) \\ & + \alpha_{(S\cup\{i\})^{\tau^{S\cup\{i\}}}(r^{S\cup\{i\}},l]\backslash R}p_m - \alpha_m p_{(S\cup\{i\})^{\tau^{S\cup\{i\}}}(r^{S\cup\{i\}},l]\backslash R} \\ \overset{(6.34),(6.35)}{>} \; & 0, \end{aligned}$$

proving (6.25).

<u>Case B.2:</u> $[i \in N^{\tau^S}(\tilde{r}^S, r^S]$ and $i \notin N^{\tau^{S\cup\{i\}}}(r^{S\cup\{i\}},l]]$
Note that this case occurs exactly if $c(\tilde{r}^S,\tau^S) < c(i,\sigma_0) < c(r^S,\tau^S)$ and player i has been moved by the algorithm with respect to coalition $S\cup\{i\}$ such that $\tau^{S\cup\{i\}}(i) > \tau^{S\cup\{i\}}(l)$. We have

$$\alpha_{S^{\tau^S}(\tilde{r}^S,r^S]}p_m - \alpha_m p_{N^{\tau^S}(\tilde{r}^S,r^S]\backslash\{i\}} = \alpha_{S^{\tau^S}(\tilde{r}^S,r^S]}p_m - \alpha_m p_{N^{\tau^S}(\tilde{r}^S,r^S]} + \alpha_m p_i \overset{(6.20)}{>} 0. \tag{6.36}$$

Then, using the same arguments as in case B.1, we can prove (6.25). Namely, where we used (6.20) in case B.1, we now use the above equation. Hence, player i has already been taken into account and thus for using the same arguments as in case B.1 we can assume $i \notin N^{\tau^S}(\tilde{r}^S, r^S]$ and $i \notin N^{\tau^{S\cup\{i\}}}(r^{S\cup\{i\}},l]$.

For example, analogous to case B.1(i), case B.2(i) goes as follows. Since $\tilde{r}^S \in (S\cup \{i\})^{\sigma_0,\tau^S}_{c(r^{S\cup\{i\}},\tau^{S\cup\{i\}})}$ and thus $c(\tilde{r}^S,\tau^S) = c(r^{S\cup\{i\}},\tau^{S\cup\{i\}})$, the assumption $c(l,\tau^{S\cup\{i\}}) = c(r^S,\tau^S)$ implies that

$$\overline{S}^{\tau^S}(\tilde{r}^S, r^S]\backslash\{i\} = \overline{S}^{\tau^{S\cup\{i\}}}(r^{S\cup\{i\}},l] = \overline{(S\cup\{i\})}^{\tau^{S\cup\{i\}}}(r^{S\cup\{i\}},l]. \tag{6.37}$$

Using exactly the same arguments as in case B.1(i) we have

$$\alpha_{(S\cup\{i\})^{\tau^{S\cup\{i\}}}(r^{S\cup\{i\}},l]\backslash S^{\tau^S}(\tilde{r}^S,r^S]}p_m - \alpha_m p_{(S\cup\{i\})^{\tau^{S\cup\{i\}}}(r^{S\cup\{i\}},l]\backslash S^{\tau^S}(\tilde{r}^S,r^S]} > 0, \tag{6.38}$$

As a consequence,

$$
\begin{aligned}
& \alpha_{(S\cup\{i\})^{\tau^{S\cup\{i\}}}(r^{S\cup\{i\}},l]}p_m - \alpha_m p_{N^{\tau^{S\cup\{i\}}}(r^{S\cup\{i\}},l]} \\
= \; & \alpha_{(S\cup\{i\})^{\tau^{S\cup\{i\}}}(r^{S\cup\{i\}},l]}p_m - \alpha_m \left(p_{(S\cup\{i\})^{\tau^{S\cup\{i\}}}(r^{S\cup\{i\}},l]} + p_{\overline{(S\cup\{i\})}^{\tau^{S\cup\{i\}}}(r^{S\cup\{i\}},l]}\right) \\
\overset{(6.37)}{=} \; & \alpha_{(S\cup\{i\})^{\tau^{S\cup\{i\}}}(r^{S\cup\{i\}},l]}p_m - \alpha_m \left(p_{(S\cup\{i\})^{\tau^{S\cup\{i\}}}(r^{S\cup\{i\}},l]} + p_{\overline{S}^{\tau^S}(\tilde{r}^S,r^S]\backslash\{i\}}\right) \\
\overset{(6.26)}{=} \; & \alpha_{S^{\tau^S}(\tilde{r}^S,r^S]}p_m - \alpha_m \left(p_{S^{\tau^S}(\tilde{r}^S,r^S]} + p_{\overline{S}^{\tau^S}(\tilde{r}^S,r^S]\backslash\{i\}}\right) \\
& + \alpha_{(S\cup\{i\})^{\tau^{S\cup\{i\}}}(r^{S\cup\{i\}},l]\backslash S^{\tau^S}(\tilde{r}^S,r^S]}p_m - \alpha_m p_{(S\cup\{i\})^{\tau^{S\cup\{i\}}}(r^{S\cup\{i\}},l]\backslash S^{\tau^S}(\tilde{r}^S,r^S]} \\
\overset{(6.38)}{\geq} \; & \alpha_{S^{\tau^S}(\tilde{r}^S,r^S]}p_m - \alpha_m \left(p_{S^{\tau^S}(\tilde{r}^S,r^S]} + p_{\overline{S}^{\tau^S}(\tilde{r}^S,r^S]\backslash\{i\}}\right) \\
= \; & \alpha_{S^{\tau^S}(\tilde{r}^S,r^S]}p_m - \alpha_m p_{N^{\tau^S}(\tilde{r}^S,r^S]\backslash\{i\}} \\
\overset{(6.36)}{>} \; & 0,
\end{aligned}
$$

proving (6.25).

<u>Case B.3:</u> $[i \in N^{\tau^S}(\tilde{r}^S, r^S]$ and $i \in N^{\tau^{S\cup\{i\}}}(r^{S\cup\{i\}}, l]]$
Note that this case occurs exactly if $c(\tilde{r}^S, \tau^S) < c(i, \sigma_0) < c(r^S, \tau^S)$ and player i has either not been moved by the algorithm with respect to coalition $S \cup \{i\}$ or it has been moved such that $\tau^{S\cup\{i\}}(i) < \tau^{S\cup\{i\}}(l)$. We have

$$\alpha_{S^{\tau^S}(\tilde{r}^S,r^S]\cup\{i\}}p_m - \alpha_m p_{N^{\tau^S}(\tilde{r}^S,r^S]} = \alpha_{S^{\tau^S}(\tilde{r}^S,r^S]}p_m - \alpha_m p_{N^{\tau^S}(\tilde{r}^S,r^S]} + \alpha_i p_m \overset{(6.20)}{>} 0.$$

Then, using the same arguments as in case B.1, we can prove (6.25).

<u>Case B.4:</u> $[i \notin N^{\tau^S}(\tilde{r}^S, r^S]$ and $i \in N^{\tau^{S\cup\{i\}}}(r^{S\cup\{i\}}, l]]$
Then,

$$\alpha_{S^{\tau^S}(\tilde{r}^S,r^S]\cup\{i\}}p_m - \alpha_m p_{N^{\tau^S}(\tilde{r}^S,r^S]\cup\{i\}} = \alpha_{S^{\tau^S}(\tilde{r}^S,r^S]}p_m - \alpha_m p_{N^{\tau^S}(\tilde{r}^S,r^S]} + \alpha_i p_m - \alpha_m p_i.$$

It suffices to show that $\alpha_i p_m - \alpha_m p_i > 0$. Together with the above equation and (6.20), we can prove (6.25) by using the same arguments as in case B.1.

We distinguish between two cases:

- case B.4(i): $c(i, \sigma_0) > c(r^S, \tau^S)$,
- case B.4(ii): $c(i, \sigma_0) < c(\tilde{r}^S, \tau^S)$.

Case B.4(i): $[c(i,\sigma_0) > c(r^S,\tau^S)]$
Note that this case occurs exactly if $\tau^{S\cup\{i\}}(i) < \tau^{S\cup\{i\}}(l)$. This means that player i is not necessarily moved by the algorithm with respect to coalition $S \cup \{i\}$. Then,

$$c(l,\sigma_0) \leq c(l,\tau^S) \leq c(r^S,\tau^S) < c(i,\sigma_0) \leq c(i,\tau^{S\cup\{i\}}) = c(i,\sigma_{S\cup\{i\}}).$$

Hence, the swap of players i and l with respect to $\sigma_{S\cup\{i\}}$ is admissible and thus, according to Lemma 6.4.1, we have $u_i \geq u_l$. Consequently, together with (6.27), we have $u_i \geq u_l > u_m$. Hence, $\alpha_i p_m - \alpha_m p_i > 0$.

Case B.4(ii): $[c(i,\sigma_0) < c(\tilde{r}^S,\tau^S)]$
Note that this case occurs exactly if player i has been moved by the algorithm with respect to coalition $S \cup \{i\}$ such that $\tau^{S\cup\{i\}}(r^{S\cup\{i\}}) < \tau^{S\cup\{i\}}(i) < \tau^{S\cup\{i\}}(l)$. Suppose $c(l,\sigma_0) < c(i,\sigma_0)$, then

$$c(l,\sigma_0) < c(i,\sigma_0) < c(\tilde{r}^S,\tau^S) = c(r^{S\cup\{i\}},\tau^{S\cup\{i\}}) = c(m,\sigma_{S\cup\{i\}}).$$

Hence, the swap of players m and l with respect to $\sigma_{S\cup\{i\}}$ is admissible and thus, according to Lemma 6.4.1, we have $u_m \geq u_l$. This is a contradiction with (6.27) and thus we know $c(l,\sigma_0) > c(i,\sigma_0)$. Therefore, using (6.18), we have $c(l,\tau^S) = c(l,\tau^{S\cup\{i\}})$ which implies $l = r^S$. As a consequence, if $k \in S^{\tau^{S\cup\{i\}}}(r^{S\cup\{i\}}, r^S]$, then also $k \in S^{\tau^S}(m, r^S]$ and thus from (6.27) we have $u_m < u_k$. With the same arguments we used for player l we can conclude $c(k,\sigma_0) > c(i,\sigma_0)$ and thus $c(k,\tau^S) = c(k,\tau^{S\cup\{i\}})$. Therefore, we have

$$N^{\tau^S}(\tilde{r}^S, r^S] \cup \{i\} = N^{\tau^{S\cup\{i\}}}(r^{S\cup\{i\}}, r^S], \tag{6.39}$$

and

$$c(k,\tau^S) = c(k,\tau^{S\cup\{i\}}), \tag{6.40}$$

for all $k \in S^{\tau^S}(\tilde{r}^S, r^S]$.

Let $\hat{r}^S$ be the direct predecessor of player i in $\tau^{S\cup\{i\}}$. Since player i is in $\tau^{S\cup\{i\}}$ behind player $\hat{r}^S$ and not behind player r^S, although this is an admissible swap, we have

$$\alpha_{(S\cup\{i\})^{\tau^{S\cup\{i\}}}(i,r^S]} p_i - \alpha_i p_{N^{\tau^{S\cup\{i\}}}(i,r^S]} \leq 0.$$

Consequently, it follows from (6.39) and (6.40) together with the fact that $c(\hat{r}^S, \tau^S) = c(i, \tau^{S\cup\{i\}})$, that

$$\alpha_{S^{\tau^S}(\hat{r}^S, r^S]} p_i - \alpha_i p_{N^{\tau^S}(\hat{r}^S, r^S]} \leq 0.$$

Moreover, since player m is moved behind player r^S and not behind player $\hat{r}^S$, we have due to property (v) of the algorithm that

$$\alpha_{S^{\tau^S}(\hat{r}^S, r^S]} p_m - \alpha_m p_{N^{\tau^S}(\hat{r}^S, r^S]} > 0.$$

Therefore, we can conclude $\frac{\alpha_i}{p_i} > \frac{\alpha_m}{p_m}$, i.e., $\alpha_i p_m - \alpha_m p_i > 0$, which is exactly what we needed.

Proof of Lemma 6.5.6

In this proof we denote $c(k, S \cup \{i\}, \sigma)$ by $c(k, \sigma)$ for every $k \in S \cup \{i\}$ and every $\sigma \in \Pi(N)$. We continue the proof by means of induction on the number

$$n_m = \{j \in S \mid \sigma_S(m) < \sigma_S(j) < \sigma_S(i)\},$$

i.e., the number of players in coalition S between player m and player i in σ_S.

Base step: If $n_m = 0$, then player m is among all players in S the closest predecessor of player i with respect to σ_S. Then,

$$\{l \in S \cap F(\tau_m, m) \mid c(l, \tau_m) < c(i, \sigma_0)\} = \emptyset,$$

and thus (6.10) is true.

Induction step: Assume that (6.10) holds for every $k \in S \cap P(\sigma_S, i)$ with $n_k < n_m$. Since the followers of player m with respect to σ_S intersected with $S \cap P(\sigma_S, i)$ are exactly the players with $n_k < n_m$, we actually assume that for every $k \in S \cap P(\sigma_S, i) \cap F(\sigma_S, m)$ and $r \in S \cap F(\tau_k, k)$ with $c(r, \tau_k) < c(i, \sigma_0)$ we have

$$\alpha_{(S\cup\{i\})^{\tau_k}(k,r]} p_k - \alpha_k p_{N^{\tau_k}(k,r]} \leq 0, \tag{6.41}$$

We distinguish between the following cases:

– $u_m \leq u_k$ for all $k \in S^{\sigma_S}(m, l]$,

– there exists a player $k \in S^{\sigma_S}(m,l]$ with $u_m > u_k$.

<u>Case 1:</u> $[u_m \leq u_k$ for all $k \in S^{\sigma_S}(m,l]]$
We will show that since moving player m behind player l in σ_S is not beneficial, also moving player m behind player l in τ_m is not beneficial.

Note that because of the induction assumption in (6.41), we know that all players in $(S \cup \{i\})^{\tau_m}(m,l]$ have not been moved by the algorithm and thus $N^{\tau_m}(m,l] \subseteq N^{\sigma_S}(m,l]$. Moreover, since player l is in both σ_S and τ_m to the left of the original component of player i in σ_0, we have

$$(S \cup \{i\})^{\tau_m}(m,l] \subseteq S^{\sigma_S}(m,l], \tag{6.42}$$

and

$$(\overline{S \cup \{i\}})^{\tau_m}(m,l] = \overline{S}^{\sigma_S}(m,l]. \tag{6.43}$$

Note, as processing order σ_S is optimal for coalition S, we have

$$\alpha_{S^{\sigma_S}(m,l]}p_m - \alpha_m p_{N^{\sigma_S}(m,l]} \leq 0. \tag{6.44}$$

Since $u_m \leq u_k$ for all $k \in S^{\sigma_S}(m,l]$, it follows from Proposition 6.5.1 applied on the set $S^{\sigma_S}(m,l]$ and player m together with (6.42) that

$$\alpha_{S^{\sigma_S}(m,l] \setminus (S \cup \{i\})^{\tau_m}(m,l]}p_m - \alpha_m p_{S^{\sigma_S}(m,l] \setminus (S \cup \{i\})^{\tau_m}(m,l]} \geq 0, \tag{6.45}$$

where there is an equality if $S^{\sigma_S}(m,l] = (S \cup \{i\})^{\tau_m}(m,l]$. As a consequence, we have

$$\begin{aligned}
& \alpha_{(S\cup\{i\})^{\tau_m}(m,l]}p_m - \alpha_m p_{N^{\tau_m}(m,l]} \\
= \; & \alpha_{(S\cup\{i\})^{\tau_m}(m,l]}p_m - \alpha_m \left(p_{(S\cup\{i\})^{\tau_m}(m,l]} + p_{\overline{(S\cup\{i\})}^{\tau_m}(m,l]}\right) \\
\overset{(6.42),(6.43)}{=} \; & \alpha_{S^{\sigma_S}(m,l]}p_m - \alpha_m \left(p_{S^{\sigma_S}(m,l]} + p_{\overline{S}^{\sigma_S}(m,l]}\right) \\
& - \left(\alpha_{S^{\sigma_S}(m,l]\setminus(S\cup\{i\})^{\tau_m}(m,l]}p_m - \alpha_m p_{S^{\sigma_S}(m,l]\setminus(S\cup\{i\})^{\tau_m}(m,l]}\right) \\
= \; & \alpha_{S^{\sigma_S}(m,l]}p_m - \alpha_m p_{N^{\sigma_S}(m,l]} \\
& - \left(\alpha_{S^{\sigma_S}(m,l]\setminus(S\cup\{i\})^{\tau_m}(m,l]}p_m - \alpha_m p_{S^{\sigma_S}(m,l]\setminus(S\cup\{i\})^{\tau_m}(m,l]}\right) \\
\overset{(6.44),(6.45)}{\leq} \; & 0,
\end{aligned}$$

and (6.10) follows.

Case 2: [there exists a player $k \in S^{\sigma_S}(m,l]$ with $u_m > u_k$]
Let player $q \in S^{\sigma_S}(m,l]$ be the closest follower of player m in σ_S with a smaller urgency than player m: $u_q < u_m \leq u_k$ for all $k \in S^{\sigma_S}(m,q)$. Since $q \in S \cap P(\sigma_S, i) \cap F(\sigma_S, m)$, it follows from the induction assumption in (6.41) that

$$\alpha_{(S\cup\{i\})^{\tau_q}(q,l]} p_q - \alpha_q p_{N^{\tau_q}(q,l]} \leq 0.$$

As a consequence, since $u_q < u_m$, we also have

$$\alpha_{(S\cup\{i\})^{\tau_q}(q,l]} p_m - \alpha_m p_{N^{\tau_q}(q,l]} < 0. \tag{6.46}$$

We distinguish between two cases:

- case 2(i): $S^{\sigma_S}(m,q) \cap S^{\tau_m}(m,l] = \emptyset$, i.e., all players in $S^{\sigma_S}(m,q)$ have been moved by the algorithm,
- case 2(ii): $S^{\sigma_S}(m,q) \cap S^{\tau_m}(m,l] \neq \emptyset$, i.e., not all players in $S^{\sigma_S}(m,q)$ have been moved by the algorithm.

Case 2(i): $[S^{\sigma_S}(m,q) \cap S^{\tau_m}(m,l] = \emptyset]$
We will show that since moving player q behind player l in τ_q is not beneficial, also moving player m behind player l in τ_m is not beneficial.

Since player m is a predecessor of player q in σ_S, we have

$$(\overline{S\cup\{i\}})^{\tau_q}(q,l] = (\overline{S\cup\{i\}})^{\sigma_S}(q,l] \subseteq (\overline{S\cup\{i\}})^{\sigma_S}(m,l] = (\overline{S\cup\{i\}})^{\tau_m}(m,l], \tag{6.47}$$

with an equality in case $c(q,\sigma_S) = c(m,\sigma_S)$. Since all players in $S^{\sigma_S}(m,q)$ have been moved by the algorithm, we have

$$(S\cup\{i\})^{\tau_m}(m,l] = (S\cup\{i\})^{\tau_q}(q,l], \tag{6.48}$$

if player q has also been moved by the algorithm, and

$$(S\cup\{i\})^{\tau_m}(m,l] = (S\cup\{i\})^{\tau_q}(q,l] \cup \{q\}, \tag{6.49}$$

if player q has not been moved by the algorithm. As a consequence,

$$\begin{aligned} & \alpha_{(S\cup\{i\})^{\tau_m}(m,l]} p_m - \alpha_m p_{N^{\tau_m}(m,l]} \\ = \quad & \alpha_{(S\cup\{i\})^{\tau_m}(m,l]} p_m - \alpha_m \left(p_{(S\cup\{i\})^{\tau_m}(m,l]} + p_{(\overline{S\cup\{i\}})^{\tau_m}(m,l]} \right) \end{aligned}$$

$$
\begin{aligned}
&\overset{(6.47),(6.48)}{=} \alpha_{(S\cup\{i\})^{\tau_q}(q,l]}p_m \\
&\quad - \alpha_m \left(p_{(S\cup\{i\})^{\tau_q}(q,l]} + p_{(\overline{S\cup\{i\}})^{\tau_q}(q,l]} + p_{(\overline{S\cup\{i\}})^{\tau_m}(m,l]\setminus(\overline{S\cup\{i\}})^{\tau_q}(q,l]}\right) \\
&\leq \alpha_{(S\cup\{i\})^{\tau_q}(q,l]}p_m - \alpha_m \left(p_{(S\cup\{i\})^{\tau_q}(q,l]} + p_{(\overline{S\cup\{i\}})^{\tau_q}(q,l]}\right) \\
&= \alpha_{(S\cup\{i\})^{\tau_q}(q,l]}p_m - \alpha_m p_{N^{\tau_q}(q,l]} \overset{(6.46)}{<} 0,
\end{aligned}
$$

if player q has been moved by the algorithm, and

$$
\begin{aligned}
& \alpha_{(S\cup\{i\})^{\tau_m}(m,l]}p_m - \alpha_m p_{N^{\tau_m}(m,l]} \\
&= \alpha_{(S\cup\{i\})^{\tau_m}(m,l]}p_m - \alpha_m \left(p_{(S\cup\{i\})^{\tau_m}(m,l]} + p_{(\overline{S\cup\{i\}})^{\tau_m}(m,l]}\right) \\
&\overset{(6.47),(6.49)}{=} \alpha_{(S\cup\{i\})^{\tau_q}(q,l]}p_m \\
&\quad - \alpha_m \left(p_{(S\cup\{i\})^{\tau_q}(q,l]} + p_{(\overline{S\cup\{i\}})^{\tau_q}(q,l]} + p_{(\overline{S\cup\{i\}})^{\tau_m}(m,l]\setminus(\overline{S\cup\{i\}})^{\tau_q}(q,l]}\right) \\
&\quad + \alpha_q p_m - \alpha_m p_q \\
&\overset{u_q<u_m}{<} \alpha_{(S\cup\{i\})^{\tau_q}(q,l]}p_m - \alpha_m \left(p_{(S\cup\{i\})^{\tau_q}(q,l]} + p_{(\overline{S\cup\{i\}})^{\tau_q}(q,l]}\right) \\
&= \alpha_{(S\cup\{i\})^{\tau_q}(q,l]}p_m - \alpha_m p_{N^{\tau_q}(q,l]} \overset{(6.46)}{<} 0,
\end{aligned}
$$

if player q has not been moved by the algorithm, which proves (6.10).

Case 2(ii): $[S^{\sigma_S}(m,q) \cap S^{\tau_m}(m,l] \neq \emptyset]$

We move player m behind player l in τ_m in two stages. In the first stage, player m will be moved behind a specific player t. Then, in the second stage, player m will be moved behind player l. Using similar arguments as in case 1 we can show that the move in the first stage is not beneficial and using similar arguments as in case 2(i) we can show that the move in the second stage is not beneficial. As a consequence, since cost differences have an additive structure and because the moves in both stages are not beneficial, moving player m behind player l in τ_m is not beneficial.

Let player $t \in S^{\sigma_S}(m,q)$ be the closest predecessor of player q in σ_S who is also a member of $S^{\tau_m}(m,l]$. Note that due to the assumption $S^{\sigma_S}(m,q) \cap S^{\tau_m}(m,l] \neq \emptyset$, player t exists. Because player t is a predecessor of player q in σ_S, we have $u_m \leq u_k$ for all $k \in S^{\sigma_S}(m,t]$. Therefore, using the same arguments as in case 1, we have

$$
\alpha_{(S\cup\{i\})^{\tau_m}(m,t]}p_m - \alpha_m p_{N^{\tau_m}(m,t]} \leq 0. \tag{6.50}
$$

Since player t is a predecessor of player q in σ_S and because $t \in S^{\tau_m}(m,l]$ (which means that player t has not been moved by the algorithm), we have

$$\begin{aligned}(\overline{S \cup \{i\}})^{\tau_q}(q,l] &= (\overline{S \cup \{i\}})^{\sigma_S}(q,l] \subseteq (\overline{S \cup \{i\}})^{\sigma_S}(t,l] \\ &= (\overline{S \cup \{i\}})^{\tau_t}(t,l] = (\overline{S \cup \{i\}})^{\tau_m}(t,l], \end{aligned} \tag{6.51}$$

with an equality in case $c(q,\sigma_S) = c(t,\sigma_S)$. Because of the definition of player t we have

$$(S \cup \{i\})^{\tau_m}(t,l] = (S \cup \{i\})^{\tau_q}(q,l], \tag{6.52}$$

if player q has been moved by the algorithm, and

$$(S \cup \{i\})^{\tau_m}(t,l] = (S \cup \{i\})^{\tau_q}(q,l] \cup \{q\}, \tag{6.53}$$

if player q has not been moved by the algorithm. As a consequence, by using the additive structure of cost differences, we have

$$\begin{aligned}
& \alpha_{(S\cup\{i\})^{\tau_m}(m,l]}p_m - \alpha_m p_{N^{\tau_m}(m,l]} \\
= \; & \left(\alpha_{(S\cup\{i\})^{\tau_m}(m,t]} + \alpha_{(S\cup\{i\})^{\tau_m}(t,l]}\right) p_m - \alpha_m \left(p_{N^{\tau_m}(m,t]} + p_{N^{\tau_m}(t,l]}\right) \\
\stackrel{(6.50)}{\leq} \; & \alpha_{(S\cup\{i\})^{\tau_m}(t,l]}p_m - \alpha_m \left(p_{(S\cup\{i\})^{\tau_m}(t,l]} + p_{(\overline{S\cup\{i\}})^{\tau_m}(t,l]}\right) \\
\stackrel{(6.51),(6.52)}{=} \; & \alpha_{(S\cup\{i\})^{\tau_q}(q,l]}p_m \\
& - \alpha_m \left(p_{(S\cup\{i\})^{\tau_q}(q,l]} + p_{(\overline{S\cup\{i\}})^{\tau_q}(q,l]} + p_{(\overline{S\cup\{i\}})^{\tau_m}(t,l]\setminus(\overline{S\cup\{i\}})^{\tau_q}(q,l]}\right) \\
\leq \; & \alpha_{(S\cup\{i\})^{\tau_q}(q,l]}p_m - \alpha_m \left(p_{(S\cup\{i\})^{\tau_q}(q,l]} + p_{(\overline{S\cup\{i\}})^{\tau_q}(q,l]}\right) \\
= \; & \alpha_{(S\cup\{i\})^{\tau_q}(q,l]}p_m - \alpha_m p_{N^{\tau_q}(q,l]} \stackrel{(6.46)}{<} 0,
\end{aligned}$$

if player q has been moved by the algorithm, and

$$\begin{aligned}
& \alpha_{(S\cup\{i\})^{\tau_m}(m,l]}p_m - \alpha_m p_{N^{\tau_m}(m,l]} \\
= \; & \left(\alpha_{(S\cup\{i\})^{\tau_m}(m,t]} + \alpha_{(S\cup\{i\})^{\tau_m}(t,l]}\right) p_m - \alpha_m \left(p_{N^{\tau_m}(m,t]} + p_{N^{\tau_m}(t,l]}\right) \\
\stackrel{(6.50)}{\leq} \; & \alpha_{(S\cup\{i\})^{\tau_m}(t,l]}p_m - \alpha_m \left(p_{(S\cup\{i\})^{\tau_m}(t,l]} + p_{(\overline{S\cup\{i\}})^{\tau_m}(t,l]}\right) \\
\stackrel{(6.51),(6.53)}{=} \; & \alpha_{(S\cup\{i\})^{\tau_q}(q,l]}p_m \\
& - \alpha_m \left(p_{(S\cup\{i\})^{\tau_q}(q,l]} + p_{(\overline{S\cup\{i\}})^{\tau_q}(q,l]} + p_{(\overline{S\cup\{i\}})^{\tau_m}(t,l]\setminus(\overline{S\cup\{i\}})^{\tau_q}(q,l]}\right) \\
& + \alpha_q p_m - \alpha_m p_q
\end{aligned}$$

$$
\begin{aligned}
&\overset{u_q<u_m}{<} \quad \alpha_{(S\cup\{i\})^{\tau_q}(q,l]}p_m - \alpha_m\left(p_{(S\cup\{i\})^{\tau_q}(q,l]} + p_{\overline{(S\cup\{i\})}^{\tau_q}(q,l]}\right)\\
&= \quad \alpha_{(S\cup\{i\})^{\tau_q}(q,l]}p_m - \alpha_m p_{N^{\tau_q}(q,l]} \overset{(6.46)}{<} 0,
\end{aligned}
$$

if player q has not been moved by the algorithm. Hence, by using the additive structure of cost differences we have shown that moving player m behind player l in τ_m is not beneficial.

On assumption 2 for Theorem 6.6.1 in Section 6.6

We will prove that without loss of generality we can assume

$$(S\cup\{j\}\cup\{i\})^{\sigma_0}_{c(j,S\cup\{j\}\cup\{i\},\sigma_0)} = \{j\},$$

and

$$(S\cup\{j\}\cup\{i\})^{\sigma_0}_{c(i,S\cup\{j\}\cup\{i\},\sigma_0)} = \{i\},$$

in order to prove the convexity of SoSi sequencing games. Suppose that player i is not the only player in his component in σ_0, i.e.,

$$\{i\} \subset (S\cup\{j\}\cup\{i\})^{\sigma_0}_{c(i,\sigma_0)}.$$

Then, for example, the direct predecessor of player i in σ_0 is a member of $S\cup\{j\}\cup\{i\}$. We can define a different one-machine sequencing situation $(\overline{N}, \overline{\sigma_0}, \overline{p}, \overline{\alpha})$ where $\overline{N} = N\cup\{d\}$ with $d \notin N$,

$$\overline{\sigma_0}(k) = \begin{cases} \sigma_0(k) & \text{if } k\in P(\sigma_0,i),\\ \sigma_0(i) & \text{if } k=d,\\ \sigma_0(k)+1 & \text{if } k\in\{i\}\cup F(\sigma_0,i),\end{cases}$$

$$\overline{p}_k = \begin{cases} p_k & \text{if } k\in N,\\ 0 & \text{if } k=d,\end{cases}$$

and

$$\overline{\alpha}_k = \begin{cases} \alpha_k & \text{if } k\in N,\\ 0 & \text{if } k=d.\end{cases}$$

Hence, this new sequencing situation $(\overline{N}, \overline{\sigma_0}, \overline{p}, \overline{\alpha})$ is obtained from the original sequencing situation (N, σ_0, p, α) by adding a dummy player, with processing time and costs per time unit both equal to zero[7], directly in front of player i such that the predecessor of player i does not belong to $S \cup \{j\} \cup \{i\}$ anymore.

Let $(\overline{N}, \overline{v})$ be the SoSi sequencing game corresponding to one-machine sequencing situation $(\overline{N}, \overline{\sigma_0}, \overline{p}, \overline{\alpha})$. Although $\alpha_d = 0$ and $p_d = 0$, we can still apply the algorithm with respect to coalition $S \cup \{j\} \cup \{i\}$ and initial processing order $\overline{\sigma_0}$, because player d is not a member of $S \cup \{j\} \cup \{i\}$. The mutual order of the players in N will be the same in $\text{Alg}((N, \sigma_0, p, \alpha), S \cup \{j\} \cup \{i\})$ and $\text{Alg}((\overline{N}, \overline{\sigma_0}, \overline{p}, \overline{\alpha}), S \cup \{j\} \cup \{i\})$. Therefore, $v(S \cup \{j\} \cup \{i\}) = \overline{v}(S \cup \{j\} \cup \{i\})$. Similarly, we have $v(S) = \overline{v}(S)$, $v(S \cup \{i\}) = \overline{v}(S \cup \{i\})$ and $v(S \cup \{j\}) = \overline{v}(S \cup \{j\})$.

Note that if also the direct follower of player i in σ_0 is a member of $S \cup \{j\} \cup \{i\}$, then we also add a dummy player directly behind player i. By adding a dummy player directly in front and behind player i, player i will be the only player in his component. Moreover, all arguments can also be applied to player j. Hence, without loss of generality we can assume that player i and j are both the only player in their component in σ_0.

Proof of claims 1 – 4 in Theorem 6.6.1

Claim 1: $[c(k, \sigma_{S \cup \{j\} \cup \{i\}}) = c(k, \sigma_{S \cup \{i\}}) \geq c(i, \sigma_S)$ for all $k \in M^i(S)]$
From (6.13) it follows that $c(k, \sigma_{S \cup \{i\}}) \geq c(i, \sigma_S)$ for all $k \in M^i(S)$. Moreover, from Proposition 6.5.4 it follows that if we go from $\sigma_{S \cup \{i\}}$ to $\sigma_{S \cup \{j\} \cup \{i\}}$, then the players to the right of player j in $\sigma_{S \cup \{i\}}$ will stay in the same component. Since all players in $M^i(S)$ are to the right of player j in $\sigma_{S \cup \{i\}}$, we have

$$c(k, \sigma_{S \cup \{j\} \cup \{i\}}) = c(k, \sigma_{S \cup \{i\}}),$$

for all $k \in M^i(S)$.

Claim 2: $[c(k, \sigma_{S \cup \{j\} \cup \{i\}}) = c(k, \sigma_{S \cup \{j\}})$ for all $k \in M^i_{1c}(S)]$
From Proposition 6.5.4 it follows that if we go from $\sigma_{S \cup \{j\}}$ to $\sigma_{S \cup \{j\} \cup \{i\}}$, then the players to the right of player i in $\sigma_{S \cup \{j\}}$ will stay in the same component. Since all players in $M^i_{1c}(S)$ are to the right of player i in $\sigma_{S \cup \{j\}}$, we have

$$c(k, \sigma_{S \cup \{j\} \cup \{i\}}) = c(k, \sigma_{S \cup \{j\}}),$$

[7] Note that in Section 6.2 we assumed that every player has strictly positive processing time and costs per time unit. However, since this dummy player never belongs to a coalition that will be considered by the algorithm, adding the dummy player is harmless.

for all $k \in M^i_{1c}(S)$.

Claim 3: $[c(k,\sigma_S) = c(k,\sigma_{S\cup\{j\}})$ for all $k \in M^i_2(S)]$
From Proposition 6.5.4 it follows that if we go from σ_S to $\sigma_{S\cup\{j\}}$, then the players to the right of player j in σ_S will stay in the same component. Since all players in $M^i_2(S)$ are to the right of player j in σ_S, we have

$$c(k,\sigma_S) = c(k,\sigma_{S\cup\{j\}}),$$

for all $k \in M^i_2(S)$.

Claim 4: $[c(k,\sigma_S) = c(k,\sigma_{S\cup\{j\}})$ for all $k \in M^i_{1a}(S)]$
From Proposition 6.5.5 it follows that the players who are in $\sigma_{S\cup\{j\}}$ to the left of the original component of player j have not been moved when going from σ_S to $\sigma_{S\cup\{j\}}$. Since all players in $M^i_{1a}(S)$ are to the left of the original component of player j in $\sigma_{S\cup\{j\}}$, we have

$$c(k,\sigma_S) = c(k,\sigma_{S\cup\{j\}}),$$

for all $k \in M^i_{1a}(S)$.

Proof of claim 5 in Theorem 6.6.1

Let $k \in M^i_{1a}(S) \cup M^i_{1b}(S) \cup M^i_2(S)$, we will prove

$$\overline{\delta}_k \leq \delta_k.$$

Note that we consider the players in $M^i_{1a}(S) \cup M^i_{1b}(S) \cup M^i_2(S)$ from the right to the left with respect to $\overline{\sigma}$ and $\sigma_{S\cup\{j\}}$. So, if $i \in M^i_{1a}(S) \cup M^i_{1b}(S) \cup M^i_2(S)$, then player i is the first player who is moved. From now on we distinguish between two cases: $k = i$ and $k \neq i$.

<u>Case 1:</u> $[k = i]$
As player i is the first player who is moved, we have $\overline{\tau}_i = \overline{\sigma}$ and $\tau_i = \sigma_{S\cup\{j\}}$. Note that we have

$$\overline{\delta}_i = \alpha_{(S\cup\{i\})^{\overline{\tau}_i}(i,\overline{r}_i]}p_i - \alpha_i p_{N^{\overline{\tau}_i}(i,\overline{r}_i]},$$

and

$$\delta_i = \alpha_{(S\cup\{j\}\cup\{i\})^{\tau_i}(i,r_i]}p_i - \alpha_i p_{N^{\tau_i}(i,r_i]}.$$

In order to prove $\overline{\delta}_i \leq \delta_i$, we compare the two sets of players that player i jumps over in $\overline{\tau}_i$ and τ_i. We will show that all players that player i jumps over in $\overline{\tau}_i$, player i also jumps over in τ_i. However, there might be some players that player i jumps over in τ_i but not in $\overline{\tau}_i$. It can be shown that these extra players that player i jumps over in τ_i all have a higher urgency than player i and thus this results in extra cost savings. Formally, we show the following three statements:

- Statement 1(a): $(\overline{S\cup\{i\}})^{\overline{\tau}_i}(i,\overline{r}_i] = (\overline{S\cup\{j\}\cup\{i\}})^{\tau_i}(i,r_i]$,
- Statement 1(b): $(S\cup\{i\})^{\overline{\tau}_i}(i,\overline{r}_i] \subseteq (S\cup\{j\}\cup\{i\})^{\tau_i}(i,r_i]$,
- Statement 1(c): if $(S\cup\{i\})^{\overline{\tau}_i}(i,\overline{r}_i] \subset (S\cup\{j\}\cup\{i\})^{\tau_i}(i,r_i]$, then

$$\alpha_{(S\cup\{j\}\cup\{i\})^{\tau_i}(i,r_i]\backslash(S\cup\{i\})^{\overline{\tau}_i}(i,\overline{r}_i]}p_i - \alpha_i p_{(S\cup\{j\}\cup\{i\})^{\tau_i}(i,r_i]\backslash(S\cup\{i\})^{\overline{\tau}_i}(i,\overline{r}_i]} > 0.$$

Proof of statement 1(a). From claim 1 it follows that $c(i,\sigma_{S\cup\{i\}}) = c(i,\sigma_{S\cup\{j\}\cup\{i\}})$ and thus player i will be moved in both processing orders to the same component, i.e.,

$$c(\overline{r}_i,\overline{\tau}_i) = c(r_i,\tau_i).$$

Moreover, by the definition of $\overline{\sigma}$ and since $i \notin S\cup\{j\}$, we have $c(i,\overline{\sigma}) = c(i,\sigma_{S\cup\{j\}})$ and thus

$$c(i,\overline{\tau}_i) = c(i,\tau_i). \tag{6.54}$$

Hence, player i is moved in $\overline{\tau}_i$ and τ_i from the same component and to the same component, so player i jumps in $\overline{\tau}_i$ and τ_i over the same players outside $S\cup\{j\}\cup\{i\}$. Moreover, because $\overline{\sigma}(j) < \overline{\sigma}(i)$ and thus $\overline{\tau}_i(j) < \overline{\tau}_i(i)$, we have $j \notin N^{\overline{\tau}_i}(i,\overline{r}_i]$. To summarize,

$$(\overline{S\cup\{i\}})^{\overline{\tau}_i}(i,\overline{r}_i] = (\overline{S\cup\{j\}\cup\{i\}})^{\overline{\tau}_i}(i,\overline{r}_i] = (\overline{S\cup\{j\}\cup\{i\}})^{\tau_i}(i,r_i].$$

Proof of statement 1(b). Let $l \in (S \cup \{i\})^{\overline{\tau}_i}(i, \overline{r}_i]$. From (6.16) we know that player l is in $\overline{\tau}_i$ in a component at most as far to the right as in τ_i, i.e.,

$$c(l, \overline{\tau}_i) \leq c(l, \tau_i).$$

Moreover, from $l \in (S \cup \{i\})^{\overline{\tau}_i}(i, \overline{r}_i]$ and (6.54) it follows that

$$c(i, \tau_i) = c(i, \overline{\tau}_i) < c(l, \overline{\tau}_i).$$

Note that there is a strict inequality because player i is the only player in his component. Combining the previous two equations we get

$$c(i, \tau_i) < c(l, \tau_i), \tag{6.55}$$

i.e., player l is also to the right of player i in τ_i. Now suppose player l is to the right of player r_i in τ_i, then player l is to the right of player i in $\sigma_{S \cup \{j\} \cup \{i\}}$ and thus

$$c(l, \sigma_{S \cup \{i\}}) \leq c(i, \sigma_{S \cup \{i\}}) = c(i, \sigma_{S \cup \{j\} \cup \{i\}}) \leq c(l, \sigma_{S \cup \{j\} \cup \{i\}}),$$

where the first inequality follows from $l \in (S \cup \{i\})^{\overline{\tau}_i}(i, \overline{r}_i]$ and the first equality follows from claim 1. Therefore, it follows that the swap of player i and player l with respect to $\sigma_{S \cup \{j\} \cup \{i\}}$ is admissible. From Lemma 6.4.1 it then follows that $u_i \geq u_l$. However, since player i jumps over player l when going from $\overline{\sigma}$ to $\sigma_{S \cup \{i\}}$, it follows from Lemma 6.4.1 that $u_l > u_i$, which contradicts $u_i \geq u_l$. Therefore l cannot be to the right of player r_i in τ_i, so player l is to the left of player r_i in τ_i. Combining this result with (6.55) we have

$$l \in (S \cup \{j\} \cup \{i\})^{\tau_i}(i, r_i].$$

Proof of statement 1(c). Let $(S \cup \{i\})^{\overline{\tau}_i}(i, \overline{r}_i] \subset (S \cup \{j\} \cup \{i\})^{\tau_i}(i, r_i]$. For every player $l \in (S \cup \{j\} \cup \{i\})^{\tau_i}(i, r_i] \backslash (S \cup \{i\})^{\overline{\tau}_i}(i, \overline{r}_i]$ we have $u_l > u_i$ since player i jumps over player l when going from $\sigma_{S \cup \{j\}}$ to $\sigma_{S \cup \{j\} \cup \{i\}}$ (cf. Lemma 6.4.1). Combining this with Proposition 6.5.1 applied on the set $U = (S \cup \{j\} \cup \{i\})^{\tau_i}(i, r_i] \backslash (S \cup \{i\})^{\overline{\tau}_i}(i, \overline{r}_i]$ and player i, we have that

$$\frac{\alpha_i}{p_i} < \frac{\alpha_{(S \cup \{j\} \cup \{i\})^{\tau_i}(i, r_i] \backslash (S \cup \{i\})^{\overline{\tau}_i}(i, \overline{r}_i]}}{p_{(S \cup \{j\} \cup \{i\})^{\tau_i}(i, r_i] \backslash (S \cup \{i\})^{\overline{\tau}_i}(i, \overline{r}_i]}},$$

i.e., $\alpha_{(S\cup\{j\}\cup\{i\})^{\tau_i}(i,r_i]\setminus(S\cup\{i\})^{\overline{\tau}_i}(i,\overline{r}_i]}p_i - \alpha_i p_{(S\cup\{j\}\cup\{i\})^{\tau_i}(i,r_i]\setminus(S\cup\{i\})^{\overline{\tau}_i}(i,\overline{r}_i]} > 0$.

Note that if in statement 1(b) we have equality, then the inequality $\overline{\delta}_i \leq \delta_i$ follows immediately from statement 1(a). Next, if in statement 1(b) we have a strict subset, then

$$\begin{aligned}
\overline{\delta}_i &= \alpha_{(S\cup\{i\})^{\overline{\tau}_i}(i,\overline{r}_i]}p_i - \alpha_i p_{N^{\overline{\tau}_i}(i,\overline{r}_i]} \\
&= \alpha_{(S\cup\{i\})^{\overline{\tau}_i}(i,\overline{r}_i]}p_i - \alpha_i \left(p_{(S\cup\{i\})^{\overline{\tau}_i}(i,\overline{r}_i]} + p_{(\overline{S\cup\{i\}})^{\overline{\tau}_i}(i,\overline{r}_i]}\right) \\
&\overset{1(c)}{<} \alpha_{(S\cup\{i\})^{\overline{\tau}_i}(i,\overline{r}_i]}p_i - \alpha_i \left(p_{(S\cup\{i\})^{\overline{\tau}_i}(i,\overline{r}_i]} + p_{(\overline{S\cup\{i\}})^{\overline{\tau}_i}(i,\overline{r}_i]}\right) \\
&\quad + \alpha_{(S\cup\{j\}\cup\{i\})^{\tau_i}(i,r_i]\setminus(S\cup\{i\})^{\overline{\tau}_i}(i,\overline{r}_i]}p_i - \alpha_i p_{(S\cup\{j\}\cup\{i\})^{\tau_i}(i,r_i]\setminus(S\cup\{i\})^{\overline{\tau}_i}(i,\overline{r}_i]} \\
&\overset{1(a)}{=} \left(\alpha_{(S\cup\{i\})^{\overline{\tau}_i}(i,\overline{r}_i]} + \alpha_{(S\cup\{j\}\cup\{i\})^{\tau_i}(i,r_i]\setminus(S\cup\{i\})^{\overline{\tau}_i}(i,\overline{r}_i]}\right)p_i \\
&\quad - \alpha_i \left(p_{(S\cup\{i\})^{\overline{\tau}_i}(i,\overline{r}_i]} + p_{(S\cup\{j\}\cup\{i\})^{\tau_i}(i,r_i]\setminus(S\cup\{i\})^{\overline{\tau}_i}(i,\overline{r}_i]} + p_{(\overline{S\cup\{j\}\cup\{i\}})^{\tau_i}(i,r_i]}\right) \\
&= \alpha_{(S\cup\{j\}\cup\{i\})^{\tau_i}(i,r_i]}p_i - \alpha_i \left(p_{(S\cup\{j\}\cup\{i\})^{\tau_i}(i,r_i]} + p_{(\overline{S\cup\{j\}\cup\{i\}})^{\tau_i}(i,r_i]}\right) \\
&= \alpha_{(S\cup\{j\}\cup\{i\})^{\tau_i}(i,r_i]}p_i - \alpha_i p_{N^{\tau_i}(i,r_i]} \\
&= \delta_i.
\end{aligned}$$

The idea behind the previous strict inequality is as follows. Player i jumps in $\overline{\tau}_i$ over the same players as in τ_i, but additionally player i jumps in τ_i also over some extra players. It follows from statement 1(c) that these extra players all have a higher urgency and thus the jump of player i over those extra players results in cost savings.

<u>Case 2:</u> $[k \neq i]$

The difference with respect to case 1 is that in this case player k is not necessarily the only player in its component, while in case 1 player i was the only player in its component of $S \cup \{j\} \cup \{i\}$ with respect to σ_0. Another difference is that now player k might not be the first player who is moved, and thus $\overline{\tau}_k$ and $\overline{\sigma}$, and τ_i and $\sigma_{S\cup\{j\}}$ might differ.

Note that

$$\overline{\delta}_k = \alpha_{(S\cup\{i\})^{\overline{\tau}_k}(k,\overline{r}_k]}p_k - \alpha_k p_{N^{\overline{\tau}_k}(k,\overline{r}_k]},$$

and

$$\delta_k = \alpha_{(S\cup\{j\}\cup\{i\})^{\tau_k}(k,r_k]}p_k - \alpha_k p_{N^{\tau_k}(k,m_k]}.$$

In order to prove $\overline{\delta}_k \leq \delta_k$, we compare the two sets of players that player k jumps over in $\overline{\tau}_k$ and τ_k. We will show that all players, excluding player j, that player k jumps over in $\overline{\tau}_k$, player k also jumps over in τ_k. Formally, we show the following three statements (which are similar to statements 1(a)-(c)):

– Statement 2(a): $\left(\overline{(S \cup \{i\})}^{\overline{\tau}_k}(k, \overline{r}_k]\right) \backslash \{j\} = \overline{(S \cup \{j\} \cup \{i\})}^{\tau_k}(k, r_k]$,

– Statement 2(b): $(S \cup \{i\})^{\overline{\tau}_k}(k, \overline{r}_k] \subseteq ((S \cup \{j\} \cup \{i\})^{\tau_k}(k, r_k]) \backslash \{j\}$,

– Statement 2(c): if $(S \cup \{i\})^{\overline{\tau}_k}(k, \overline{r}_k] \subset ((S \cup \{j\} \cup \{i\})^{\tau_k}(k, r_k]) \backslash \{j\}$, then

$$\alpha_{(((S\cup\{j\}\cup\{i\})^{\tau_k}(k,r_k])\backslash\{j\})\backslash((S\cup\{i\})^{\overline{\tau}_k}(k,\overline{r}_k])} p_k - \alpha_k p_{(((S\cup\{j\}\cup\{i\})^{\tau_k}(k,r_k])\backslash\{j\})\backslash((S\cup\{i\})^{\overline{\tau}_k}(k,\overline{r}_k])} > 0.$$

Note that in the proof of statement 1(a) and 1(c) the fact that player i is the only player in its component is not used and therefore the proofs for statements 2(a) and 2(b) are similar.

Proof of statement 2(b). Let $l \in (S \cup \{i\})^{\overline{\tau}_k}(k, \overline{r}_k]$. From (6.17) we know that player l is in $\overline{\tau}_k$ in a component at most as far to the right as in τ_k, i.e.,

$$c(l, \overline{\tau}_k) \leq c(l, \tau_k).$$

Moreover, from (6.15) we have $c(k, \overline{\sigma}) = c(k, \sigma_{S\cup\{j\}})$ and thus $c(k, \overline{\tau}_k) = c(k, \tau_k)$. Together with $l \in (S \cup \{i\})^{\overline{\tau}_k}(k, \overline{r}_k]$ it follows that

$$c(k, \tau_k) = c(k, \overline{\tau}_k) \leq c(l, \overline{\tau}_k).$$

Combining the previous two equations we get

$$c(k, \tau_k) \leq c(l, \tau_k). \tag{6.56}$$

This implies

$$\tau_k(k) < \tau_k(l).$$

For this, note that if we have an equality in (6.56), then $c(k, \overline{\tau}_k) = c(l, \overline{\tau}_k)$. Moreover, since $l \in (S \cup \{i\})^{\overline{\tau}_k}(k, \overline{r}_k]$, we know $\overline{\tau}_k(k) < \overline{\tau}_k(l)$ and thus also $\tau_k(k) < \tau_k(l)$ (because both τ_k and $\overline{\tau}_k$ are urgency respecting processing orders and moreover because the tiebreaking rule, in case of equal urgencies, mentioned in condition (iii) of $\overline{\sigma}$ is the same tiebreaking rule as in property (iv) of the algorithm). On the other hand, if there is a strict inequality in (6.56), then automatically $\tau_k(k) < \tau_k(l)$. Using similar arguments as in the proof of statement 1(b) we have

$$l \in N^{\tau_k}(k, r_k],$$

and since $l \in S \cup \{i\}$, we have

$$l \in ((S \cup \{j\} \cup \{i\})^{\tau_k}(k, r_k]) \setminus \{j\}.$$

Now we continue with the main line of the proof. Using similar arguments as in case 1, it follows from statement 2(a)-(c) that

$$\begin{aligned} & \alpha_{(S\cup\{i\})^{\overline{\tau}_k}(k,\overline{r}_k]} p_k - \alpha_k \left(p_{(S\cup\{i\})^{\overline{\tau}_k}(k,\overline{r}_k]} + p_{\left((\overline{S\cup\{i\}})^{\overline{\tau}_k}(k,\overline{r}_k]\right)\setminus\{j\}} \right) \\ \leq \; & \alpha_{((S\cup\{j\}\cup\{i\})^{\tau_k}(k,r_k])\setminus\{j\}} p_k \\ & - \alpha_k \left(p_{((S\cup\{j\}\cup\{i\})^{\tau_k}(k,r_k])\setminus\{j\}} + p_{(\overline{S\cup\{j\}\cup\{i\}})^{\tau_k}(k,r_k]} \right). \end{aligned} \tag{6.57}$$

We distinguish from now on between the following four cases: case 2(i) where player k does not jump over player j in both $\overline{\tau}_k$ and τ_k, case 2(ii) where player k jumps over player j in both $\overline{\tau}_k$ and τ_k, case 2(iii) where player k jumps over player j in $\overline{\tau}_k$ but not in τ_k and case 2(iv) where player k jumps over player j in τ_k but not in $\overline{\tau}_k$.

Case 2(i): $[j \notin N^{\overline{\tau}_k}(k, \overline{r}_k]$ and $j \notin N^{\tau_k}(k, r_k]]$
Note that if in statement 2(b) we have equality, then the inequality $\overline{\delta}_k \leq \delta_k$ follows immediately from statement 2(a). Next, if in statement 2(b) we have a strict subset, then

$$\begin{aligned} \overline{\delta}_k & = \alpha_{(S\cup\{i\})^{\overline{\tau}_k}(k,\overline{r}_k]} p_k - \alpha_k p_{N^{\overline{\tau}_k}(k,\overline{r}_k]} \\ & \overset{j\notin N^{\overline{\tau}_k}(k,\overline{r}_k]}{=} \alpha_{(S\cup\{i\})^{\overline{\tau}_k}(k,\overline{r}_k]} p_k - \alpha_k \left(p_{(S\cup\{i\})^{\overline{\tau}_k}(k,\overline{r}_k]} + p_{\left((\overline{S\cup\{i\}})^{\overline{\tau}_k}(k,\overline{r}_k]\right)\setminus\{j\}} \right) \\ & \overset{(6.57)}{<} \alpha_{((S\cup\{j\}\cup\{i\})^{\tau_k}(k,r_k])\setminus\{j\}} p_k - \alpha_k \left(p_{((S\cup\{j\}\cup\{i\})^{\tau_k}(k,r_k])\setminus\{j\}} + p_{(\overline{S\cup\{j\}\cup\{i\}})^{\tau_k}(k,r_k]} \right) \\ & \overset{j\notin N^{\tau_k}(k,r_k]}{=} \alpha_{(S\cup\{j\}\cup\{i\})^{\tau_k}(k,r_k]} p_k - \alpha_k \left(p_{(S\cup\{j\}\cup\{i\})^{\tau_k}(k,r_k]} + p_{(\overline{S\cup\{j\}\cup\{i\}})^{\tau_k}(k,r_k]} \right) \\ & = \alpha_{(S\cup\{j\}\cup\{i\})^{\tau_k}(k,r_k]} p_k - \alpha_k p_{N^{\tau_k}(k,r_k]} \\ & = \delta_k. \end{aligned}$$

The idea behind the strict inequality is as follows. Player k jumps in $\overline{\tau}_k$ over the same players as in τ_k, but additionally player k jumps in τ_k also over some extra players. It follows from statement 2(c) that these extra players all have a higher urgency and thus the jump of player k over those extra players results in cost savings.

Case 2(ii): $[j \in N^{\overline{\tau}_k}(k, \overline{r}_k]$ and $j \in N^{\tau_k}(k, r_k]]$
It follows that

$$\begin{aligned}
\overline{\delta}_k &= \alpha_{(S\cup\{i\})^{\overline{\tau}_k}(k,\overline{r}_k]}p_k - \alpha_k p_{N^{\overline{\tau}_k}(k,\overline{r}_k]} \\
&\overset{j\in N^{\overline{\tau}_k}(k,\overline{r}_k]}{=} \alpha_{(S\cup\{i\})^{\overline{\tau}_k}(k,\overline{r}_k]}p_k - \alpha_k \left(p_{(S\cup\{i\})^{\overline{\tau}_k}(k,\overline{r}_k]} + p_{\left((\overline{S\cup\{i\}})^{\overline{\tau}_k}(k,\overline{r}_k]\right)\setminus\{j\}} + p_j\right) \\
&\overset{(6.57)}{\leq} \alpha_{((S\cup\{j\}\cup\{i\})^{\tau_k}(k,r_k])\setminus\{j\}}p_k \\
&\quad - \alpha_k \left(p_{((S\cup\{j\}\cup\{i\})^{\tau_k}(k,r_k])\setminus\{j\}} + p_{(\overline{S\cup\{j\}\cup\{i\}})^{\tau_k}(k,r_k]} + p_j\right) \\
&\overset{j\in N^{\tau_k}(k,r_k]}{=} \alpha_{((S\cup\{j\}\cup\{i\})^{\tau_k}(k,r_k])\setminus\{j\}}p_k - \alpha_k \left(p_{(S\cup\{j\}\cup\{i\})^{\tau_k}(k,r_k]} + p_{(\overline{S\cup\{j\}\cup\{i\}})^{\tau_k}(k,r_k]}\right) \\
&\overset{\alpha_j p_k>0}{<} \left(\alpha_{((S\cup\{j\}\cup\{i\})^{\tau_k}(k,r_k])\setminus\{j\}} + \alpha_j\right) p_k \\
&\quad - \alpha_k \left(p_{(S\cup\{j\}\cup\{i\})^{\tau_k}(k,r_k]} + p_{(\overline{S\cup\{j\}\cup\{i\}})^{\tau_k}(k,r_k]}\right) \\
&\overset{j\in N^{\tau_k}(k,r_k]}{=} \alpha_{(S\cup\{j\}\cup\{i\})^{\tau_k}(k,r_k]}p_k - \alpha_k \left(p_{(S\cup\{j\}\cup\{i\})^{\tau_k}(k,r_k]} + p_{(\overline{S\cup\{j\}\cup\{i\}})^{\tau_k}(k,r_k]}\right) \\
&= \alpha_{(S\cup\{j\}\cup\{i\})^{\tau_k}(k,r_k]}p_k - \alpha_k p_{N^{\tau_k}(k,r_k]} \\
&= \delta_k.
\end{aligned}$$

The idea behind the strict inequality is as follows. All players, including player j, that player k jumps over in $\overline{\tau}_k$, player k also jumps over in τ_k. However, as player j belongs to coalition $S \cup \{j\} \cup \{i\}$ and not to coalition $S \cup \{i\}$, there are some extra cost savings in δ_k. These extra cost savings are due to the reduction of the processing time for player j due to the jump of player k, namely $\alpha_j p_k$.

Case 2(iii): $[j \in N^{\overline{\tau}_k}(k, \overline{r}_k]$ and $j \notin N^{\tau_k}(k, r_k]]$
It follows that

$$\begin{aligned}
\overline{\delta}_k &= \alpha_{(S\cup\{i\})^{\overline{\tau}_k}(k,\overline{r}_k]}p_k - \alpha_k p_{N^{\overline{\tau}_k}(k,\overline{r}_k]} \\
&\overset{j\in N^{\overline{\tau}_k}(k,\overline{r}_k]}{=} \alpha_{(S\cup\{i\})^{\overline{\tau}_k}(k,\overline{r}_k]}p_k - \alpha_k \left(p_{(S\cup\{i\})^{\overline{\tau}_k}(k,\overline{r}_k]} + p_{\left((\overline{S\cup\{i\}})^{\overline{\tau}_k}(k,\overline{r}_k]\right)\setminus\{j\}} + p_j\right) \\
&\overset{(6.57)}{\leq} \alpha_{((S\cup\{j\}\cup\{i\})^{\tau_k}(k,r_k])\setminus\{j\}}p_k \\
&\quad - \alpha_k \left(p_{((S\cup\{j\}\cup\{i\})^{\tau_k}(k,r_k])\setminus\{j\}} + p_{(\overline{S\cup\{j\}\cup\{i\}})^{\tau_k}(k,r_k]} + p_j\right)
\end{aligned}$$

$$
\begin{aligned}
&\stackrel{j\notin N^{\tau_k}(k,r_k]}{=} \quad \alpha_{(S\cup\{j\}\cup\{i\})^{\tau_k}(k,r_k]}p_k - \alpha_k\left(p_{(S\cup\{j\}\cup\{i\})^{\tau_k}(k,r_k]} + p_{\overline{(S\cup\{j\}\cup\{i\})}^{\tau_k}(k,r_k]} + p_j\right)\\
&\stackrel{\alpha_k p_j>0}{<} \quad \alpha_{(S\cup\{j\}\cup\{i\})^{\tau_k}(k,r_k]}p_k - \alpha_k\left(p_{(S\cup\{j\}\cup\{i\})^{\tau_k}(k,r_k]} + p_{\overline{(S\cup\{j\}\cup\{i\})}^{\tau_k}(k,r_k]}\right)\\
&= \quad \alpha_{(S\cup\{j\}\cup\{i\})^{\tau_k}(k,r_k]}p_k - \alpha_k p_{N^{\tau_k}(k,r_k]}\\
&= \quad \delta_k.
\end{aligned}
$$

The idea behind the strict inequality is as follows. All players, excluding player j, that player k jumps over in $\overline{\tau}_k$, player k also jumps over in τ_k. However, as player k jumps over player j in $\overline{\tau}_k$ and not in τ_k, the completion time of player k will increase with at least p_j more in τ_k than in $\overline{\tau}_k$ and thus the cost savings in $\overline{\delta}_k$ are less than in δ_k.

<u>Case 2(iv):</u> $[j \notin N^{\overline{\tau}_k}(k,\overline{r}_k]$ and $j \in N^{\tau_k}(k,r_k]]$
We will show that this case is not possible. As player k jumps over player j when going from $\sigma_{S\cup\{j\}}$ to $\sigma_{S\cup\{j\}\cup\{i\}}$, player j did not jump over player k when going from σ_S to $\sigma_{S\cup\{j\}}$ (otherwise there would be a contradiction with respect to the urgencies, cf. Lemma 6.4.1). Hence, the mutual order of player k and player j is in σ_S the same as in $\sigma_{S\cup\{j\}}$ and thus $\sigma_S(k) < \sigma_S(j)$. Using Figure 6.14 this implies that $k \in M_1^i(S)$. Moreover, as $j \notin N^{\overline{\tau}_k}(k,\overline{r}_k]$ it follows from Figure 6.16 that $k \notin M_{1a}^i(S)$ and thus $k \in M_{1b}^i(S)$. Therefore, it follows from Figure 6.14(c) that $c(k,\sigma_{S\cup\{j\}}) \geq c(j,\sigma_S)$. Hence,

$$
c(j,\sigma_S) \leq c(k,\sigma_{S\cup\{j\}}) \leq c(j,\sigma_{S\cup\{j\}}),
$$

where the last inequality follows from $j \in N^{\tau_k}(k,r_k]$. Therefore, it follows that the swap of player k and player j with respect to $\sigma_{S\cup\{j\}}$ is admissible.

From Lemma 6.4.1 it then follows that $u_k \geq u_j$. However, since player k jumps over player j when going from $\sigma_{S\cup\{j\}}$ to $\sigma_{S\cup\{j\}\cup\{i\}}$, it follows from Lemma 6.4.1 that $u_j > u_k$ too, a contradiction.

References

Abraham, M., R. Kötter, A. Krumnack, and E. Wanke (2006). A connectivity rating for vertices in networks. *Fourth IFIP International Conference on Theoretical Computer Science-TCS*, 283–298.

Alparslan-Gök, S., R. Branzei, V. Fragnelli, and S. Tijs (2013). Sequencing interval situations and related games. *Central European Journal of Operations Research*, **21**, 225–236.

Bietenhader, T. and Y. Okamoto (2006). Core stability of minimum coloring games. *Mathematics of Operations Research*, **31**, 418–431.

Bilbao, J., J. Fernández, N. Jiménez, and J. López (2000). Generating functions for computing power indices efficiently. *Top*, **8**, 191–213.

Bilbao, J., J. Fernández, N. Jiménez, and J. López (2002). Voting power in the European Union enlargement. *European Journal of Operational Research*, **143**, 181–196.

Bird, C. (1976). On cost allocation for a spanning tree: a game theoretic approach. *Networks*, **6**, 335–350.

Borm, P., G. Fiestras-Janeiro, H. Hamers, E. Sánchez, and M. Voorneveld (2002). On the convexity of games corresponding to sequencing situations with due dates. *European Journal of Operational Research*, **136**, 616–634.

Borm, P., H. Hamers, and R. Hendrickx (2001). Operations research games: a survey. *TOP*, **9**, 139–199.

Breton, M. le, G. Owen, and S. Weber (1992). Strongly balanced cooperative games. *International Journal of Game Theory*, **20**, 419–427.

Çiftçi, B., P. Borm, H. Hamers, and M. Slikker (2013). Batch sequening and cooperation. *Journal of Scheduling*, **16**, 405–415.

Chudnovsky, M., N. Robertson, P. Seymour, and R. Thomas (2006). The strong perfect graph theorem. *Annals of Mathematics*, **164**, 51–229.

Curiel, I. (2015). Compensation rules for multi-stage sequencing games. *Annals of Operations Research*, **225**, 65–82.

Curiel, I., G. Pederzoli, and S. Tijs (1989). Sequencing games. *European Journal of Operational Research*, **40**, 344–351.

Curiel, I., J. Potters, R. Prasad, S. Tijs, and B. Veltman (1993). Cooperation in one machine scheduling. *Zeitschrift für Operations Research*, **38**, 113–129.

Deng, X., T. Ibaraki, and H. Nagamochi (1999). Algorithmic aspects of the core of combinatorial optimization games. *Mathematics of Operations Research*, **24**, 751–766.

Deng, X., T. Ibaraki, H. Nagamochi, and W. Zang (2000). Totally balanced combinatorial optimization games. *Mathematical Programming*, **87**, 441–452.

Derks, J. and J. Kuipers (1997). On the core of routing games. *International Journal of Game Theory*, **26**, 193–205.

Dubey, P. (1975). On the uniqueness of the Shapley value. *International Journal of Game Theory*, **4**, 131–139.

Estévez-Fernández, A., M. Mosquera, P. Borm, and H. Hamers (2008). Proportionate flow shop games. *Journal of Scheduling*, **11**, 433–447.

Felleman, D. and D. van Essen (1991). Distributed hierarchical processing in the primate cerebral cortex. *Cerebral Cortex*, **1**, 1–47.

Freeman, L. (1977). A set of measures of centrality based on betweenness. *Sociometry*, **40**, 35–41.

Gerichhausen, M. and H. Hamers (2009). Partitioning sequencing situations and games. *European Journal of Operational Research*, **196**, 207–216.

Gillies, D. (1959). Solutions to general non-zero-sum games. *Contributions to the Theory of Games*, **4**, 47–85.

Gómez, D., E. González-Arangüena, C. Manuel, G. Owen, M. del Pozo, and J. Tejada (2003). Centrality and power in social networks: a game theoretic approach. *Mathematical Social Sciences*, **46**, 27–54.

Grundel, S., B. Çiftçi, P. Borm, and H. Hamers (2013). Family sequencing and cooperation. *European Journal of Operational Research*, **226**, 414–424.

Hamers, H., P. Borm, and S. Tijs (1995). On games corresponding to sequencing situations with ready times. *Mathematical Programming*, **69**, 471–483.

Hamers, H., F. Klijn, and B. van Velzen (2005). On the convexity of precedence sequencing games. *Annals of Operations Research*, **137**, 161–175.

Hamers, H., S. Miquel, and H. Norde (2014). Monotonic stable solutions for minimum coloring games. *Mathematical Programming*, **145**, 509–529.

Harsanyi, J. (1958). *A bargaining model for the cooperative n-person game*. Ph. D. thesis, Stanford university.

Hausken, K. and M. Mohr (2001). The value of a player in n-person games. *Social Choice and Welfare*, **18**, 465–483.

Ichiishi, T. (1981). Super-modularity: applications to convex games and the greedy algorithm for LP. *Journal of Economic Theory*, **25**, 283–286.

Johnston, R. and A. Hunt (1977). Voting power in the E.E.C.'s Council of Ministers: An Eassay on Method in Political Geography. *Geoforum*, **8**, 1–9.

Klijn, F. and E. Sánchez (2006). Sequencing games without initial order. *Mathematical Methods of Operations Research*, **63**, 53–62.

Kötter, R., A. Reid, A. Krumnack, E. Wanke, and O. Sporns (2007). Shapley ratings in brain networks. *Frontiers in Neuroinformatics*, **1**, Article 2.

Kötter, R. and K. Stephan (2003). Network participation indices: characterizing component roles for information processing in neural networks. *Neural Networks*, **16**, 1261–1275.

Kuipers, J., T. Solymosi, and H. Aarts (2000). Computing the nucleolus of some combinatorially-structured games. *Mathematical Programming*, **88**, 541–563.

Lohmann, E., P. Borm, and M. Slikker (2014). Sequencing situations with just-in-time arrival, and related games. *Mathematical Methods of Operations Research*, **80**, 285–305.

McDiarmid, C. and B. Reed (2000). Channel assignment and weighted coloring. *Networks*, **36**, 114–117.

Moretti, S. (2013). On some open problems arising from the application of coalitional games in medicine. *International Game Theory Review*, **15(3)**, 1340020: 1–18.

Musegaas, M., P. Borm, and M. Quant (2015a). Step out - Step in sequencing games. *European Journal of Operational Research*, **246**, 894–906.

Musegaas, M., P. Borm, and M. Quant (2015b). Three-valued simple games. CentER Discussion paper 2015-026, Tilburg University, Tilburg.

Musegaas, M., P. Borm, and M. Quant (2016a). On the convexity of Step out - Step in sequencing games. CentER Discussion paper 2016-043, Tilburg University, Tilburg.

Musegaas, M., P. Borm, and M. Quant (2016b). Simple and three-valued simple minimum coloring games. *Mathematical Methods of Operations Research*, **84**, 239–258.

Musegaas, M., B. Dietzenbacher, and P. Borm (2016). A note on Shapley ratings in brain networks. *Frontiers in Neuroinformatics*, **10**, Article 51.

Neumann, J. von and O. Morgenstern (1944). *Theory of Games and Economic Behavior*. Princeton University Press, Princeton.

Norde, H., S. Moretti, and S. Tijs (2004). Minimum cost spanning tree games and population monotonic allocation schemes. *European Journal of Operational Research*, **154**, 84–97.

Okamoto, Y. (2003). Submodularity of some classes of the combinatorial optimization games. *Mathematical Methods of Operations Research*, **58**, 131–139.

Okamoto, Y. (2008). Fair cost allocations under conflicts –a game-theoretic point of view–. *Discrete Optimization*, **5**, 1–18.

Pardalos, P., T. Mavridou, and J. Xue (1999). The graph coloring problem: a bibliographic survey. In D. Du and P. Pardalos (Eds.), *Handbook of combinatorial optimization*, pp. 1077–1141. Springer US.

Potters, J., I. Curiel, and S. Tijs (1992). Traveling salesman games. *Mathematical Programming*, **53**, 199–211.

Randerath, B. and I. Schiermeyer (2004). Vertex colouring and forbidden subgraphs – a survey. *Graphs and Combinatorics*, **20**, 1–40.

Restrepo, J., E. Ott, and B. Hunt (2006). Characterizing the dynamical importance of network nodes and links. *Physical review letters*, **97**.

Schmeidler, D. (1969). The nucleolus of a characteristic function game. *SIAM Journal of Applied Mathematics*, **17**, 1163–1170.

Shapley, L. (1953). A value for n-person games. *Annals of Mathematics Studies*, **28**, 307–317.

Shapley, L. (1971). Cores of convex games. *International Journal of Game Theory*, **1**, 11–26.

Shapley, L. and M. Shubik (1954). A method for evaluating the distribution of power in a committee system. *American Political Science Review*, **48**, 787–792.

Slikker, M. (2005). Balancedness of sequencing games with multiple parallel machines. *Annals of Operations Research*, **137**, 177–189.

Slikker, M. (2006a). Balancedness of multiple machine sequencing games revisited. *European Journal of Operations Research*, **174**, 1944–1949.

Slikker, M. (2006b). Relaxed sequencing games have a nonempty core. *Naval Research Logistics*, **53**, 235–242.

Smith, W. (1956). Various optimizers of single-stage production. *Naval Research Logistics Quaterly*, **3**, 59–66.

Suijs, J. (2003). Cost allocation in spanning network enterprises with stochastic connection costs. *Games and Economic Behavior*, **42**, 156–171.

Tijs, S. (1981). Bounds for the core and the τ-value. In O. Moeschlin and D. Pallaschke (Eds.), *Game Theory and Mathematical Economics*, pp. 123–132. North-Holland, Amsterdam.

Velzen, B. van and H. Hamers (2003). On the balancedness of relaxed sequencing games. *Mathematical Methods of Operations Research*, **57**, 287–297.

Walker, A. (1940). A cytoarchitectural study of the prefrontal area of the macaque monkey. *Journal of Comparative Neurology*, **73**, 59–86.

Warshall, S. (1962). A theorem on boolean matrices. *Journal of the ACM*, **9**, 11–12.

Watts, D. and S. Strogatz (1998). Collective dynamics of ‘small-world’ networks. *Nature*, **393**, 440–442.

Widgrén, M. (1994). Voting power in the EC decision making and the consequences of two different enlargements. *European Economic Review*, **38**, 1153–1170.

Young, M. (1992). Objective analysis of the topological organization of the primate cortical visual system. *Nature*, **358**, 152–155.

Author index

Subject index