Tilburg University

Processing Games with Restricted Capacities

Meertens, M.; Borm, P.E.M.; Reijnierse, J.H.; Quant, M.

Publication date:
2004

Link to publication in Tilburg University Research Portal

Citation for published version (APA):
Meertens, M., Borm, P. E. M., Reijnierse, J. H., & Quant, M. (2004). *Processing Games with Restricted Capacities*. (CentER Discussion Paper; Vol. 2004-83). Operations research.

General rights
Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal

Take down policy
If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.

Download date: 22. Jan. 2022

CentER
Discussion Paper

No. 2004–83

PROCESSING GAMES WITH RESTRICTED CAPACITIES

By M. Meertens, P.E.M. Borm, J.H. Reijnierse, M. Quant

September 2004

ISSN 0924-7815

Tilburg University

Processing games with restricted capacities

Marc Meertens[1,2] Peter Borm[3] Marieke Quant[3]
Hans Reijnierse[3]

Abstract

This paper analyzes processing problems and related cooperative games. In a processing problem there is a finite set of jobs, each requiring a specific amount of effort to be completed, whose costs depend linearly on their completion times. There are no restrictions whatsoever on the processing schedule. The main feature of the model is a capacity restriction, i.e., there is a maximum amount of effort per time unit available for handling jobs.

Assigning to each job a player and letting each player have an individual capacity for handling jobs, each coalition of cooperating players in fact faces a processing problem with the coalitional capacity being the sum of the individual capacities of the members. The corresponding processing game summarizes the minimal joint costs for every coalition. It turns out that processing games are totally balanced. An explicit core element is constructed.

Keywords: scheduling, individual capacity, cooperation, core allocation.

JEL Classification Numbers: C63, C71.

1 Introduction

Consider the situation in which a number of jobs has to be completed, each requiring its own amount of effort, and in which there is a capacity constraint to process jobs.

[1]Corresponding author. Email: M.Meertens@science.ru.nl

[2]Department of Mathematics, University of Nijmegen, Toernooiveld 1, 6525 ED Nijmegen, The Netherlands.

[3]CentER and Department of Econometrics & OR, Tilburg University, P.O. Box 90153, 5000 LE Tilburg, The Netherlands.

The terminology has been chosen very general in order to let several interpretations fit. Jobs can involve maintenance problems, the manufacturing of products, computational tasks, or investments under periodic budget raises. Capacity constraints can be induced by limited availability of labor and/or engine power, by periodic supplies of raw material, by maximum computational speed of a computer facility, or by budget. In these examples, effort represents performance of men and/or machinery, or volumes of raw material, calculations or money. In all cases, capacity means the maximum available effort per time unit. It is assumed that for each time unit that a job is uncompleted, a fixed cost has to be paid. The objective is to find a processing schedule, taking the capacity constraint into account, to perform all jobs such that the total costs are minimized. There are no restrictions on the schedule with respect to, for instance, pre-emption, semi-activeness or serial vs parallel planning. We have baptized this type of problem a *processing problem with restricted capacity* or *processing problem* for short.

It turns out that in order to minimize costs in a processing problem with restricted capacity, the jobs have to be performed one by one. So, until all jobs have been completed, all capacity should be used on one job at a time. Thus, it suffices to find an optimal order on the jobs. From this observation it follows that from an operations research point of view, processing problems with restricted capacity and the well-known *sequencing problems with one machine and aggregated (weighted) completion times* are equivalent. Applying Smith's rule (Smith (1956)), i.e. process the jobs in the order of decreasing *urgencies*, provides in both problems an optimal order on the jobs. Here, the urgency of a job is defined to be its cost-coefficient divided by its processing demand.

However, the problems diverge when analyzed in a cooperative game theory framework. Problems are extended to situations in which each job belongs to a (different) player and each player has a personal capacity to handle jobs. Besides minimizing total costs, costs have to be allocated to each player individually. In order to find fair allocations, a cooperative game is constructed. The approach to associate a cooperative game to an operation research problem is quite common in the literature (see Borm, Hamers and Hendrickx (2001) for an overview). Sequencing problems with one machine have been analyzed from a game-theoretical point of view in several ways, starting from the basic paper by Curiel, Pederzoli and Tijs (1989) (see Curiel, Hamers and Klijn (2002) for an overview). In this paper we associate games to processing situations. These games are called *processing games (with restricted capacities)* and differ from sequencing games. This diversion is due to two main differences between a processing and a sequencing situation. The first difference is that in a processing situation the players have *individual* and generally different capacities to handle jobs, while in a sequencing situation with one machine there are *no* individual capacities: in fact, the machine processes all jobs with a constant capacity. The second difference is that in a processing situation with restricted capacities there is *no* fixed initial order in which the jobs stand in line in front of a machine. So, in a processing situation there are *no* initial restrictions nor rights on the order in which players may process their jobs.

Let us elaborate on the way players can cooperate in processing situations with restricted capacities. If a coalition is formed, costs savings can be made by helping each other by means of using a player's capacity to speed up the job of another coalition member. To put it differently, the members of the coalition have to their disposal the sum of their individual capacities in order to complete all jobs of the coalition. This situation can be modelled as a processing problem and as a result one can easily determine an optimal schedule and its costs. However, the problem of minimizing the total costs is supplemented with the problem of dividing these costs among the players involved. The latter is of a typical game theoretical nature and in order to solve it, we analyze the complete processing game with respect to core elements. Here, a processing game is a cooperative cost-game, in which the costs of a coalition equal the costs of an optimal schedule of its corresponding processing problem.

The main result of this paper states that every processing game with restricted capacities is *totally balanced*, i.e., every subgame of a processing game has a non-empty core. To prove this statement, we construct from a given processing situation an *exchange economy with land*. In this Debreu-type of exchange economy (Debreu (1959)) each player initially owns a part of a perfectly divisible two dimensional commodity, referred to as *land*. One dimension is time and the other one is effort per time unit. In the context of processing situations, one can interpret this commodity as an *agenda*. In order to complete their jobs, players must make reservations in the agenda, i.e., a player must book a block of time and capacity, which is sufficiently large to complete his job. A price is introduced such that the market clears, i.e., no part of the agenda is booked by more than one player. Clearing the market will, as usual, lead to a *price equilibrium*. From this price equilibrium, we construct an allocation contained in the core of the processing game. Hence, we explicitly provide a core allocation for every processing game. Since a subgame of a processing game is again a processing game, we obtain totally balancedness. Furthermore, an interpretation of this core allocation is included, along with a proof that this core allocation is independent on which optimal schedule is chosen (in case of coinciding urgencies). As a consequence, the allocation x depends continuously on the processing times, capacities and costs coefficients.

The paper is organized as follows. In Section 2 we introduce the formal model of a processing problem with restricted capacity. Section 3 studies total balancedness of processing games. Section 4 provides a proof for the main result, following the line described above.

2 Processing problems with restricted capacity

A processing problem with restricted capacity $\mathcal{P}$ can be described formally by a tuple

$$\langle J, p = (p_j)_{j \in J}, \alpha = (\alpha_j)_{j \in J}, \beta \rangle.$$

Here, J is a finite set of jobs that need to be completed. The vector p in $\mathbb{R}^J_+$ contains the processing demands or efforts of the jobs, furthermore α in $\mathbb{R}^J_+$ is the vector of cost coefficients and β is a strictly positive real denoting the maximum available effort per time unit, or shortly capacity. The costs for job j to be uncompleted for a period of time t equals $\alpha_j \cdot t$. A *feasible schedule* can be described by a map

$$F : J \times \mathbb{R}_+ \longrightarrow \mathbb{R}_+,$$

with the following properties:

(i) $F(j,t) : \mathbb{R}_+ \longrightarrow \mathbb{R}_+$ is continuous and weakly increasing in t for all $j \in J$,

(ii) $\sum_{j \in J} \big[F(j,t) - F(j,s)\big] \ \leq \ \beta \cdot (t-s)$ for every $s, t \in \mathbb{R}_+$ with $s \leq t$.

The value $F(j,t)$ for job j in J at time t in $\mathbb{R}_+$, can be interpreted as the cumulative amount of effort which has been used for job j up to time t. Property (ii) states that for each segment $[s,t]$ of time, the total effort spend on all jobs together is restricted linearly in the length of the segment by the capacity constraint. We denote $\mathcal{F}$ as the family of all feasible schedules. Given a feasible schedule F in $\mathcal{F}$, the *completion time* $T_j(F)$ of job j in J is defined by

$$T_j(F) \ := \ \inf\{t \in \mathbb{R}_+ \mid F(j,t) \ \geq \ p_j\}.$$

We allow $T_j(F)$ to be infinity. The objective is to find a feasible schedule such that the sum of costs over all jobs is minimized. This minimum is expressed by

$$c(\mathcal{P}) \ := \ \min_{F \in \mathcal{F}} \sum_{j \in J} \alpha_j \cdot T_j(F).$$

Observe that the minimum exists and therefore the value $c(\mathcal{P})$ is well-defined for every processing problem $\mathcal{P}$.

Example 2.1. Suppose, a farmer has to harvest three acres with different types of crop, say type 1, 2 and 3. The tasks require 20, 30 and 10 days of work for one man respectively. His workforce consists of himself and 5 employees. He has contracts with distributors to deliver the types of crop, but he is already over time. Every extra day of delay results in penalties of size 24, 30 and 6 respectively. The farmer wants to harvest the acres in such a way that the total sum of penalties will be minimal. This problem can be modelled as the processing problem $\mathcal{P} := \langle J, p, \alpha, \beta \rangle$, in which $J := \{1, 2, 3\}$, $p := (20, 30, 10)$, $\alpha := (24, 30, 6)$ and $\beta := 6$.

One approach to complete the jobs, is dividing the capacity β over the jobs proportionally to their processing demands. Then after 10 days all jobs are finished simultaneously. This approach corresponds with the schedule F defined as follows:

$$F(1,t) := 2{\cdot}t, \quad F(2,t) := 3{\cdot}t \;\text{ and }\; F(3,t) := t \;\text{ for all } t \in \mathbb{R}_+.$$

It yields a total cost of 600. Another approach is to finish the jobs one after another. If the jobs are done in the order $(1, 2, 3)$, the corresponding schedule F' will be

$$F'(1,t) := \begin{cases} 6{\cdot}t & \text{if } \; 6{\cdot}t \in [0,20], \\ 20 & \text{if } \; 6{\cdot}t \geq 20, \end{cases}$$

$$F'(2,t) := \begin{cases} 0 & \text{if } \; 6{\cdot}t \in [0,20], \\ 6{\cdot}t - 20 & \text{if } \; 6{\cdot}t \in [20,50], \\ 30 & \text{if } \; 6{\cdot}t \geq 50, \end{cases}$$

$$F'(3,t) := \begin{cases} 0 & \text{if } \; 6{\cdot}t \in [0,50], \\ 6{\cdot}t - 50 & \text{if } \; 6{\cdot}t \in [50,60], \\ 10 & \text{if } \; 6{\cdot}t \geq 60. \end{cases}$$

The schedule F' induces completion times $T(F') = (\frac{20}{6}, \frac{50}{6}, 10)$, which yield a total cost of 390. ◦

In the example above is already illustrated that it may be profitable to use the total capacity for exactly one job at a time. We will now demonstrate that in order to minimize aggregate costs in an arbitrary processing problem, one should indeed choose this approach. To do so, we first need some preparations.

Let $\sigma : \{1, \ldots, |J|\} \longrightarrow J$ be a bijection. It can be seen as the order in which the jobs in J are completed, i.e., the job at position j in the order σ is denoted by $\sigma(j)$. We write $\Pi(J)$ for the family of all such bijections. In case the jobs in J are completed in the order σ, we get as corresponding schedule $F^\sigma : J \times \mathbb{R}_+ \longrightarrow \mathbb{R}_+$:

$$F^\sigma(\sigma(j),t) \;:=\; \begin{cases} 0 & \text{if} \qquad\qquad\quad \beta\cdot t \;\leq\; \sum_{k<j} p_{\sigma(k)}, \\ \beta\cdot t - \sum_{k<j} p_{\sigma(k)} & \text{if} \quad \sum_{k<j} p_{\sigma(k)} \;\leq\; \beta\cdot t \;\leq\; \sum_{k\leq j} p_{\sigma(k)}, \\ p_{\sigma(j)} & \text{if} \quad \sum_{k\leq j} p_{\sigma(k)} \;\leq\; \beta\cdot t. \end{cases}$$

Lemma 2.1. *There exists a bijection $\sigma \in \Pi(J)$ such that $\sum_{j\in J} \alpha_j \cdot T_j(F^\sigma) = c(\mathcal{P})$.*

Proof. Let $F \in \mathcal{F}$ be an optimal schedule, then $T_j(F) < \infty$ for all j in J. Let $\sigma \in \Pi(J)$ be a bijection such that

$$T_{\sigma(1)}(F) \;\leq\; T_{\sigma(2)}(F) \;\leq \ldots \leq\; T_{\sigma(|J|)}(F).$$

For $1 \leq j \leq |J|$ we have

$$T_{\sigma(j)}(F) \;\geq\; \tfrac{1}{\beta} \sum_{k=1}^{j} p_{\sigma(k)} \;=\; T_{\sigma(j)}(F^\sigma).$$

This yields

$$c(\mathcal{P}) \;=\; \sum_{j=1}^{|J|} \alpha_{\sigma(j)} \cdot T_{\sigma(j)}(F) \;\geq\; \sum_{j=1}^{|J|} \alpha_{\sigma(j)} \cdot T_{\sigma(j)}(F^\sigma).$$

Hence, $c(\mathcal{P}) = \min_{\sigma\in\Pi(J)} \sum_{j\in J} \alpha_j \cdot T_j(F^\sigma)$. □

This result in fact shows that a processing problem boils down to a sequencing problem with one machine. The processing time of a job j can be found by dividing the processing demand p_j by the capacity constraint β. Therefore, the optimal schedule can be found by applying the well-known *Smith's rule*, i.e. process the jobs in the order of decreasing urgencies, in which the urgency of job j in J is given by $\frac{\alpha_j}{p_j}$.

Proposition 2.2. [cf. Smith (1956)] *Let $\mathcal{P}$ be a processing problem such that $J = \{1, \ldots, |J|\}$ and the jobs are numbered such that $\frac{\alpha_1}{p_1} \geq \cdots \geq \frac{\alpha_{|J|}}{p_{|J|}}$. Then it is optimal to process the jobs in increasing order and*

$$c(\mathcal{P}) \;=\; \tfrac{1}{\beta} \sum_{i=1}^{|J|} \alpha_i \cdot [p_1 + \ldots + p_i].$$ □

3 Processing games with restricted capacities

In this section we introduce processing situations with restricted capacities and associated processing games. Examples are provided and the main theorem of the paper is stated: processing games are totally balanced. An explicit core element is provided.

In a *processing situation* $\langle N, J, p = (p_j)_{j\in J}, \alpha = (\alpha_j)_{j\in J}, (\beta_i)_{i\in N}\rangle$ there is a finite set of *players* N, in which each player i in N is endowed with a strictly positive *capacity* β_i, in order to perform jobs. Each *job* j in J has a *processing demand* p_j and cost coefficient α_j, both in $\mathbb{R}_+$. As long as job j is uncompleted, it generates a cost of size α_j per time unit. Each player has to complete one specific job in J.

Since each player is obliged to a different job, there is a one-one correspondence between players an jobs and no confusion occurs when the processing demand and the cost coefficient of the job of player i are denoted by p_i and α_i respectively. This one-one correspondence simplifies notations and proofs, but is not essential for our results. The model can be extended to situations in which players are obliged to several jobs or jobs are in the interest of several players. These generalizations lie beyond the scope of this paper and will be studied in Quant, Meertens and Reijnierse (2004).

Let $S \subseteq N$ be a coalition of players who decide to cooperate. This coalition has the disposal of the individual capacities of all of its members, i.e., coalition S can maximally generate an amount of effort of size $\beta(S) := \sum_{i\in S}\beta_i$ per time unit. The aim of coalition S is to complete all jobs of its members, such that aggregate costs are minimized. This situation gives rise to the processing problem

$$\mathcal{P}(S) := \langle J(S), (p_i)_{i\in S}, (\alpha_i)_{i\in S}, \beta(S)\rangle,$$

in which $J(S)$ denotes the set of jobs corresponding to players in S. Proposition 2.2 provides a method to calculate an optimal schedule such that the aggregate costs of coalition S are minimized. However, constructing such a schedule is only part of the problem. That is, in addition to minimizing total costs, the problem remains how to allocate these costs among the players in S in a fair way. To study this problem, we analyze a *processing game* $\langle N, c^{\mathcal{P}}\rangle$ in which $c^{\mathcal{P}} : 2^N \longrightarrow \mathbb{R}_+$ is the map defined by

$$c^{\mathcal{P}}(S) \ := \ c(\mathcal{P}(S)) \ \text{ for all } S \subseteq N.$$

The processing game $\langle N, c^{\mathcal{P}}\rangle$ is a so-called *cost-game.* A cost-game is a transferable utility game, or TU-game, but in stead of rewards it depicts costs of coalitions. Because of the different interpretation, the definitions of standard properties and solution concepts for TU-games have to be adjusted. For the sake of completeness we provide the concepts that will be considered. The *core* of a cost-game $\langle N, c\rangle$ is defined by

$$\mathcal{C}(c) \;:=\; \{x \in \mathbb{R}^N \mid \sum_{i\in N} x_i \;=\; c(N) \text{ and } \sum_{i\in S} x_i \;\le\; c(S) \text{ for all } S \subset N\}.$$

A game with a non-empty core is called *balanced*. A TU-game is said to be *totally balanced* whenever every subgame[1] has a non-empty core. A cost-game $\langle N, c\rangle$ is said to be *concave* if $c(S) + c(T) \ge c(S\cup T) + c(S\cap T)$ for all $S, T \subseteq N$. As is well-known, concavity of a cost-game implies (totally) balancedness.

The main goal of this paper is to prove that every processing game has a non-empty core. The core is generally considered to be a normative for an allocation to be fair, since *no* coalition can improve with respect to a core allocation.

We revisit Example 2.1 to show a processing game $\langle N, c^{\mathcal{P}}\rangle$. It points out in particular that a processing game is in general *not* concave and that there can be players to whom are assigned a negative cost (i.e., a reward) in any core allocation. As a result, solutions based on a proportional type of costs allocation with respect to processing demands and/or capacities will in general *not* generate core allocations.

Example 2.1 (continued). This time the three acres are owned by different farmers. Farmers 1 and 2 have small farms and no employees. Farmer 3 has 3 employees.

In the processing situation comporting with the story, the player set N consists of the players 1, 2 and 3 of which the processing demands are given by $p := (20, 30, 10)$, the cost coefficients are given by $\alpha := (24, 30, 6)$ and the individual capacities of the players are $\beta := (1, 1, 4)$. Observe that the players are numbered in such a way that $\frac{\alpha_1}{p_1} \ge \frac{\alpha_2}{p_2} \ge \frac{\alpha_3}{p_3}$. According to Proposition 2.2, the corresponding processing game $\langle N, c^{\mathcal{P}}\rangle$ is given by

$$c^{\mathcal{P}}(N) \;=\; \tfrac{1}{6}\big[24\cdot 20 + 30\cdot(20+30) + 6\cdot(20+30+10)\big] \;=\; 390,$$

$$c^{\mathcal{P}}(\{1,2\}) \;=\; 990,\;\; c^{\mathcal{P}}(\{1,3\}) \;=\; 132,\;\; c^{\mathcal{P}}(\{2,3\}) \;=\; 228,$$

$$c^{\mathcal{P}}(\{1\}) \;=\; 480,\;\; c^{\mathcal{P}}(\{2\}) \;=\; 900 \text{ and } \;c^{\mathcal{P}}(\{3\}) \;=\; 15.$$

Observe that $c^{\mathcal{P}}(N) + c^{\mathcal{P}}(\{2\}) = 1290 > 1218 = c^{\mathcal{P}}(\{1,2\}) + c^{\mathcal{P}}(\{2,3\})$, so the game is not concave. Furthermore, if $x \in \mathcal{C}(c^{\mathcal{P}})$,

$$\begin{aligned} x_1 + x_3 &\;\le\; 132, \\ x_2 + x_3 &\;\le\; 228, \\ x_1 + x_2 + x_3 &\;=\; 390. \end{aligned}$$

[1]For each coalition S, the subgame $\langle S, c_{|S}\rangle$ of $\langle N, c\rangle$ is defined by $c_{|S}(T) = c(T)$ for all $T \subseteq S$.

Hence, $390 + x_3 = x_1 + x_2 + 2 \cdot x_3 \leq 360$. As a result, this yields $x_3 < 0$. Note that player 3 is rewarded for his participation in every core allocation because of his relatively large capacity. It is left to the reader to verify that the allocation $(195, 310, -115)$ is contained in the core $\mathcal{C}(c^{\mathcal{P}})$. $\circ$

The core allocation of the example above has been found by applying the proof of the following theorem, which is the main result of this paper.

Theorem 3.1. *A processing game is totally balanced.* $\square$

A proof can be found in Section 4, in which the allocation x in $\mathbb{R}^N$, given by

$$x_i := \tfrac{\alpha_i}{\beta(N)} \sum_{k=1}^{i} p_k + \tfrac{p_i}{\beta(N)} \cdot \big[\, \tfrac{1}{2} \cdot \alpha_i + \sum_{k=i+1}^{n} \alpha_k \big] - \tfrac{\beta_i}{\beta(N)} \sum_{k=1}^{n} \tfrac{p_k}{\beta(N)} \cdot \big[\tfrac{1}{2} \cdot \alpha_k + \sum_{\ell=k+1}^{n} \alpha_\ell \big]$$

for all $i \in N$, will be proven to be a core allocation of the processing game $\langle N, c^{\mathcal{P}} \rangle$, provided that $N := \{1, \ldots, n\}$ and $\frac{\alpha_1}{p_1} \geq \ldots \geq \frac{\alpha_n}{p_n}$.

Let us give an interpretation of this core allocation. Since the urgencies are ordered in the way described above it is optimal for the grand coalition N to first use the total capacity $\beta(N)$ on the job of player 1, then on the job of player 2 and so on. According to this schedule, player i has to wait for a period of time with length $\frac{1}{\beta(N)} \sum_{k=1}^{i} p_k$ until his job has been completed. As a result his individual direct costs will be

$$\alpha_i \cdot \tfrac{1}{\beta(N)} \sum_{k=1}^{i} p_k. \tag{1}$$

If each player i would pay this amount, the costs are divided in an efficient way. It would not be very fair though. A player whose job is placed at the end of the line should be compensated. Furthermore, players who have a relatively large capacity β_i should be rewarded. For this reason, besides the actual costs (1), a *tax* is introduced on the jobs. The tax proceeds then will be used to subsidize the players with large capacities. More particularly, the sum of the tax deposits is redivided proportionally to the capacities of the players. We now explain the reasoning behind the explicit format of the tax deposits. At each moment of time t, a cost-rate is introduced. The player whose job is in process must pay this rate. The cost-rate at time t equals

$$\sum_{k \in N} \alpha_k \cdot [\text{the proportion of job } j_k \text{ that has not been finished yet at time } t].$$

During a period of time with length $\frac{p_i}{\beta(N)}$, all players are working on the job of player i. Player i must pay $\alpha_k \cdot \frac{p_i}{\beta(N)}$ for each player k whose job is still waiting to be processed. This is exactly the loss of player k because of the fact that the job of player i is processed before his own job. Additionally, player i has to pay $\frac{1}{2} \cdot \alpha_i \cdot \frac{p_i}{\beta(N)}$, since the mean proportion of his own job that has not been finished yet during its processing time equals $\frac{1}{2}$. The sum of these amounts equals the *tax deposit* τ_i of player i:

$$\tau_i := \tfrac{p_i}{\beta(N)} \cdot \left[\tfrac{1}{2} \cdot \alpha_i + \sum_{k=i+1}^{n} \alpha_k\right]. \tag{2}$$

Finally, the total amount of collected tax money is returned to the players, proportional to their individual capacities. This yields a *subsidy* for player i of

$$\tfrac{\beta_i}{\beta(N)} \sum_{k=1}^{n} \tau_k. \tag{3}$$

Subtracting expression (3) from the sum of the expressions (1) and (2), yields the amount player i has to pay according to the core allocation x:

$$x_i \;:=\; \tfrac{\alpha_i}{\beta(N)} \sum_{k=1}^{i} p_k + \tau_i - \tfrac{\beta_i}{\beta(N)} \sum_{k=1}^{n} \tau_k, \tag{4}$$

Let us return once more to the processing situation arising from Example 2.1.

Example 2.1 (continued). Let $\langle N, J, p, \alpha, \beta \rangle$ be the processing situation with $N := \{1, 2, 3\}$, $J := \{j_1, j_2, j_3\}$, $p := (20, 30, 10)$, $\alpha := (24, 30, 6)$ and $\beta := (1, 1, 4)$. The players are numbered in such a way that the optimal order is $(1, 2, 3)$. We already stressed out that the allocation $(195, 310, -115)$ is a core allocation of the corresponding processing game. This allocation arises as follows. The first part consists of the individual direct costs of (expression (1)) and equals

$$24 \cdot \tfrac{1}{6} \cdot 20 = 80,\; 30 \cdot \tfrac{1}{6} \cdot 50 = 250 \text{ and } 6 \cdot \tfrac{1}{6} \cdot 60 = 60.$$

The tax that the players have to pay is (expression (2))

$\tau_1 = \frac{20}{6} \cdot [\frac{1}{2} \cdot 24 + 30 + 6] = 160,\; \tau_2 = \frac{30}{6} \cdot [\frac{1}{2} \cdot 30 + 6] = 105$ and $\tau_3 = \frac{10}{6} \cdot [\frac{1}{2} \cdot 6] = 5.$

According to expression (3) the players are subsidized

$$\tfrac{1}{6} \cdot [\tau_1 + \tau_2 + \tau_3] = 45,\; \tfrac{1}{6} \cdot [\tau_1 + \tau_2 + \tau_3] = 45 \text{ and } \tfrac{4}{6} \cdot [\tau_1 + \tau_2 + \tau_3] = 180$$

respectively. The core allocation x becomes

$$(80 + 160 - 45\;,\; 250 + 105 - 45\;,\; 60 + 5 - 180) \;=\; (195\;,\; 310\;,\; -115). \qquad \circ$$

Observe that the direct costs as well as the tax deposits are based on the given optimal order of decreasing urgencies. At first sight, the core allocation x depends therefore on the optimal order chosen. The following proposition shows that this is *n*ot the case.

Proposition 3.2. *Let $\langle N, J, p, \alpha, \beta \rangle$ be a processing situation. The core allocation x as given by* (4) *does not depend on the choice of which optimal order is used to process the jobs.*

Proof: Two optimal orders be obtained from each other by a series of switches of two adjacent jobs with equal urgencies. It is sufficient to show that x does not change at each of these switches. Assume that one optimal order is $(1, \ldots, n)$ and that players i and $i+1$ have equal urgencies: $\frac{\alpha_i}{p_i} = \frac{\alpha_{i+1}}{p_{i+1}}$. We have to show that x and $\bar{x}$ coincide, with x and $\bar{x}$ denoting the allocations which correspond to the orders $(1, \ldots, n)$ and $(1, \ldots, i-1, i+1, i, i+2, \ldots, n)$, where i and $i+1$ have been switched, respectively. The vectors of taxes corresponding to these orders are denoted by τ and $\bar{\tau}$ respectively.

We first show that the total amount of taxes paid in both orders is the same. Note that for players k unequal to i and $i+1$, the taxes τ_k and $\bar{\tau}_k$ coincide. It is shown below that the sum of the taxes paid by i and $i+1$ does not change either.

$$\begin{aligned}
\tau_i + \tau_{i+1} &= \tfrac{p_i}{\beta(N)}(\tfrac{1}{2}\alpha_i + \sum_{\ell=i+1}^{n} \alpha_\ell) + \tfrac{p_{i+1}}{\beta(N)}(\tfrac{1}{2}\alpha_{i+1} + \sum_{\ell=i+2}^{n} \alpha_\ell) \\
&= \tfrac{p_i}{\beta(N)}(\tfrac{1}{2}\alpha_i + \sum_{\ell=i+2}^{n} \alpha_\ell) + \tfrac{p_i}{\beta(N)} \cdot \alpha_{i+1} + \tfrac{p_{i+1}}{\beta(N)}(\tfrac{1}{2}\alpha_{i+2} + \sum_{\ell=i+2}^{n} \alpha_\ell) \\
&= \tfrac{p_i}{\beta(N)}(\tfrac{1}{2}\alpha_i + \sum_{\ell=i+2}^{n} \alpha_\ell) + \tfrac{p_{i+1}}{\beta(N)} \cdot \alpha_i + \tfrac{p_{i+1}}{\beta(N)}(\tfrac{1}{2}\alpha_{i+1} + \sum_{\ell=i+2}^{n} \alpha_\ell) \\
&= \bar{\tau}_i + \bar{\tau}_{i+1}.
\end{aligned}$$

The third equality follows from the fact that i and $i+1$ have equal urgencies. Since the total sum of amount of taxes is equal in both orders, it is immediately clear that $x_k = \bar{x}_k$ for all k unequal to i and $i+1$. We now prove that $x_i = \bar{x}_i$.

$$\begin{aligned}
x_i &= \tfrac{\alpha_i}{\beta(N)} \sum_{k=1}^{i} p_k + \tau_i - \tfrac{\beta_i}{\beta(N)} \sum_{k=1}^{n} \tau_k \\
&= \tfrac{\alpha_i}{\beta(N)} \sum_{k=1}^{i} p_k + \tfrac{p_i}{\beta(N)} \cdot \alpha_{i+1} + \tfrac{p_i}{\beta(N)}(\tfrac{1}{2}\alpha_i + \sum_{k=i+2}^{n} \alpha_k) - \tfrac{\beta_i}{\beta(N)} \sum_{k=1}^{n} \bar{\tau}_k \\
&= \tfrac{\alpha_i}{\beta(N)} \sum_{k=1}^{i} p_k + \tfrac{p_{i+1}}{\beta(N)} \cdot \alpha_i + \bar{\tau}_i - \tfrac{\beta_i}{\beta(N)} \sum_{k=1}^{n} \bar{\tau}_k \\
&= \bar{x}_i.
\end{aligned}$$

The third equality uses the fact that i and $i+1$ have the same urgency. In the same way it can be proved that x_{i+1} and $\bar{x}_{i+1}$ coincide. $\square$

Because of this result, the allocation x is uniquely determined by α, β and p. It is clear that x is continuous in points with just one optimal order. Proposition 3.2 shows that it is also continuous in points with more than one order.

Corollary 3.3. *The core allocation x is continuous in α, β and p.*

Before closing this section we will look at a particular class of cost-games that can be derived from processing situations. As a result the games within this class are totally balanced.

Example 3.1. Let N be a set of players. Consider the situation in which all jobs have equal cost coefficients and equal processing times, say all of size 1. Let $\beta \in \mathbb{R}^N_+$ be a strictly positive vector. The corresponding processing situation yields the cost-game $\langle N, c \rangle$, defined by

$$c(S) \; := \; \tfrac{1}{\beta(S)} \cdot \frac{|S| \cdot (|S|+1)}{2} \quad \text{for all } S \subseteq N.$$

This can be easily verified by applying Proposition 2.2 (notice that all players have urgency 1). In order to determine the core allocation x, any order can be used since they are all optimal. If we choose the order $(1, \ldots, n)$, the direct costs of player i are $\frac{i}{\beta(N)}$. His tax τ_i equals $\frac{1}{\beta(N)}(\frac{1}{2} + n - i)$. The aggregate of the taxes $\sum_{i \in N} \tau_i$ equals $\frac{n^2}{2\beta(N)}$. Hence, for all i in N we have

$$x_i = \tfrac{1}{\beta(N)}(n + \tfrac{1}{2}) - \tfrac{\beta_i}{\beta(N)} \frac{n^2}{2\beta(N)}. \qquad \circ$$

4 Proof of Theorem 3.1

Let us first give an outline of the proof. Given a processing situation, we construct an exchange economy and find a price equilibrium. This equilibrium is situated in the core of the economy. It induces a core allocation of the corresponding processing game. A similar technique has been used in Klijn, Tijs and Hamers (2000) to construct core elements of *permutation games*.

An (initially) empty agenda is given. It will be a two-dimensional commodity. Of course time is one dimension, the other one is effort per time unit. In principle, there is no time restriction. The amount of effort per time unit is bounded by the capacity $\beta(N)$ of the grand coalition. At each moment of time, one can buy any (measurable) part of the capacity available. Because of the two dimensions, it is customary to speak of *land* rather than of an agenda. So, we consider a Debreu-type of exchange economy (Debreu (1959)) in which each player initially owns a part of a perfectly divisible good, land. This type of economies has been studied in several papers (see for instance Legut, Potters and Tijs (1994)). In order to complete their jobs, players must make reservations in the agenda. Only if a player books a block of time and effort per time unit sufficiently large to process his job, it will be completed. A price is chosen such that the market clears, i.e., no part of the agenda is booked by more than one player. This gives rise to a feasible schedule $F \in \mathcal{F}$. Clearing the market will, as usual, lead to a price equilibrium, which is situated in the core of the exchange economy. It is converted to a core element of the processing game. This will end the proof.

Let $\langle N, J, p, \alpha, \beta \rangle$ be a processing situation. Throughout this section we assume, without loss of generality, that $N = \{1, \ldots, n\}$ and that $\frac{\alpha_1}{p_1} \geq \cdots \geq \frac{\alpha_n}{p_n}$.

Let $\mathcal{E}(\mathcal{P}) := \langle N, (L, \mathcal{B}, \lambda), (A_i, V_i)_{i \in N} \rangle$ be an *exchange economy with land*, in which:

- A commodity, modelled by a measure space $(L, \mathcal{B}, \lambda)$ has to be reallocated among the group of players N. Here, $L := [0, \beta(N)] \times \mathbb{R}_+$ denotes a piece of *land*, $\mathcal{B}$ is the *Borel-σ-algebra* of L and $\lambda : \mathcal{B} \longrightarrow \mathbb{R}_+$ denotes the *Lebesgue-measure* on L.

- Each player i in N has *endowment* $A_i := [\beta_i] \times \mathbb{R}_+$ in which $[\beta_i]$ denotes the interval $[\sum_{k<i} \beta_i, \sum_{k \leq i} \beta_i]$. Observe that $\bigcup_{i \in N} A_i = L$ and $\lambda(A_i \cap A_k) = 0$ whenever $i \neq k$.

- Each player i has *reservation value* $V_i(B)$ for all sets B in $\mathcal{B}$ defined by

$$V_i(B) \ := \ -\alpha_i \cdot T_i(B),$$

  in which $T_i(B) := \inf\{t \in \mathbb{R}_+ \mid \int_0^t \int_0^{\beta(N)} 1_B(x, \tau) \, \mathrm{d}x\mathrm{d}\tau \ \geq \ p_i\}$. Here, $1_B(\cdot)$ is the indicator-function of set B. $T_i(B)$ denotes the moment of time at which the job of player i will be finished in the case that part B of the land (agenda) is used to work on his job. In case subset B is not sufficient, $T_i(B)$ equals infinity.

- Player i has a *quasi-linear utility function* $U_i : \mathcal{B} \times \mathbb{R} \longrightarrow \mathbb{R}$ that denotes his valuation for bundles of land and money. It is defined by

$$U_i(B, y) \ := \ V_i(B) + y \ \text{ for all } B \in \mathcal{B} \text{ and } y \in \mathbb{R}.$$

Let $S \subseteq N$ be a coalition. An *S-redistribution* is a set $\{B_i\}_{i \in S}$ of λ-measurable subsets of L, with $\bigcup_{i \in S} B_i \subseteq \bigcup_{i \in S} A_i$ and $\lambda(B_i \cap B_j) = 0$ whenever $i \neq j$. Let $\{B_i\}_{i \in S}$ be an S-redistribution and $z \in \mathbb{R}^S$ such that $\sum_{i \in S} z_i = 0$. Then the set $\{(B_i, z_i)\}_{i \in S}$ is called an *S-reallocation*.

If $\{(B_i, z_i)\}_{i\in N}$ is an N-reallocation then an S-reallocation $\{(C_i, y_i)\}_{i\in S}$ is called an *improvement* upon $\{(B_i, z_i)\}_{i\in N}$ if $V_i(C_i) + y_i \; > \; V_i(B_i) + z_i$ for all i in S. An N-reallocation $\{(B_i, z_i)\}_{i\in N}$ is a *core reallocation* if *no* coalition S has an improvement upon $\{(B_i, z_i)\}_{i\in N}$.

An N-reallocation $\{(B_i, z_i)\}_{i\in N}$ is called a *price equilibrium* if there exists a price density function $\pi : L \longrightarrow \mathbb{R}$ such that

(i) $P_\pi(B_i) + z_i \; = \; P_\pi(A_i)$ for all $i \in N$, (budget constraints)

(ii) If $V_i(C) + y \; > \; V_i(B_i) + z_i$ for certain $C \subseteq L$, $y \in \mathbb{R}$ and $i \in N$, then $P_\pi(C) + y > P_\pi(A_i)$, (maximality conditions)

in which $P_\pi(B) := \int_0^\infty \int_0^{\beta(N)} 1_B(x,t) \cdot \pi(x,t)\, \mathrm{d}x\mathrm{d}t$ for all $B \subseteq L$.

Given an exchange economy with land $\mathcal{E}(\mathcal{P})$ we define a TU-game $\langle N, v_{\mathcal{E}(\mathcal{P})}\rangle$ with the value for coalition $S \subseteq N$ as follows:

$$v_{\mathcal{E}(\mathcal{P})}(S) \; := \; \sup\{\sum_{i\in S} V_i(C_i) \mid \{C_i\}_{i\in S} \text{ is an } S\text{-redistribution}\},$$

i.e., the *maximum social welfare* in the sub-economy in which only the actions of the players in coalition S are considered.

The TU-game $\langle N, v_{\mathcal{E}(\mathcal{P})}\rangle$ is in fact the TU-game $\langle N, -c^{\mathcal{P}}\rangle$, as the following lemma demonstrates.

Lemma 4.1. *$v_{\mathcal{E}(\mathcal{P})}(S) \; = \; -c^{\mathcal{P}}(S)$ for all $S \subseteq N$.*

Proof. Take $S \subseteq N$, say $S := \{i(1), \ldots, i(s)\}$ with $i(1) < \ldots < i(s)$. Define, for all j with $1 \leq j \leq s$:,

$$B_j \;\; := \;\; \bigcup_{i\in S} [\beta_i] \times \tfrac{1}{\beta(S)} \cdot [p_{i(j)}],$$

in which $[p_{i(j)}]$ denotes the interval $[\sum_{\ell<j} p_{i(\ell)}, \sum_{\ell\leq j} p_{i(\ell)}]$ with length $p_{i(j)}$.

Clearly, $\{B_j\}_{1\leq j\leq s}$ is an S-redistribution. Furthermore, it is easy to verify that

$$v_{\mathcal{E}(\mathcal{P})}(S) \; = \; \sum_{j=1}^{s} V_{i(j)}(B_j).$$

Hence,

$$\begin{aligned} v_{\mathcal{E}(\mathcal{P})}(S) \; = \; \sum_{j=1}^{s} V_{i(j)}(B_j) \;\; &= \;\; -\sum_{j=1}^{s} \alpha_{i(j)} \cdot T(B_j) \\ &= \;\; -\sum_{j=1}^{s} \tfrac{\alpha_{i(j)}}{\beta(S)} [p_{i(1)} + \ldots + p_{i(j)}] \; = \; -\, c^{\mathcal{P}}(S). \quad \square \end{aligned}$$

The following lemma provides a relation between the existence of a price equilibrium in the exchange economy $\mathcal{E}(\mathcal{P})$ and the non-emptiness of the core of the TU-game $\langle N, -c^{\mathcal{P}} \rangle$. In fact, using a standard argument, one can show that if $\{(B_i, z_i)\}_{i \in N}$ is a price equilibrium, then the corresponding vector $(V_i(B_i) + z_i)_{i \in N}$ is contained in the core $\mathcal{C}(-c^{\mathcal{P}})$. For the sake of completeness, we provide a proof.

Lemma 4.2. *If the exchange economy $\mathcal{E}(\mathcal{P})$ has a price equilibrium, then the TU-game $\langle N, -c^{\mathcal{P}} \rangle$ has a non-empty core.* □

Proof. Let $\{(B_i, z_i)\}_{i \in N}$ be a price equilibrium supported by the price density function $\pi : L \longrightarrow \mathbb{R}$. We prove that the vector $(V_i(B_i) + z_i)_{i \in N}$ is contained in the core $\mathcal{C}(-c^{\mathcal{P}})$ of the TU-game $\langle N, -c^{\mathcal{P}} \rangle$.

Suppose there exists a coalition $S \subseteq N$ such that

$$\sum_{i \in S} (V_i(B_i) + z_i) \ < \ -c^{\mathcal{P}}(S).$$

Let $\{C_i\}_{i \in S}$ be an S-redistribution such that $v_{\mathcal{E}(\mathcal{P})}(S) = -c^{\mathcal{P}}(S) = \sum_{i \in S} V_i(C_i)$ and define, for all i in S,

$$y_i \ := \ V_i(B_i) + z_i - V_i(C_i) + \tfrac{1}{|S|} \sum_{i \in S} [V_i(C_i) - V_i(B_i) - z_i].$$

It is straightforward to verify that the S-reallocation $\{(C_i, y_i)\}_{i \in S}$ is an improvement upon the price equilibrium $\{(B_i, z_i)\}_{i \in N}$. Hence, according to the maximality conditions, this yields

$$P_\pi(C_i) + y_i \ > \ P_\pi(A_i) \ \text{ for all } i \in S.$$

Taking the sum over all i in S yields the desired contradiction. □

So, the existence of a price equilibrium in $\mathcal{E}(\mathcal{P})$ implies balancedness of the TU-game $\langle N, -c^{\mathcal{P}} \rangle$ and thus also balancedness of the cost-game $\langle N, c^{\mathcal{P}} \rangle$. Therefore, the proof of Theorem 3.1 boils down to the following proposition.

Proposition 4.3. *The exchange economy with land $\mathcal{E}(\mathcal{P})$ has a price equilibrium.*

Proof. Denote $[p_i]$ as the interval $[\sum_{j<i} p_j, \sum_{j \leq i} p_j]$ with length p_i for all i in N and define

$$\mu(t) \ := \ \begin{cases} \frac{\alpha_i}{p_i} & \text{if } t \in [p_i] \\ 0 & \text{if } t \geq \sum_{j \in N} p_j. \end{cases}$$

Observe that $\mu : \mathbb{R}_+ \longrightarrow \mathbb{R}_+$ is weakly decreasing.

Define the N-reallocation $\{(B_i, z_i)\}_{i\in N}$, by

$$\begin{aligned} B_i &:= [0,\beta(N)] \times [p_i], \\ z_i &:= -\tfrac{p_i}{\beta(N)}\cdot[\tfrac{1}{2}\cdot\alpha_i + \alpha_{i+1} + \ldots + \alpha_n] + \tfrac{\beta_i}{\beta(N)}\sum_{j\in N}\tfrac{p_j}{\beta(N)}\cdot[\tfrac{1}{2}\cdot\alpha_j + \alpha_{j+1} + \ldots + \alpha_n]. \end{aligned}$$

Furthermore, we define the price density function $\pi : L \longrightarrow \mathbb{R}$ by

$$\pi(x,t) := \tfrac{1}{\beta(N)^2}\int_t^\infty \mu(\tau)\mathrm{d}\tau.$$

We prove that the N-reallocation $\{(B_i, z_i)\}_{i\in N}$ is a price equilibrium supported by this price density function.

For all i in N we have

$$\begin{aligned} P_\pi(B_i) &= \int_0^\infty\int_0^{\beta(N)} 1_{B_i}(x,t)\cdot\pi(x,t)\,\mathrm{d}x\mathrm{d}t \\ &= \int_{[p_i]}\int_0^{\beta(N)} \pi(x,t)\,\mathrm{d}x\mathrm{d}t \\ &= \tfrac{1}{\beta(N)}\int_{[p_i]}\int_t^\infty \mu(\tau)\,\mathrm{d}\tau\mathrm{d}t \\ &= \tfrac{1}{\beta(N)}\int_{[p_i]} \tfrac{\alpha_i}{p_i}\cdot[\sum_{j\le i} p_j - t] + \alpha_{i+1} + \ldots + \alpha_n\mathrm{d}t \\ &= \tfrac{1}{\beta(N)}\Big(\alpha_i\sum_{j\le i} p_j - \tfrac{1}{2}\tfrac{\alpha_i}{p_i}\cdot[(\sum_{j\le i} p_j)^2 - (\sum_{j<i} p_j)^2] + p_i\cdot[\alpha_{i+1} + \ldots + \alpha_n]\Big) \\ &= \tfrac{\alpha_i}{\beta(N)}\sum_{j\le i} p_j - \tfrac{1}{2}\tfrac{\alpha_i}{\beta(N)\cdot p_i}\cdot[p_i^2 + 2\cdot p_i\sum_{j<i} p_j] + \tfrac{p_i}{\beta(N)}\cdot[\alpha_{i+1} + \ldots + \alpha_n] \\ &= \tfrac{p_i}{\beta(N)}\cdot[\tfrac{1}{2}\cdot\alpha_i + \alpha_{i+1} + \ldots + \alpha_n]. \end{aligned}$$

Similarly, we have for all i in N

$$\begin{aligned} P_\pi(A_i) &= \int_0^\infty\int_0^{\beta(N)} 1_{A_i}(x,t)\cdot\pi(x,t)\,\mathrm{d}x\mathrm{d}t \\ &= \int_0^\infty\int_{[\beta_i]} \pi(x,t)\,\mathrm{d}x\mathrm{d}t \\ &= \int_0^\infty \tfrac{\beta_i}{\beta(N)^2}\int_t^\infty \mu(\tau)\,\mathrm{d}\tau\mathrm{d}t \\ &= \tfrac{\beta_i}{\beta(N)}\sum_{j\in N}\tfrac{1}{\beta(N)}\int_{[p_j]}\int_t^\infty \mu(\tau)\,\mathrm{d}\tau\mathrm{d}t \\ &= \tfrac{\beta_i}{\beta(N)}\cdot\sum_{j\in N}\tfrac{p_j}{\beta(N)}\cdot[\tfrac{1}{2}\cdot\alpha_j + \alpha_{j+1} + \ldots + \alpha_n]. \end{aligned}$$

This yields, for all i in N,

$$P_\pi(B_i) + z_i \;=\; P_\pi(A_i).$$

Hence, the budget constrains are satisfied. Now we prove that the maximality conditions also hold. To obtain a contradiction, suppose there exists $C \subseteq L$, $y \in \mathbb{R}$ and $i \in N$ such that

$$V_i(C) + y \;>\; V_i(B_i) + z_i \;\text{ and }\; P_\pi(C) + y \;\leq\; P_\pi(A_i).$$

Because $P_\pi(A_i) = P_\pi(B_i) + z_i$, these two inequalities yield

$$V_i(C) - P_\pi(C) \;>\; V_i(B_i) - P_\pi(B_i). \tag{5}$$

Since the price density function $\pi(x,t)$ does not depend on x and is decreasing in t we can assume without loss of generality that $C := C_t = [0,\, \beta(N)] \times [t, t+p_i]$ for a certain number t. Define the function $f : \mathbb{R}_+ \longrightarrow \mathbb{R}$ by

$$f(t) \;:=\; V_i(C_t) - P_\pi(C_t) \;=\; -\tfrac{\alpha_i}{\beta(N)}\cdot(t+p_i) - \tfrac{1}{\beta(N)} \int_t^{t+p_i}\int_\tau^\infty \mu(\zeta)\mathrm{d}\zeta\mathrm{d}\tau.$$

Observe that f is differentiable on $\mathbb{R}_+$ and

$$\begin{aligned} f'(t) \;&=\; -\tfrac{\alpha_i}{\beta(N)} - \tfrac{1}{\beta(N)}\Big[\int_{t+p_i}^\infty \mu(\zeta)\mathrm{d}\zeta \;-\; \int_t^\infty \mu(\zeta)\mathrm{d}\zeta\Big] \\ &=\; -\tfrac{\alpha_i}{\beta(N)} + \tfrac{1}{\beta(N)} \int_t^{t+p_i} \mu(\zeta)\mathrm{d}\zeta. \end{aligned}$$

Hence, f' is also differentiable on $\mathbb{R}_+$ and $f''(t) \;=\; \frac{1}{\beta(N)}[\mu(t+p_i) - \mu(t)] \;\leq\; 0$. This inequality follows from the fact μ is weakly decreasing. So, f is a concave function. Therefore its maximal value is taken in t whenever $f'(t) = 0$. Hence, f takes its maximal value in the point $t := \sum_{j<i} p_j$. Therefore,

$$f(t) \;\leq\; f\big(\sum_{j<i} p_j\big) \;=\; V_i(B_i) - P_\pi(B_i) \;\text{ for all } t \in \mathbb{R}_+.$$

This contradicts equation (5) and as a result it follows that the N-reallocation $\{(B_i, z_i)\}_{i\in N}$ is a price equilibrium supported by the price density function π. $\square$

The proof of Theorem 3.1 is now straightforward.

Proof of Theorem 3.1. Because the exchange economy with land $\mathcal{E}(\mathcal{P})$ has a price equilibrium, the TU-game $\langle N, -c^{\mathcal{P}}\rangle$ has according to Lemma 4.1 and Lemma 4.2 a non-empty core. This means there exists a vector x in $\mathbb{R}^N$ such that $x(N) = -c^{\mathcal{P}}(N)$ and $x(S) \geq -c^{\mathcal{P}}(S)$ for all $S \subseteq N$. Equivalently, there exists a vector x in $\mathbb{R}^N$ such that $x(N) = c^{\mathcal{P}}(N)$ and $x(S) \leq c^{\mathcal{P}}(S)$ for all $S \subseteq N$. Hence, the cost-game $\langle N,\, c^{\mathcal{P}}\rangle$ is balanced.

The reason for the cost-game $\langle N, c^{\mathcal{P}} \rangle$ to be totally balanced, is the fact that the processing game restricted to a coalition $S \subset N$ is again a processing game and thus balanced.

According to Lemma 4.2, we also obtain for the vector $-(V_i(B_i) + z_i)_{i \in N}$, in which B_i and z_i are defined for all i in N as in the proof of Proposition 4.3, to be a core allocation in the cost-game $\langle N, c^{\mathcal{P}} \rangle$. Verifying this expression provides the core allocation stated below Theorem 3.1. $\square$

References

Borm P.E.M, Hamers, H. and Hendrickx, R.L.P (2001) Operations Research Games: A Survey, *TOP* 9, 139–216.

Curiel I., Hamers, H. and Klijn F. (2002) Sequencing games: a survey, Chapters in Game Theory, in honour of Stef Tijs (eds. Borm P. and Peters H.) Kluwer Academic Publishers, The Netherlands, 27–50.

Curiel I, Pederzoli G. and Tijs S.H. (1989) Sequencing Games, *European Journal of Operations Research* 40, 344-351.

Debreu G. (1959) Theory of Value, New York: John Wiley & Son Inc.

Klijn F., Tijs S.H. and Hamers H. (2000) A Simple Proof of the Total Balancedness of Permutation Games, *Economics Letters* 69, 323-326.

Legut J., Potters J.A.M. and Tijs S.H. (1994) Economies with Land–A Game Theoretical Approach, *Games and Economic Behavior* 6, 416–430.

Quant M., Meertens M. and Reijnierse J.H. (2004), Processing games with shared interest, MIMEO.

Smith W. (1956) Various optimizers for single-stage production, *Naval Research Logistics Quarterly* 3, 59–66.