Working Paper in Economics

# 202221

# Designing Rotation Programs: Limits and Possibilities

Ville Korpela
Michele Lombardi
Riccardo Saulle

# Designing Rotation Programs: Limits and Possibilities*

Ville Korpela† Michele Lombardi‡ Riccardo D. Saulle§

July 28, 2022

**Abstract**

Rotation programs are widely used in our society. For instance, a job rotation program is an HR strategy where employees rotate between two or more jobs in the same business. We study rotation programs within the standard implementation framework under complete information. When the designer would like to attain a Pareto efficient goal, we provide sufficient conditions for its implementation in a rotation program. However, when, for instance, every employee transitions through all different lateral jobs before rotating back to his original one, the conditions fully characterize the class of Pareto efficient goals that are implementable in rotation programs.

**Keywords:** Rotation Programs; Job Rotation; Assignment Problems; Implementation; Rights Structures; Stability.

**JEL Codes:** C71; D71; D82.

*Special thanks go to Yves Sprumont and Acelya Altuntas, whose comments and suggestions have led to substantial improvements in the manuscript. We also thank audiences at the $20^{th}$ Annual SAET Conference, the International Conference on Social Choice and Voting Theory 2021, the European Meeting on Game Theory (SING16), and the research seminar at Deakin University. All opinions and errors are ours alone. Riccardo Saulle aknowledge financial support of by the University of Padova under STARS Grant

†Turku School of Economics, University of Turku. E-mail: vipeko@utu.fi.

‡University of Liverpool Management School, Liverpool, UK. Email: michele.lombardi@liverpool.ac.uk.

§Department of Economics and Management, University of Padova. Email: riccardo.saulle@unipd.it

# Contents

# 1 Introduction

An economics department must distribute the administrative load among its professors. However, most professors want to avoid these tasks due to the workload. Often, departments agree to implement a rotation program to resolve this impasse. Each professor will perform new administrative duties before returning to his original tasks.

As mentioned in the abstract, society uses rotation programs widely. The business practice of job rotations is a prominent example, which consists of periodically rotating the jobs assigned to the employees throughout their employment. This practice has been used in many industries for a wide array of employees, from factory line workers to executives (Osterman, 1994, 2000; Gittleman, Horrigan and Joyce, 1998) and for different reasons.[1] Furthermore, rotation programs have been practiced in managing common-pool resources as an alternative to quotas and lotteries. In many areas of the world, rotating groups are formed for farming, grazing, gaining access to water, and allocating fishing spots (Ostrom, 1990; Berkes, 1992; Sneath, 1998). Recently, Ely, Galeotti and Jakub (2021) show that rotation schemes can be used to prevent the spread of infections. In this view, a rotation scheme is a mechanism to shape social interactions to minimize the risk of contagion. Further, as illustrated by the problem of allocating admin duties to professors, rotation programs can help achieving fairness in assignment problems. Indeed, human beings tend to solve these conflicts by using lotteries or rotation schemes. Although the literature on assignment problems focuses mainly on randomization (Hofstee, 1990; Bogomolnaia and Moulin , 2001; Budish, Che, Kojima and Milgrom, 2013), experimental evidence (Eliaz and Rubinstein, 2014; Andreoni, Aydin, Barton, Bernheim, and Naecker, 2020) indicates that lotteries do not avoid ex-post envy.

In this paper, we propose an implementation approach to studying rotation

[1] From one side, employees who rotate accumulate more human capital because they gain a broader range of experiences. On the other side, the employer learns more about its employees if it can observe how they perform at different jobs (Arya and Mittendorf, 2004).

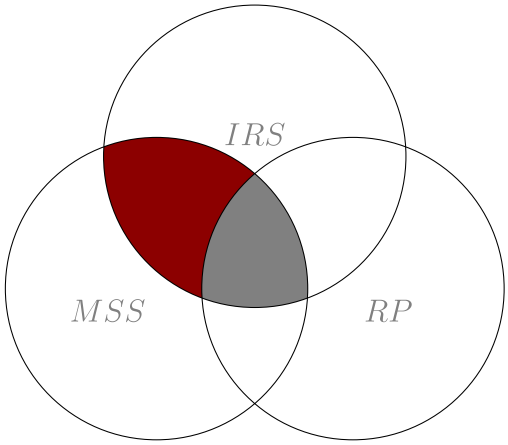
programs where agents rotate among (Pareto) efficient allocations. Therefore, our challenge lies in designing a mechanism (i.e., game form) in which agents' behavior always coincides with the recommendation given by a social choice rule (SCR). If such a mechanism exists, the SCR is implementable in rotation programs.

The first difficulty in adopting this approach concerns the choice of the solution concept. Most of the game-theoretical solutions used in literature, such as the core, the (strong) Nash equilibrium, and the stable set (von Neumann and Morgenstern, 1944), satisfy a property called *internal stability*. Roughly speaking, a set of outcomes is internally stable if it is free of inner contradictions, i.e., for every outcome in the set, no agent or group can directly move to another outcome of the set and be better off. However, this property is incompatible with our objective to study how to rotate positions among agents. Thus, a theory of implementation in rotation programs cannot rely on solutions that satisfy internal stability. Instead, internal stability is relaxed in solution concepts that are modifications, extensions, or generalizations of the stable set. One of the most prominent is the "absorbing set." As Inarra, Kuipers and Oilazola (2005) point out, the notion of absorbing sets appears in the literature under different names and settings. Kalai, Pazner, and Schmeidler (1976) study the "admissible set" in various bargaining situations, and Shenoy (1979) defines the "elementary dynamic solution" for coalitional games. More recently, Jackson and Watts (2002) study the "closed cycle" for network formation and Inarra, Larrea and Molis (2013) study the absorbing set for roommate problems. Finally, the myopic stable set (MSS), defined by Demuynck, Herings, Saulle and Seel (2019a) for a general class of games, includes all previous notions of absorbing sets. The MSS is the smallest set of states such that the following properties are satisfied: 1) There are no profitable deviations from a state **inside** the set to a state **outside** the set, and 2) for each state outside the set, there is a sequence of agents' deviations converging to the set. Thus, the MSS is a valid prediction of agents' play, though it violates internal stability because it allows deviations within the set.

Furthermore, the prediction offered by the MSS is robust in the following terms: Though agents may reach an agreement on a state outside the set, a sequence of myopic improvements will bring them back to the MSS. For these reasons, we adopt the MSS as our solution concept.

From a methodological point of view, we exploit a novel implementation technique, named implementation via *rights structures* (Section 2), a notion introduced by Sertel (2001) and recently developed by Koray and Yildiz (2018). A rights structure formalizes power distribution within society. Thus, in contrast to the classical mechanism design exercise, our design exercise consists of allocating rights to agents such that their behavior always coincides with the recommendation given by an SCR. We follow this approach for three reasons. Firstly, a persistent critique in economic design is that canonical mechanisms for implementing socially desirable outcomes have unnatural features (Jackson, 1992). Typically, canonical mechanisms are complex and challenging to explain in natural terms since they rely on tail-chasing constructions. By contrast, agents can easily understand the meaning of a rights structure. Secondly, though rights structures do not model time, they effectively describe all the paths generated by agents' interactions. Finally, rights structures suit very well the environment of the MSS. Indeed, a rights structure together with a preference profile returns a social environment (Chwe, 1994), which is the natural setting of the MSS (Demuynck, Herings, Saulle and Seel, 2019a).

However, implementation in MSS cannot always guarantee the order of rotation. Indeed, it cannot exclude the possibility that a rotation gets stuck in a cycle, which rules out some agents from the process. To solve this issue, Section 4 introduces the notion of *implementation in rotation programs*. Implementation in rotation programs is a particular kind of implementation in MSS, in which every cycle generated within the MSS needs to be a rotation scheme.




## Synopsis

The paper builds upon three blocks: implementation via rights structures (IRS), myopic stable sets (MSS), and rotation programs (RP). The paper's contribution lies in investigating the implications which stem from either $[IRS \cap MSS]$ or $[IRS \cap MSS \cap RP]$. The Venn diagram above depicts our contributions.

Section 2 provides the model. Section 3 studies implementation in MSS via rights structures $[IRS \cap MSS]$. We show that a condition, which we refer to as *indirect monotonicity*, is sufficient for implementing efficient SCRs in MSS via a finite rights structure.[2] It is worth mentioning that *indirect monotonicity* is weaker than an invariance condition, now widely referred to as Maskin monotonicity, and that, for the finite case, the characterization result encompasses implementation in core and generalized stable sets (van Deemen, 1991; Page and Wooders, 2009). Moreover, for marriage problems (Knuth, 1976) and a class of exchange economies with property rights (Balbuzanov and Kotowski, 2019), we show that the set of stable outcomes is implementable in MSS. It is worth stressing that the devised implementing rights structure has well-defined convergence properties (Appendix A presents these properties). In Section 4, we study implementation in rotation programs via rights structures as a particular case of implementation in MSS $[IRS \cap MSS \cap RP]$. We identify a necessary condition, named *rotation*

[2]A finite rights structure is a rights structure where the set of states is finite.

*monotonicity*, for implementation in rotation programs of efficient SCRs. When a multi-valued SCR describes the planner's goal, *rotation monotonicity* fully characterizes the class of implementable SCRs.[3] Finally, Section 5 studies two classes of assignment problems where efficient SCRs are implementable in rotation programs. Assignment problems in which agents share the same best/worst outcome, and assignment problems in which the planner knows that two agents have the same top outcome.

The main takeaway points can be summarized as follows. First, when the set of allocations is fixed, the implementability of efficient goals through rotation programs is tricky. However, as it happens in the context of auction design (Milgrom, 2004), the design of the set of allocations is crucial for successfully implementing efficient goals. Indeed, by cleverly designing the set of allocations, many significant assignment problems become implementable in rotation programs.

Appendix B includes proofs not in the main body.

## Related Literature

To the best of our knowledge, we are the first to study the economic design of rotation programs in an implementation framework that allows agents to rotate among efficient allocations. The previous contributions that come closest to what we are doing are Yu and Zhang (2020a) and Yu and Zhang (2020b). Whereas they study the properties of one particular mechanism for task rotation, we aim to characterize the class of implementing rotation schemes.

Our contribution is also in line with Arya and Mittendorf (2004), who study job rotations within a principal-agent framework. In particular, they identify conditions under which job rotation and specialization are both optimal. However, in contrast to us, their job rotation scheme does not guarantee the rotation of employees among lateral jobs.

Finally, our paper contributes to the literature on implementation via rights

[3] See, for instance, Mukherjee, Muto, Ramaekers, and Sen (2019).

structure (Koray and Yildiz, 2018, 2019; Korpela, Lombardi and Vartiainen, 2019, 2020) and it is broadly related to the literature on assignment problems (Shapley and Shubik, 1971; Hylland and Zeckhauser, 1979; Roth and Sotomayor, 1990; Abdulkadiroğlu and Sönmez, 1998).

## 2 The Setup

We consider a finite (nonempty) set $N$ of *agents*, whose cardinality is denoted by $n$, and a finite (nonempty) set of *alternatives*, denoted by $Z$. We endow $Z$ with a metric $\hat{d}$. The power set of $N$ is denoted by $\mathcal{N}$ and $\mathcal{N}_0 \equiv \mathcal{N} - \{\varnothing\}$ is the set of all nonempty subsets of $N$. Each element $K \in \mathcal{N}_0$ is called a *coalition*. A *preference ordering* $R_i$ is a complete and transitive binary relation over $Z$. Each agent $i(\in N)$ has a preference ordering $R_i$ over $Z$. The *asymmetric* part $P_i$ of $R_i$ is defined by $xP_iy$ if and only if $xR_iy$ and not $yR_ix$, whereas the *symmetric* part $I_i$ of $R_i$ is defined by $xI_iy$ if and only if $xR_iy$ and $yR_ix$. A *preference profile* is an $n$-tuple of preference orderings $R \equiv (R_i)_{i\in N}$. For all $R$, all $x, y \in Z$ and all $K \in \mathcal{N}_0$, we write $xR_Ky$ for $xR_iy$ for all $i \in K$, write $xP_Ky$ for $xP_iy$ for all $i \in K$, and write $xI_Ky$ for $xI_iy$ for all $i \in K$. As usual, $L_i(x, R)$ denotes agent $i$'s lower contour set of $x$ at $R$. The *preference domain*, denoted by $\mathscr{R}$, consists of the set of admissible preference profiles satisfying the following property:

$$R \in \mathscr{R} \iff \text{for all } x, y \in Z: \text{ if } xI_Ny, \text{ then } x = y. \tag{1}$$

The domain of preferences underlying classical assignment problems satisfies the above property and is our main focus.

The goal of the planner is to implement an SCR $F : \mathscr{R} \longrightarrow Z$ such that $F(R) \neq \varnothing$ for all $R \in \mathscr{R}$. We refer to $x \in F(R)$ as an $F$-optimal outcome at $R$. The *range* of $F$ is the set

$$F(\mathscr{R}) \equiv \{x \in Z | x \in F(R) \text{ for some } R \in \mathscr{R}\}.$$

The *graph* of $F$ is the set

$$Gr(F) \equiv \{(x, R) | x \in F(R), R \in \mathscr{R}\}$$

We impose the following assumption on $F$:

**Definition 1** (*Efficiency*). $F : \mathscr{R} \longrightarrow Z$ is (Pareto) *efficient* if for all $R \in \mathscr{R}$ and all $z \in F(R)$, there does not exist any $x \in Z$ such that $xR_N z$ and $xP_i z$ for some $i \in N$.

In developing our framework, we find it convenient to move away from classical mechanisms or game forms. Indeed, the rights structure is our design variable. Thus, we rely on an implementation framework that models rights distribution within the society. Roughly speaking, we assume that a planner first describes the available alternatives via a set of possible states. Then, he specifies which agent or group has the right to move from a state to another. The rights distribution is such that the prediction of the solution concept returns the socially desirable alternatives for any preference profile. Formally, to implement $F$, the planner constructs a *rights structure* $\Gamma = ((S, d), h, \gamma)$, where $S$ is the *state space* equipped with a metric $d$, $h : S \to Z$ the *outcome function*, and $\gamma$ a *code of rights*, which is a (possibly empty-valued) correspondence $\gamma : S \times S \rightrightarrows \mathcal{N}$. Subsequently, a code of rights specifies, for each pair of distinct states $(s, t)$, the family of coalitions $\gamma(s, t) \subseteq \mathcal{N}$ that is entitled to move from $s$ to $t$. If $\gamma(s, t) = \emptyset$, then no coalition is entitled to move from $s$ to $t$. A rigths structure $\Gamma$ is a *finite rights structure* if the cardinality of the space $S$ of $\Gamma$ is finite. We denote the set of all possible rights structures by $\mathscr{G}$.

The rights structure $\Gamma$ presented here is an augmented version of the rights structure introduced by Koray and Yildiz (2018) which does not include the metric $d$. Our formulation would allow us to properly define the solution concept over a possibly infinite state space. From an economic design perspective, the rights structure is the planner's design variable and corresponds to a "mechanism" in the economic theory jargon. A rights structure $\Gamma$ is termed finite if the state space $S$ is a finite set.

A rights structure together with a preference profile returns a *social environment* (Chwe, 1994), a general framework to model strategic interaction among agents or groups. We assume that in every social environment, the true preference profile is common knowledge among the agents. This is the case of *complete information* among the agents.

**Definition 2** (Social Environment)**.** For all $(\Gamma, R) \in \mathscr{G} \times \mathscr{R}$, the pair $(\Gamma, R)$ is a *game* or *social environment*, in which there is complete information among the agents.

Next, a behavior model is needed to predict in what state the agents will end up with. To this end, we need to select a solution concept. Formally, a solution concept is a map $\Sigma$, defined over $\mathscr{G} \times \mathscr{R}$, such that for each social environment $(\Gamma, R) \in \mathscr{G} \times \mathscr{R}$, $\Sigma(\Gamma, R)$ is a nonempty subset of the set of states $S$ associated with $\Gamma$. Elements of $\Sigma(\Gamma, R)$ are the equilibrium states of the game $(\Gamma, R)$. We can now provide a definition of implementation by a rights structure. An SCR is implementable in a solution $\Sigma$ by a finite rights structure if the set of equilibrium outcomes induced by the game coincides with the set of socially optimal outcomes at any preference profile.

**Definition 3** (Implementation)**.** A finite rights structure $\mathit{\Gamma}$ implements $F$ in $\Sigma$ if $F(R) = h \circ \Sigma(\mathit{\Gamma}, R)$ for all $R \in \mathscr{R}$. If such a rights structure exists, $F$ is implementable in $\Sigma$ by a finite rights structure.

## 3 Towards Implementation In Rotation Programs

As outlined above, the fundamental idea of our notion of implementation in rotation programs relies on the Myopic Stable Set (MSS), introduced by Demuynck, Herings, Saulle and Seel (2019a). Thus, as a first step, this section presents the MSS and studies its implementation via rights structures.

## 3.1 Implementation In Myopic Stable Set

To define the MSS, we need the notion of a *myopic improvement path*.[4] There is a myopic improvement path from $s$ to $T$ if a sequence of coalitional deviations from $s$ to a state arbitrarily close to $T$ exists such that every coalition involved in the sequence has the power as well as the incentive to move.

**Definition 4** (*Myopic Improvement Path*)**.** Given a social environment $(\Gamma, R)$, a sequence of states $s_1, \ldots, s_m$ is called a *myopic improvement path* from $s_1$ to $T \subseteq S$ at $R$ if, for all $\epsilon > 0$, there exist $s \in T$ such that $d(s, s_m) < \epsilon$ and a collection of coalitions $K_1, \ldots, K_{m-1}$ such that, for $j = 1, \ldots, m-1$,

(i) $K_j \in \gamma(s_j, s_{j+1})$

(ii) $h(s_{j+1}) P_{K_j} h(s_j)$.

An MSS can be defined as follows:[5]

**Definition 5** (*Myopic Stable Set*)**.** For every social environment $(\Gamma, R)$, $mss(\Gamma, R) \subseteq S$ is an MSS for $(\Gamma, R)$ if it is closed and satisfies the following three conditions:

1. *Deterrence of external deviations*: For all $s \in mss(\Gamma, R)$ and all $t \in S \backslash mss(\Gamma, R)$, there is no coalition $K \in \gamma(s, t)$ such that $h(t) P_K h(s)$.

2. *Asymptotic external stability*: For all $t \in S \backslash mss(\Gamma, R)$, there exists a myopic improvement path from $t$ to $mss(\Gamma, R)$.

3. *Minimality*: There is no closed set $M' \subset mss(\Gamma, R)$ that satisfies the two conditions above.

*Deterrence of external deviations* requires that, from any state in the set, there are no coalitional deviations to states outside it. *Asymptotic external stability* requires the existence of a myopic improvement path to the set from any state

[4]If the state space is finite, then Definition 4 reduces to the following: A sequence of states $s_1, \ldots, s_m$ is called a *myopic improvement path* from $s_1$ to $T \subseteq S$ at $R$ if $s_m \in T$ and there exists a collection of coalitions $K_1, \ldots, K_{m-1}$ such that, for $j = 1, \ldots, m-1$,, (i) $K_j \in \gamma(s_j, s_{j+1})$ and (ii) $h(s_{j+1}) P_{K_j} h(s_j)$.

[5]When the set of states is finite, Condition 2 reduces to the following one: Iterated External stability: For all $t \in S \backslash M$, there exists a direct myopic improvement path from $t$ to $M$.

outside it. Finally, *Minimality* requires that the MSS is the smallest closed set satisfying deterrence of external deviations and asymptotic external stability.

Let MSS($\Gamma$, $R$)=$\{s \in S \mid s \in mss(\Gamma, R)\}$ be the union of all MSSs at $(\Gamma, R)$. Thus, according to Definition 3, an SCR is implementable in MSS by a finite rights structure if the outcomes selected by $F$ coincide with those of the MSS for each preference profile.

Our characterization result relies on the following definition.

**Definition 6** (*Chain*)**.** Given a triple $(z, R, R') \in Z \times \mathscr{R} \times \mathscr{R}$, a sequence of outcomes $z_1, \ldots, z_h$, with $z = z_1$ and $z \neq z_h$, is a *chain* if there are agents $i_1, \ldots, i_{h-1}$ (not necessarily distinct) such that:

**(A.0)** $z_{k+1} P'_{i_k} z_k$ for all $k \in \{1, \ldots, h-1\}$;

**(A.1)** $L_i(z_h, R) \nsubseteq L_i(z_h, R')$ for some $i \in N$.

Condition (A.0) states that for each outcome of the sequence, an agent prefers its successor. Condition (A.1) requires that an agent experiences a preference reversal around the last element of the chain when the profile moves from $R$ to $R'$.

Our first characterization result relies on the following invariance condition.

**Definition 7** (*Indirect Monotonicity*)**.** $F : \mathscr{R} \longrightarrow Z$ satisfies *indirect monotonicity* if for all $(z, R, R') \in Z \times \mathscr{R} \times \mathscr{R}$ the following is true: if $z \in F(R)\backslash F(R')$ and $L_i(z, R) \subseteq L_i(z, R')$ for all $i \in N$, then there exists a sequence $z_1, \ldots, z_h$ with $z = z_1$, $z \neq z_h$ and $z_i \in F(R)$ for all $i = 1, ..., h$, which is a *chain*.

Suppose that $z$ is $F$-optimal at $R$. Further, suppose that preferences change from $R$ to $R'$ so that the standing of $z$ improves for every agent. Finally, suppose that $z$ is not $F$-optimal at $R'$. Therefore, $F$ violates Maskin monotonicity.[6] *Indirect monotonicity* says that from $z$, there is a chain of $F$-optimal outcomes at $R$.

[6]Maskin monotonicity says that if an outcome $z$ is $F$-optimal at the profile $R$ and this $z$ does not strictly fall in preference of anyone when the profile changes to $R'$, then $z$ must remain a $F$-optimal outcome at $R'$

Note that Maskin monotonicity implies *indirect monotonicity* and that they are equivalent when $F$ is single-valued. Also, note that our notion of *indirect monotonicity* resembles Condition $\alpha$ of Abreu and Sen (1990). However, in contrast to Abreu and Sen (1990), we require a sequence of $F$-optimal outcomes at $R$. The following example is illustrative.

**Example 1.** Suppose that $N = \{1, 2, 3\}$, $Z = \{x, y, z\}$, and $\mathscr{R} = \{R, R'\}$. The table below displays agents' preferences.

| $R$ | | $R'$ | |
|---|---|---|---|
| 1 | 2 | 1 | 2 |
| $y$ | $x, z$ | $x$ | $y, z$ |
| $x$ | $y$ | $y$ | $x$ |
| $z$ | | $z$ | |

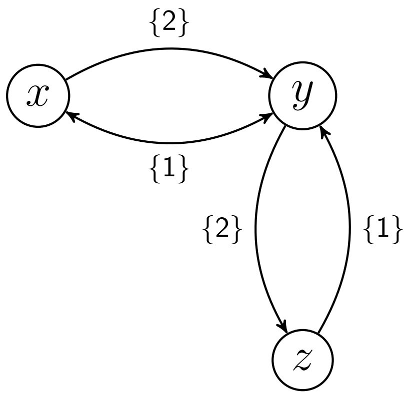



Figure 1: Implementing rights structure satisfying indirect monotonicity.

Let $F$ be such that $F(R) = \{y, z\}$ and $F(R') = \{x, y\}$. Note that $F$ selects efficient outcomes. Also, note that $F$ violates Maskin monotonicity. Indeed, the outcome $z$ is $F$-optimal at $R$, and it does not fall down in agents' preferences when they change from $R$ into $R'$, but it is not $F$-optimal at $R'$. This SCR satisfies *indirect monotonicity*.

When preferences move from $R'$ to $R$, indirect monotonicity is vacuously satisfied because $x \in F(R') \backslash F(R)$ and $L_1(x, R') = Z \subsetneq L_1(x, R) = \{z, x\}$. Let us consider the case when preferences move from $R$ to $R'$. By construction of $F$, only $z$ is such that $z \in F(R) \backslash F(R')$. Moreover, note that $L_1(z, R) = \{z\} \subseteq L_1(z, R') = \{z\}$ and $L_2(z, R) = L_2(z, R') = Z$. To satisfy condition (A.0), let the sequence of outcomes be $(z_1, z_2)$ such that $z_1 = z$ and $z_2 = y$, so that $z_1, z_2 \in F(R)$, and observe that $y P_1 z$. Thus, condition (A.0) is satisfied. Condition (A.1) is satisfied because $L_1(y, R) = Z \subsetneq L_1(y, R') = \{y, z\}$. Thus, $f$ satisfies indirect monotonicity.

Moreover, let us discuss how the rights structure depicted in Figure 1 implements $F$ in MSS. In this rights structure, states are outcomes. The rights structure is represented by an oriented graph in which vertices are the states, and the edges illustrate the code of rights. Agent 2 can move from $x$ to $y$ and from $y$ to $z$. Agent 1 can move from $z$ to $y$ and from $y$ to $x$. According to this rights structure, the unique MSS at $R$ and $R'$ are respectively $mss(\Gamma, R) = \{y, z\}$ and $mss(\Gamma, R') = \{x, y\}$. To see this, let us first consider the preference profile $R$. The set $\{y, z\}$ satisfies deterrence of external deviations because only agent 1 can deviate to $x$, but such a deviation is not profitable for him. It satisfies iterated external stability because from $x$ there is a myopic improvement path to $y$ by agent 1. It satisfies minimality because any subset of $\{y, z\}$ would violate deterrence of external stability. To this end, observe that $yP_1z$ and $zP_2y$. Finally, let us consider the preference profile $R'$. The set $\{y, x\}$ satisfies deterrence of external deviations because only agent 2 can deviate to $z$, but such a deviation is not profitable for him. It satisfies iterated external stability because from $z$ there is a myopic improvement path to $y$ by agent 1. It satisfies minimality because any subset of $\{y, x\}$ would violate deterrence of external stability. To this end, observe that $yP'_2x$ and $xP'_1y$.

The following result establishes our characterization result for the implementation in MSS via rights structures.[7]

**Theorem 1.** *Any efficient $F : \mathscr{R} \longrightarrow Z$ satisfying indirect monotonicity is implementable in MSS by a finite rights structure.*

*Indirect monotonicity* is a sufficient condition for implementation in MSS via rights structures, though it is not necessary. Example 1 in Korpela, Lombardi and Saulle (2021) makes the point. Moreover, the implementing rights structure in Example 1 consists of a rotation scheme in which society rotates between $x$ and $y$ at $R$ and between $y$ and $z$ at $R'$. However, there are circumstances in

[7]When $Z$ is not a finite set, by using the rights structure designed in the proof of Theorem 1, it is possible to show that it implements $F$ in MSS when $F$ is closed valued and upper hemicontinuous, the set of alternatives $Z$ is compact and the domain $\mathscr{R}$ is also compact.

which the implementation in MSS cannot always guarantee a rotation scheme. Before discussing this point and solving the issue by elaborating a notion of rotation programs, we discuss in the following subsections the relevance of Theorem 1. Then, however, the impatient reader can move to Section 4 without loss of understanding.

## 3.2 Convergence Property

As Jackson (1992) and Moore (1992) point out, canonical mechanisms for implementing socially desirable outcomes have unnatural futures: they are highly complex and challenging to explain in natural terms. In particular, when agents are boundedly rational, such mechanisms may lead to the convergence of undesirable outcomes. Our result shows that even unsophisticated agents, using elementary adjustment rules, can reach $F$-optimal outcomes; our mechanism is robust to some bounded rationality. Indeed, Theorem 1 demonstrates that the implementing rights structure guarantees the convergence to a myopic stable state in a finite number of transitions among states. The reason is that our implementation problems are solved by devising a finite rights structure. This property assures that the MSS can be reached in a finite sequence of myopic improvements from any state outside it.

**Corollary 1.** *Every efficient and monotonic $F : \mathscr{R} \longrightarrow Z$ is implementable in MSS via a finite rights structure.*

This result can be thought of as the counterpart of recurrent implementation in better-response dynamics studied by Cabrales and Serrano (2011), in which agents myopically adjust their actions in the direction of better-responses. When combined with a "no-worst-alternative condition," these authors show that a variant of monotonicity is essential for implementing recurrent strategies. Corollary 1 shows that for assignment problems of indivisible goods, monotonicity and Pareto efficiency are sufficient for a similar type of implementability.

We study two models where convergence is desirable in Appendix A. In particular, we consider exchange economies with complex endowment systems recently introduced by Balbuzanov and Kotowski (2019) as well as the class of "pure marriage problems" studied by Knuth (1976).[8] Both models do not satisfy any converge property. We show that the direct exclusion core of Balbuzanov and Kotowski (2019) and the solution that selects all stable matchings in the sense of (Knuth, 1976) can be implemented in MSS.

## 3.3 Connections To Other Notions Of Implementation

We conclude this section by showing that implementation in MSS by a finite rights structure is equivalent to implementation in absorbing sets and implementation in generalized stable sets (van Deemen, 1991; Page and Wooders, 2009). However, before showing it, let us formally introduce these alternative notions of stability.

**Definition 8** (Absorbing Set)**.** Let us assume that $S$ is finite. The set $A(\Gamma, R) \subseteq S$ is an absorbing set at $(\Gamma, R)$ if it satisfies the following two conditions:
***(a)*** For all $s, t \in A(\Gamma, R)$, if $s \neq t$, then there exists a finite myopic improvement path from $s$ to $\{t\}$.
***(b)*** For all $s \in A(\Gamma, R)$, there does not exist any finite myopic improvement path from $s$ to $S \backslash A(\Gamma, R)$.

Condition (a) affirms that it is possible for any state in the absorbing set to reach any other state via a myopic improvement path. Finally, by Condition (b), it is impossible to leave the absorbing set via a myopic improvement path from any state in the absorbing set.

van Deemen (1991) and Page and Wooders (2009) propose an extension of the stable set (von Neumann and Morgenstern, 1944) which replace the standard dominance relation with its transitive closure. The following is an equiva-

[8]Pure marriage problems are marriage problems in which no agent can stay single.

lent definition of the generalized stable set based on our notion of improvement path.

**Definition 9** (*Generalized Stable Set*)**.** Let us assume that $S$ is finite. The set $V(\Gamma, R) \subseteq S$ is a generalized stable set at $(\Gamma, R)$ if it satisfies the following two conditions:

**1.** *Iterated Internal Stability:* For all $s \in V(\Gamma, R)$, there is not a $t \in V(\Gamma, R)$ with $s \neq t$ such that $s = s_1, ..., s_m = t$ is a myopic improvement path from $s$ to $V(\Gamma, R)$.

**2.** *Iterated External Stability:* For all $s \in S \backslash V(\Gamma, R)$, there exists a finite myopic improvement path from $s$ to $V(\Gamma, R)$.

Inarra, Kuipers and Oilazola (2005) and Nicolas (2009) study the relation between absorbing sets and generalized stable sets. Korpela, Lombardi and Saulle (2021) (Theorem 2) provide further insights into the relationship between these solution concepts. In particular, they show that when the state space is finite, the union of generalized stable sets is equivalent to the union of absorbing sets, which, in turn, is equivalent to the unique myopic stable set.

Theorem 1, when combined with Theorem 2 in Korpela, Lombardi and Saulle (2021), gives us the following significant result.

**Corollary 2.** *Any efficient $F : \mathscr{R} \longrightarrow Z$ satisfying indirect monotonicity is implementable in absorbing sets by a finite rights structure, and in generalized stable sets by a finite rights structure.*

# 4 Rotation Programs

As noted earlier, implementation in MSS is only a preliminary step towards implementation in rotation programs. Indeed, on the one hand, implementation in MSS gives the planner the ability to design cycles among socially optimal outcomes. However, on the other hand, the planner does not have complete control of the cycles because he cannot always guarantee that agents circulate through all socially optimal outcomes. We illustrate this point through the following example.

**Example 2.** Suppose that $N = \{1, 2, 3\}$, $Z = \{x, y, z\}$, and $\mathscr{R} = \{R, R'\}$. The table below displays agents' preferences.

| $R$ | | | $R'$ | | |
|---|---|---|---|---|---|
| 1 | 2 | 3 | 1 | 2 | 3 |
| $x$ | $z$ | $y$ | $x$ | $x$ | $y$ |
| $y$ | $x$ | $z$ | $y$ | $y$ | $x$ |
| $z$ | $y$ | $x$ | $z$ | $z$ | $z$ |

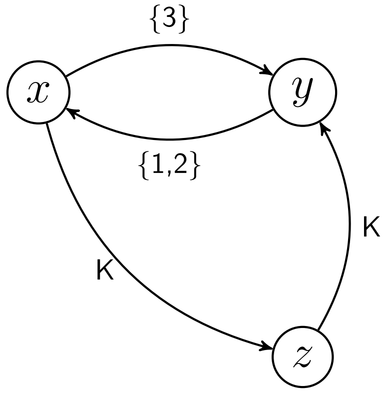



Figure 2: MSS does not imply rotation programs. $K \in \mathcal{N}_0$ such that $\#K \geqslant 2$

Let $F$ be such that $F(R) = Z$ and $F(R') = \{x, y\}$. This SCR satisfies *indirect monotonicity* vacuously because $F(R) \backslash F(R') = \{z\}$ and $L_3(z, R) = \{x, z\} \nsubseteq L_3(z, R') = \{z\}$.

Figure 2 also displays a rights structure $\Gamma$ that implements $F$ in MSS. In this rights structure, the set of states is $Z$, and the outcome function $h$ is the identity map. The codes of rights $\gamma$ is such that only agent 3 is effective in moving from $x$ to $y$, that is, $x \rightarrow^{\{3\}} y$, only the coalition 1,2 is effective in moving from $y$ to $x$, that is, $y \rightarrow^{\{1,2\}} x$, and so on.

$\Gamma$ implements $F$ in MSS because $MSS(\Gamma, R) = Z$ and $MSS(\Gamma, R') = \{x, y\}$. Since the game $(\Gamma, R)$ generates a sub-cycle in which the outcome $z$ is ruled out, it follows that devised implementing rights structure $\Gamma$ does not guarantee that agents rotate through outcomes in $Z$ when agents' preferences are $R$.

We solve this drawback by focusing on a refinement of the MSS.

## 4.1 Implementation In Rotation Programs

Let us start by defining our notion of rotation programs.

**Definition 10** (*Rotation Program*)**.** A *rotation program* for $(\Gamma, R)$ is an ordered subset of states $\bar{S} = \{s_1, ..., s_m\} \subseteq S$ such that for all $s_i, s_{i+1} \in \bar{S}$:

**(*i*)** For all $s \in \bar{S} \backslash \{s_i\}$, $h(s_i) \neq h(s)$.

**(*ii*)** For all $s \in S \backslash \{s_i, s_{i+1}\}$ and all $K \in \mathcal{N}_0$, if $K \in \gamma(s_i, s)$, then not $h(s)\, P_K h(s_i)$.

**(*iii*)** There exists $K \in \mathcal{N}_0$ such that $K \in \gamma(s_i, s_{i+1})$ and $h(s_{i+1})\, P_K h(s_i)$.

Condition (i) says that in a rotation program, no two states yield the same outcome. Conditions (ii)-(iii) imply that the only possible transitions occur among adjacent states in a uni-directional, cyclical way.

Let us now state our notion of implementation in rotation programs.

**Definition 11** (*Implementation in Rotation Programs*)**.** A rights structure $\Gamma$ implements $F : \mathscr{R} \longrightarrow Z$ in rotation programs if the following requirements are satisfied:

- $\Gamma$ implements $F$ in MSS.
- For all $R \in \mathscr{R}$, states in $MSS(\Gamma, R)$ can be used to form $m$ rotation programs $\{S_1, ..., S_m\}$ such that $h \circ S_i = F(R)$ for all $i = 1, ..., m$ and $\bigcup_{i=1}^{m} S_i = MSS(\Gamma, R)$.

If such a rights structure exists, we say that $F$ is *implementable in rotation programs*.

Conceptually, the above notion of implementation refines our notion of implementation in MSS. Indeed, it requires arranging all myopic stable states in several ordered sets. For every ordered set, the implementing rights structure induces a unique directed cycle graph among all its states, where the ordered set's direction dictates the cycle's orientation. Note that a rights structure implementing $F$ in rotation programs may have an empty core.

## 4.2 Characterization Results

In what follows, we introduce two conditions, named *Rotation Monotonicity* and *Property M*, which are at the heart of our characterization results. To this end, we need the notion of *ordered chain*.

**Definition 12** (*Ordered Chain*)**.** For all $(R, R') \in \mathscr{R} \times \mathscr{R}$ and all sets of ordered outcomes $\{z_1, ..., z_m\}$ of $Z$, a sequence $z_k, \ldots, z_{k+h}$ (modulo $m$), with $1 \leqslant k \leqslant m$ and $1 \leqslant h \leqslant m-1$, is an *ordered chain* if there are agents $i_k, \ldots, i_{k+h}$ (not necessarily distinct) and an outcome $z \in Z$ such that the following two conditions are satisfied:

**(B.0)** $z_{k+1+\ell} P'_{i_{k+\ell}} z_{k+\ell}$ for $\ell \in \{0, ..., h-1\}$;

**(B.1)** $z_{k+h} R_{i_{k+h}} z$ and $z P'_{i_{k+h}} z_{k+h}$

An ordered chain recalls the notion of a chain provided in Definition 6, though they are different. Whereas the sequence in Definition 6 does not need to follow any ordered set, the sequence in Definition 12 must satisfy restrictions imposed by the given ordered set. Moreover, both condition (B.0) and condition (A.0) require that for each outcome in the sequence, there is an agent preferring its successor. However, whereas condition (A.0) requires that also the last agent of the sequence, $i_{h-1}$, prefers $z_{h-1}$ to $z_h$, condition (B.0) does not require it. Moreover, condition (B.1) requires that the last agent of the sequence, $i_{k+h}$, has a preference reversal around the last element of the sequence, $z_{k+h}$ when preferences move from $R$ to $R'$. In contrast to (B.1), condition (A.1) is looser because any agent can have a preference reversal around the last element of the sequence $z_h$ when preferences move from $R$ to $R'$.

Rotation monotonicity, a necessary condition for implementation, can be stated as follows.

**Definition 13** (*Rotation Monotonicity*)**.** $F : \mathscr{R} \longrightarrow Z$ satisfies *rotation monotonicity* if for all $R \in \mathscr{R}$, $F(R)$ can be ordered as $z_{1,R}, \ldots, z_{m,R}$ for some integer $m \geqslant 1$, and for all $R, R' \in \mathscr{R}$, the following requirement is satisfied: if $F(R) \neq F(R')$ and either $\#F(R') > 1$ or $[\#F(R') = 1$ and $F(R') \subsetneq F(R)]$, then for all $z_{k,R} \in F(R)$ for $1 \leqslant k \leqslant m$, the sequence $z_{k,R}, \ldots, z_{k+h,R}$ (modulo $m$), with $1 \leqslant k \leqslant m$ and $1 \leqslant h \leqslant m-1$, is an *ordered chain*.

Roughly speaking, when preferences move from $R$ to $R'$ and $F(R) \neq F(R')$, rotation monotonicity requires that from every $F$-optimal outcome at $R$ starts a sequence involving only $F$-optimal outcomes at $R$ that leads to an outcome outside $F(R)$, and around which there is a preference reversal when preferences move from $R$ to $R'$.

More formally, note that *rotation monotonicity* requires that the $F$-optimal outcomes form an ordered set at every profile. Moreover, for any two profiles, $R$ and $R'$, *rotation monotonicity* applies when $F(R) \neq F(R')$ and either more than one outcome is $F$-optimal at $R'$ or the unique $F$-optimal outcome at $R'$ is not $F$-optimal at $R$. When these requirements are satisfied (not vacuously), *rotation monotonicity* states that if $z_k$ is an $F$-optimal outcome at $R$, then an ordered chain with the following two properties exists. Firstly, the sequence starts from this $z_k$. Secondly, the sequence involves only $F$-optimal outcomes at $R$.

*Rotation monotonicity* implies *indirect monotonicity* when the SCR $F$ is such that it always selects more than an outcome at each admissible profile. In contrast to *indirect monotonicity*, *rotation monotonicity* requires arranging all $F$-optimal outcomes circularly.

We now show that only SCRs satisfying *rotation monotonicity* are implementable in rotation programs.

**Theorem 2** (*Necessity*)**.** *If $F : \mathscr{R} \longrightarrow Z$ is implementable in rotation programs, then it satisfies rotation monotonicity.*

**Example 3.** The $F$ in Example 1 satisfies *rotation monotonicity*. For the convenience of the reader, Figure 3 below reproduces agents'preferences and the implementing rights structure in MSS of Example 1.

To see this, first, note that $F(R) \neq F(R')$ and both are multi-valued. We proceed according to whether $R$ moves to $R'$ or not.

- Let us consider the case where the profile $R$ moves to $R'$. At $R$, the $F$-optimal outcomes can be ordered as $z, y$. From $y \in F(R)$, the ordered chain is such that agent 1 has a preference reversal around $y$ when $R$ moves to $R'$ (i.e., $yP_1x$ and $xP_1'y$). From $z$, the ordered chain is such that $yP_1'z$ and agent 1 has a preference reversal around $y$ when $R$ moves to $R'$.

| $R$ | | $R'$ | |
|---|---|---|---|
| 1 | 2 | 1 | 2 |
| $y$ | $x, z$ | $x$ | $y, z$ |
| $x$ | $y$ | $y$ | $x$ |
| $z$ | | $z$ | |

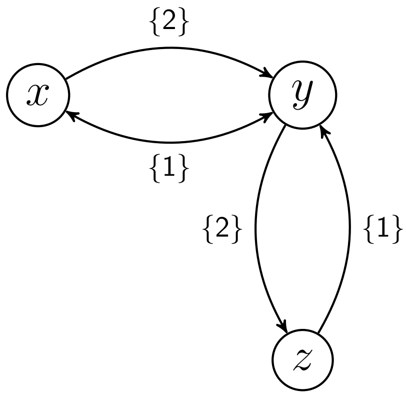



Figure 3: The implementing rights structure in Example 1 satisfies rotation monotonicity.

- Let us now consider the case where the profile $R'$ moves to $R$. At $R'$, the $F$-optimal outcomes can be ordered as $x, y$. From $y \in F(R')$, the ordered chain is such that agent 2 has a preference reversal around $y$ when $R'$ moves to $R$ (i.e., $yP'_2x$ and $xP_2y$). From $x$, the ordered chain is such that $yP_1x$ and agent 2 has a preference reversal around $y$ when $R'$ moves to $R$.

Thus, $F$ satisfies rotation monotonicity.

$F$ is also implementable in rotation programs. The implementing rights structure $\Gamma$ is that depicted in Figure 3. To see it, recall that $\Gamma$ implements $F$ in MSS, where the unique MSS at $(\Gamma, R)$ is $MSS(\Gamma, R) = \{y, z\}$, and the unique MSS at $(\Gamma, R')$ is $MSS(\Gamma, R') = \{x, y\}$. To show that $\Gamma$ implements $F$ in rotation programs, we are left to show that $MSS(\Gamma, R)$ is a rotation program for $(\Gamma, R)$ and that $MSS(\Gamma, R')$ is a rotation program for $(\Gamma, R')$. To see that $MSS(\Gamma, R) = \{y, z\}$ is a rotation program for $(\Gamma, R)$, note that parts 1-2 of Definition 10 are satisfied because $y \neq z$ and $\#MSS(\Gamma, R) = 2$. Part (iii) of Definition 10 is satisfied because $\{2\} \in \gamma(y, z)$ and $zP_2y$, and $\{1\} \in \gamma(z, y)$ and $yP_1z$. Finally, to see that $MSS(\Gamma, R') = \{y, x\}$ is a rotation program for $(\Gamma, R')$, note that parts 1-2 of Definition 10 are satisfied because $y \neq x$ and $\#MSS(\Gamma, R') = 2$. Part (iii) of Definition 10 is satisfied because $\{2\} \in \gamma(x, y)$ and $yP'_2x$, and $\{1\} \in \gamma(y, x)$ and $xP'_1y$.

**Example 4.** The $F$ in Example 2 does not satisfy *rotation monotonicity*. To see this, note that only two cyclic orderings of $F(R)$ are admissible–$x, y, z$ and $x, z, y$.

Both of them violate *rotation monotonicity*. The cycle $x, y, z$ violates *rotation monotonicity* because $L_i(y, R) \subseteq L_i(y, R')$ and $z \in L_i(y, R')$ for all $i \in N$. The cycle $x, z, y$ violates *rotation monotonicity* because $L_i(x, R) \subseteq L_i(x, R')$ and $z \in L_i(x, R')$ for all $i \in N$.

As noted earlier, *rotation monotonicity* does not bite when the unique $F$-optimal outcome at $R'$ is also $F$-optimal at $R$ and $F(R) \neq F(R')$. Therefore, *rotation monotonicity* cannot be a sufficient condition for implementing $F$ in rotation programs if no other restrictions are imposed on $F$. Indeed, *rotation monotonicity*, when combined with an auxiliary condition, termed *Property M*, is sufficient for implementation. *Property M* can be defined as follows.

**Definition 14** (*Property M*). $F : \mathscr{R} \longrightarrow Z$ satisfies *property M* if for all $R \in \mathscr{R}$, the set $F(R)$ can be ordered as $z_{1,R}, \ldots, z_{m,R}$ for $m = \#F(R)$, and for all $R, R' \in \mathscr{R}$, the following requirement is satisfied: if $F(R) \neq F(R')$, $\#F(R') = 1$ and $F(R') = z_{j,R}$ for $1 \leqslant j \leqslant m$, then for each $z_{k,R} \in F(R) \backslash F(R')$ for $1 \leqslant k \leqslant m$ and $k \neq j$,

- either the sequence $z_{k,R}, \ldots, z_{k+h,R}$ (modulo $m$) is an ordered chain;
- or there is a sequence of agents $i_1, \ldots, i_\ell$ such that:

  1. $F(R')P'_{i_\ell} z_{j-1,R} P'_{i_{\ell-1}} \ldots P'_{i_2} z_{k+1,R} P'_{i_1} z_{k,R}$

     and

  2. $L_i(z_{j,R}, R) \cup \{z_{j+1,R}\} \subseteq L_i(z_{j,R}, R') \quad \forall i \in N.$

Take any profiles $R, R'$ such that $F(R) \neq F(R')$ and such that rotation monotonicity does not bite. For instance, let $F(R') = \{z_{j,R}\} \subseteq F(R)$. *Property M* requires that for each $F$-optimal outcome at $R$ that is not $F$-optimal at $R'$, either the conclusion of *rotation monotonicity* holds, or there exists a sequence of agents who myopically prefer to move from $z_{k,R}$ to $F(R')$ at $R'$ via a sequence of $F$-optimal outcomes at $R$ and for every agent $i$, there is a monotonic change of agent $i$'s preferences around $z_{j,R}$ when the profile changes from $R$ to $R'$, and the change is such that $z_{j,R} R'_i z_{j+1,R}$.

**Theorem 3** (*Sufficiency*)**.** *If $F : \mathscr{R} \longrightarrow Z$ is efficient, and it satisfies rotation monotonicity and Property M with respect to the same set of ordered outcomes of $F(R)$, for all $R \in \mathscr{R}$, then it is implementable in rotation programs by a finite rights structure.*

Before discussing an important implication of the above characterization result, it is worth noting that the proof of Theorem 3 does not work without efficiency. The reason is that the implementing rights structure is a variant of the rights structure devised for the proof of Theorem 1.

We conclude this section by focussing on cases where rotation programs are not trivial, that is, on cases where the planner's goal is multi-valued at each admissible profile. In these cases, *Property M* does not have any bite. It follows from Theorem 3 and **??** that *rotation monotonicity* fully characterizes the class of efficient SCRs that are implementable in rotation programs.

**Corollary 3.** *Suppose that $F : \mathscr{R} \longrightarrow Z$ is efficient. Suppose $\#F(R) > 1$ for all $R \in \mathcal{R}$. $F$ is implementable in rotation programs if and only if $F$ satisfies rotation monotonicity.*

# 5 Assignment Problems

A fundamental problem in economics is how to allocate indivisible objects to agents. We refer to this problem as an assignment problem. The objective is to assign students to rooms, for instance. In what follows, we assume that there is a set of indivisible objects, which we refer to as "jobs," and the goal is to allocate jobs to agents optimally. Since the model applies to many resource allocation settings in which the objects can be public houses, school seats, course enrollments, car park spaces, chores, joint assets of a divorcing couple, or time slots in schedules, we now apply Corollary 3 to this fundamental setting.

A job rotation problem $(N, J, P)$ is a triplet where $N = \{1, ..., n\}$ is a finite set of agents with $n \geqslant 2$, $J = \{j_1, ..., j_n\}$ is a finite set of jobs, $P = (P_i)_{i \in N}$ is a profile of linear orderings such that every $P_i \subseteq J \times J$. For every job rotation problem $(N, J, P)$, every agent $i$'s preferences over $J$ at $P_i$ can be extended to an ordering over the set of all feasible allocations $\bar{J} = \{j \in J^n | j_k \neq j_l \text{ for all } k, l \in N\}$ in the following natural way:

$$j R_i j' \Leftrightarrow \text{either } j_i P_i j'_i \text{ or } j_i = j'_i, \quad \text{for all } j, j' \in \bar{J}.$$

Let $\mathscr{R}$ denote the set of all (extended) preference profiles.

Example 5 in Appendix A shows that not every efficient $F$ on $\mathscr{R}$ is implementable in rotation programs. Given this impossibility, we focus on two classes of job rotation problems that are implementable in rotation programs.

### 5.1 A Job Rotation Problem With Restricted Domain

There are situations where there is a common best/worst job among the available ones. For instance, suppose that the head of an Economics Department has to allocate a microeconomics course to each of its microeconomics teachers. Courses can be ranked, for example, according to their sizes. The best possible assignment for everyone is to be assigned to the Ph.D. course with the lowest number of students. In contrast, the common worst possible outcome for every teacher is to be assigned to the largest possible class at the undergraduate level.

We consider assignment problems where a common best job exists. Let us denote it by $j_1^*$.[9] The set of jobs $J$ is given by $\{j_1^*, j_2, ..., j_n\}$. Let $\bar{\mathscr{R}}$ be preference domain such that
$\bar{\mathscr{R}} = \{R \in \mathscr{R} | \text{for all } i \in N, \arg\max_J R_i = \{j_1^*\}\}$. With abuse of notation, we also use $\bar{\mathscr{R}}$ to denote the set of all (extended) preference profiles.

All efficient SCRs defined over $\bar{\mathscr{R}}$ are implementable in rotation programs.

**Theorem 4.** *The efficient $F : \bar{\mathscr{R}} \to \bar{J}$ is implementable in rotation programs.*

Note that, by construction, the efficient $F$ over $\bar{\mathscr{R}}$ is multi-valued. Thus, to check Theorem 4, it suffices to show that the efficient $F$ over $\bar{\mathscr{R}}$ is rotation monotonic.

[9]Since assignment problems where a common worst job exists can be treated symmetrically, we omit their analysis here.

The intuition behind this theorem is that for each $R$, elements of $F(R)$ can be arranged circularly as $x(1,R),...,x(m,R),x(1,R)$ such that no two consecutive allocations of the arrangement allocate $j_1^*$ to the same agent. Thus, an admissible ordered set required by *rotation monotonicity* is $x(1,R),...,x(m,R)$. Take any $R'$ such that $F(R) \neq F(R')$. Since $F$ is Maskin monotonic, it follows that there exists an $x(i,R) \in F(R)$ such that $x(i,R)\,R_\ell z$ and $zP'_\ell x(i,R)$ for some agent $\ell \in N$ and an allocation $z \in \bar{J}$. Since, by the way we arranged the elements of $F(R)$, it holds that for all $k \neq i$, $x(k+1,R)\,P'_j x(k,R)$ for some agent $j$, it is clear that $F$ satisfies *rotation monotonicity*.

In the context of auction design, Milgrom (2004) states that, in contrast to much of the theoretical literature, the set of outcomes is rarely fixed in practice and is itself subject to design. This observation also extends to our assignment problems.

Let us go back to our problem of allocating courses to teachers to see it. In this context, the head of the department can design syllabuses in a way that there is a common best course, in the sense that it is, for example, the less time-consuming one. Since, in many cases, the designer can design jobs to meet the requirements of Theorem 4, the set of its applications is broad.

## 5.2 A Job Rotation Problem With Partially Informed Planner

As another application, we consider assignment problems where the designer knows that two agents have the same top-choice. Specifically, we assume that the designer knows that agent 1 and agent 2 have a common top-ranked job, although he does not know it. The domain of admissible profiles of linear orderings is given by $\hat{\mathscr{R}} = \{R \in \mathscr{R} | \tau(R_1) = \tau(R_2)\}$, where $\tau(R_i)$ denote the top-ranked job of agent $i$ at $R_i$. With abuse of notation, we also use $\hat{\mathscr{R}}$ to denote the set of all (extended) preference profiles over $\bar{J}$.

We are interested in implementing a sub-solution $\phi : \hat{\mathscr{R}} \longrightarrow \mathcal{J}_0$ of the efficient solution. We construct $\phi$ at $R$ following three sequential steps.

- **Step 1:** Assign $\tau(R_1)$ either to agent 1 or to agent 2.
- **Step 2:** Assign the remaining jobs $J \setminus \{\tau(R_1)\}$ to $N \setminus \{1,2\}$ in an efficient way.
- **Step 3:** Assign the remaining job to agent 2 if agent 1 has received his top-ranked job, otherwise, assign it to agent 1.

The set $\phi(R)$ can be thought of as the set of outcomes generated by an underlying random *serial dictatorship mechanism* (Abdulkadiroğlu and Sönmez, 1998), in which the only permutations that are admissible are those in which the first agent and the last agent of the ordering are respectively either agent 1 and agent 2 or agent 2 and agent 1.

**Theorem 5.** *$\phi : \hat{\mathscr{R}} \to \bar{J}$ is implementable in rotation programs.*

## 6 Concluding Remarks

This paper studies rotation programs in an implementation framework. A rotation program is a circular arrangement of the states of an MSS (Demuynck, Herings, Saulle and Seel, 2019a).

Implementation in MSS is robust in the following sense: at any preference profile, every non-stable allocation converges to a stable allocation via a sequence of myopic deviations. Moreover, implementation in MSS encompasses implementation in absorbing sets and in generalized stable sets. We identify a sufficient condition for implementing efficient SCRs in MSS, named *indirect monotonicity*. This condition is weaker than (Maskin) monotonicity.

Regarding the implementation in rotation programs, we show that *rotation monotonicity*, when combined with an auxiliary condition, is sufficient for implementing efficient SCRs. Moreover, rotation monotonicity fully characterizes the class of efficient SCRs that can be implemented in rotation programs when the planner's goal is multi-valued at each admissible profile.

Finally, we study some welfare implications of our characterization results. We learn that implementation in rotation programs is somewhat restrictive when the set of outcomes is fixed. However, as in the context of auction design (Milgrom, 2004), the design of the outcome space plays an essential role in implementing assignment problems. Indeed, by cleverly designing the set of outcomes, many significant assignment problems become implementable in rotation programs.

# References


Abdulkadiroğlu, Atila and Tayfun Sönmez (1998), *Random serial dictatorship and the core from random endowments in house allocation problems* Econometrica 66, 689; 6, 25

Abreu, Dilip and Arunava Sen (1990), *Subgame perfet implementation: A necessary and almost sufficient condition*, Journal of Economic Theory, 50, 258-290; 11

Andreoni, James, Deniz Aydin, Blake Barton, B. Douglas Bernheim, and Jeffrey Naecker (2020), *When Fair Isn't Fair: Understanding Choice Reversals Involving Social Preferences*, Journal of Political Economy, 128, 5, 1673-1711; 1

Arya, Anil and Brian Mittendorf (2004), *Using Job Rotation to Extract Employee Information*, The Journal of Law, Economics, and Organization, 20, 2, 400–414; 1, 5

Balbuzanov, Ivan and Maciej H. Kotowski (2019), *Endowments, Exclusion, and Exchange*, Econometrica, 87,1663-1692; 4, 14, 30, 31

Berkes, Fikret (1992), *Success and failure in marine coastal fisheries of Turkey*, Making the commons work: Theory, practice, and policy, 161–182; 1

Bogomolnaia, Anna and Hervé Moulin (2001), *A New Solution to the Random Assignment Problem*, Journal of Economic Theory, 100, 2, 295-328; 1

Budish, Eric, Yeon-Koo Che, Fuhito Kojima and Paul Milgrom (2013), *Designing Random Allocation Mechanisms: Theory and Applications*, American Economic Review 103, 2, 585-623; 1

Cabrales, Antonio and Roberto Serrano (2011), *Implementation in adaptive better-response dynamics: Towards a general theory of bounded rationality in mechanisms*, Games and Economic Behavior, 73, 2, 360-374; 13

Chwe, Michael Suk-Young (1994), *Farsighted Coalitional Stability*, Journal of Economic Theory, 63, 299–325; 3, 8

Demuynck, Thomas, P. Jean-Jacques Herings, Riccardo D. Saulle and Christian Seel (2019a), *The Myopic Stable Set for Social Environments* Econometrica, 87, 111-138; 2, 3, 8, 25, 34

Demuynck, Thomas, P. Jean-Jacques Herings, Riccardo D. Saulle and Christian Seel (2019b), *Supplement to "The Myopic Stable Set for Social Environments"* Econometrica, 87, 111-138; 32

Eliaz, Kfir and Ariel Rubinstein (2014), *On the fairness of random procedures*, Economics Letters, 123(2), 168-170; 1

Jeffrey, Ely Andrea Galeotti and Jakub Steiner (2021), *Rotation as Contagion Mitigation*, Management Science; 1

Gale, David and Lloyd Shapley (1962), *College admissions and the stability of marriage*, American Mathematical Monthly 69, 9-15; 33

Gittleman Maury, Micheal Horrigan and Mary Joyce (1998), *Flexible Workplace Practices: Evidence from a Nationally Representative Survey*, ILR Review, 52, 1,99-115; 1

Hylland, Aanund and Richard Zeckhauser (1979), *The efficient allocation of individuals to positions*, J. Polit. Econ. 87, 293–314; 6

Hofstee, Willem K.B. (1990), *Allocation by lot: A conceptual and empirical analysis*,Social Sci.Inform., 29, 745-763; 1

Jackson, Matthew O. (1992), *Implementation in Undominated Strategies: A Look at Bounded Mechanisms*, The Review of Economic Studies, 59, 4, 757–775; 3, 13

Jackson, Matthew O. and Alison Watts (2002), *The evolution of social and economic networks*, Journal of Economic Theory, 106, 265–295; 2

Kalai, Ehud, Elisha A. Pazner and David Schmeidler (1976), *Collective Choice*

*Correspondences as Admissible Outcomes of Social Bargaining Processes*, Econometrica, 44, 2, 233-240; 2

Knuth, Donald E. (1976), *Marriage stables*. Montreal: Les presses de l'Universite de Montreal: Montreal. 4, 14, 33

Koray, Semih and Kemal Yildiz (2018), *Implementation via rights structures*, Journal of Economic Theory, 176, 479-502; 3, 6, 7

Koray, Semih and Kemal Yildiz (2019), *Implementation via Rights Structures with Minimal State Spaces* Journal, Review of Social, Economic and Administrative Studies, 33, 1, 1-12; 6

Korpela, Ville, Michele Lombardi and Hannu Vartiainen (2020), *Do Coalitions Matter inDesigning Institutions?*, Forthcoming in Journal of Economic Theory; 6

Korpela, Ville, Michele Lombardi and Hannu Vartiainen (2019), *Mechanism design with farsighted agents*, Forthcoming in Games and Economic Behaviour 6

Korpela, Ville, Michele Lombardi and Riccardo Saulle (2019), *An Implementation Approach to Rotation programs*, preprint; 12, 15

Inarra, E. Kuipers, J. and Olaizola, N. (2005), *Absorbing and generalized stable sets*, Social Choice and Welfare, 24,3, 433-437; 2, 15

Inarra, Elena, Conception Larrea, and Elena Molis (2013), *Absorbing sets in roommate problems*, Games and Economic Behavior, 81, 165–178; 2

Milgrom, P. (2004), *Putting Auction Theory to Work*, Cambridge University Press 5, 24, 26

Moore, John (1992), *Implementation in environments with complete information*, in "Advances in Economic Theory: Sixth World Congress" (J. J. Laffont, Ed.), Econometric Society Monograph, Cambridge University Press, Cambridge; 13

Mukherjee, Saptarshi, Nozomu Muto, Eve Ramaekers and Arunava Sen (2019), *Implementation in undominated strategies by bounded mechanisms: The Pareto correspondence and a generalization*, Journal of Economic Theory, 180, 229-243; 5

Nicolas, Houy (2009), *More on the stable, generalized stable, absorbing and admissible sets*, Social Choice and Welfare, 33, 691; 15

Osterman, Paul (1994), *How common is workplace transformation and who adopts it?*, ILR Review, 47, 2, 173–188; 1

Osterman, Paul (2000), *Work reorganization in an era of restructuring: Trends in diffusion and effects on employee welfare*, ILR Review, 53, 2, 179–196; 1

Frank H. Page and Myrna Wooders (2009), *Strategic basins of attraction, the path dominance core, and network formation games*, Games and Economic Behavior, 66, 1, 462-487; 4, 14

Roth, Alvin E. and Marilda A. Oliveira Sotomayor (1990),*Two Sided Matching: A Study in Game-Theoretic Modelingand Analysis*, Econometric Society Monographs, Cambridge University Press, Cambridge, UK; 6

Roth, Alvin E. and John H. Vande Vate (1990), *Random paths to stability in two-sided matching*. Econometrica 58, 1475-1480; 33

Ostrom, Elinor (1990), *Governing the Commons: The Evolution of Institutions for Collective Action*, Cambridge University Press 1

Sertel, M. (2001), *Designing rights: Invisible hand theorems, covering and membership* mimeo, Bogazici University; 3

Shenoy, P.P. (1979), *On coalition formation: A game theoretical approach*, Int J Game Theory, 8, 133–164; 2

Shapley, Lloyd and Martin Shubik (1971) *The assignment game I: The core*, Int J Game Theory 1, 111–130; 6

Sneath, David (1998), *Ecology - State policy and pasture degradation in inner Asia*, Science, 281, 5380, 1147-114; 1

Tamura, Akihisa (1993), *Transformation from Arbitrary Matchings to Stable Matchings* Journal of Combinatorial Theory, Series A, 62, 310-323; 34

van Deemen, A.M.A. (1991), *A note on generalized stable sets*, Social Choice and Welfare 8,255–260; 4, 14

von Neumann, John and Oskar Morgenstern (1944), *Theory of Games and Economic Behavior*. Princeton University Press, Princeton; 2, 14

Yu, Jingsheng and Jun Zhang (2020a), *A market design approach to job rotation*, Games and Economic Behavior, 120, 180–192; 5

Yu, Jingsheng and Jun Zhang (2020b), *Job rotation: core and mechanism*, preprint; 5

# Appendix A

## Convergence in Exchange Economy

Let us consider the class of exchange economies studied by Balbuzanov and Kotowski (2019) and consider the notion of *direct exclusion core*. We show, through an example, that the free exchange of goods does not necessarily converge to the direct exclusion core. However, the direct exclusion core is implementable in MSS via a finite rights structure. This result implies that irrespective of the initial allocation of objects, it is possible to converge to a direct exclusion core allocation in a finite sequence of coalitional moves.

An *economy* is a quadruplet $(N, H, P, \omega)$ where $N = \{1, ..., n\}$ is a finite non-empty set of agents, $H = \{h_1, ..., h_m\}$ is a finite set of houses that can be allocated among the agents, $P = (P_i)_{i\in N}$ is a profile of linear orderings, where each linear ordering is defined over $H \cup \{h_0\}$, and the endowment system $\omega : 2^N \longrightarrow 2^H$ is a function that specifies the houses owned by each coalition. For each coalition $K \in \mathcal{N}_0$, we write $\omega(K) = \bigcup_{T \in \mathcal{K}_0} \omega(T)$. Let us assume that the endowment system $\omega$ satisfies the following four properties: (A1) *Agency*: $\omega(\emptyset) = \emptyset$, (A2) *Monotonicity*: $K \subseteq K' \implies \omega(K) \subseteq \omega(K')$, (A3) *Exhaustivity*: $\omega(N) = H$, and (A4) *Non-contestability*: For each $h \in H$, there exists $K^h \in \mathcal{N}_0$ such that $h \in \omega(K) \iff K^h \subseteq K$.

Property A1 restricts ownership to agents or groups. Property A2 requires that a coalition has in its endowment anything that belongs to any sub-coalition. Property A3 states that the grand coalition $N$ jointly owns everything. In property A4, coalition $K^h$ is called the minimal controlling coalition of house $h$. It guarantees that each house has a set of one or more "co-owners" without opposing and mutually exclusive claims. As Balbuzanov and Kotowski (2019, Lemma 1) show, these properties are needed to assure that the direct exclusion core is nonempty.

We assume that each agent may live in at most one house, and each house $h \in H$ may accommodate at most one agent. A house may be vacant, and an agent can be homeless. We can model this latter outcome by the agent's assignment to an outside option $h_0 \notin H$, which has unlimited capacity.

An allocation $\mu : N \longrightarrow H \cup \{h_0\}$ is an assignment of agents to houses such that $\#\mu^{-1}(h) \leqslant 1$ for all $h \in H$. We write $\mu(K)$ to denote $\bigcup_{i \in K} \mu(i)$ for any $K \in \mathcal{N}_0$. Let $(N, H, R, \omega)$ be an economy. Every linear ordering $R_i$ can be extended to an ordering over the collection $\mathcal{M}$ of allocations in the following way: $\mu R_i \mu' \iff$ either $\mu(i) P_i \mu'(i)$ or $\mu(i) = \mu'(i)$, for all $\mu, \mu' \in \mathcal{M}$. With little abuse of notation, we denote both by $R_i$. Let $\mathscr{R}$ denote the class of admissible preference profiles of extended preferences.

**Definition 15.** Given an economy $(N, H, R, \omega)$, a coalition $K \in \mathcal{N}_0$ can *directly exclusion block* the allocation $\mu$ at $R$ with allocation $\sigma$ if

(a) $\sigma(i) P_i \mu(i)$ for all $i \in K$ and

(b) $\mu(j) P_j \sigma(j) \implies \mu(j) \in \omega(K)$ for all $j \in N \backslash K$.

In words, a coalition can directly exclusion block an assignment whenever each member strictly gains from an alternative, and anyone harmed by the reallocation is excluded from a house belonging to the coalition. The *direct exclusion core* is the set of allocations that cannot be directly exclusion blocked by any nonempty coalition.

**Definition 16** (*Direct Exclusion core*)**.** Given an economy $(N, H, R, \omega)$, its *direct exclusion core*, denoted by $CO(R, \omega)$, is defined by $CO(R, \omega) = \{\mu \in \mathcal{M} |$no coalition can directly exclusion block $\mu$ at $R\}$.

Thus, no coalition can gainfully destabilize a direct exclusion core allocation by invoking their collective exclusion rights. Balbuzanov and Kotowski (2019, Lemma 1) show that the direct exclusion core is never empty, and all its allocations are efficient.

Let us show that the direct exclusion core does not satisfy any external stability requirement. To this end, let us represent an allocation $\mu$ by a permutation matrix with columns indexed by elements of $N$ and rows indexed by elements of $H \cup \{h_0\}$, where $h_0$ is the last row. If for some $h \in H \cup \{h_0\}$ and some $i \in N$, entry $\mu_{hi} = 1$, then good $h$ has been assigned to agent $i$.

Let us consider an economy with three agents and three houses.[10] Each house $i \in H$ is owned by agent $i$. The table below displays agents' preferences.

$R$

| 1 | 2 | 3 |
|---|---|---|
| 2 | 3 | 1 |
| 3 | 1 | 2 |
| 1 | 2 | 3 |
| $h_0$ | $h_0$ | $h_0$ |

$$\mu = \begin{bmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

The direct exclusion core at $R$ consists of the allocation $\mu$. Let us consider the following allocations:

$$\sigma^1 = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix}, \sigma^2 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \text{ and } \sigma^3 = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}.$$

Although the direct exclusion core is not empty, the process of 'free' exchange of houses may not lead to $\mu$ because such a process may cycle. Indeed, agents may myopically cycle around $\sigma^1$, $\sigma^2$ and $\sigma^3$.

To see it, note that for each agent $i$, his endowment $\omega(i) = i$ corresponds to his third choice–his last choice is to become homeless. Therefore, given this initial situation, coalition $\{1, 2\}$ can trade so that they can achieve the allocation $\sigma^1$. At $\sigma^1$, agent 1 obtains his first best choice. Thus, coalition $\{2, 3\}$ is the only coalition that can achieve a strict improvement. The only allocation that $\{2, 3\}$ can move to is allocation $\sigma^2$, where agent 2 obtains his first best choice. At $\sigma^2$,

[10] We borrow this example from Demuynck, Herings, saulle and Seel (2019b, pp. 12-13).

only coalition $\{1, 3\}$ can achieve a strict improvement by moving to the only attainable allocation $\sigma^3$, where agent 3 obtains his first best choice. At $\sigma^3$, only coalition $\{1, 2\}$ can achieve a strict improvement by moving to the only attainable allocation $\sigma^1$. Therefore, the free exchange may lock agents in a cycle of exchanges.

A natural question from the preceding example is whether achieving the direct exclusion core through a different exchange process is possible. Corollary 4 answers this question by showing that the direct exclusion core is implementable in MSS via a finite rights structure. To formalize our answer, fix any endowment system $\omega$ satisfying the above four properties. Let us define $F_{\omega}^{CO}$ by $F_{\omega}^{CO}(R) = CO(R, \omega)$ for all $R \in \mathscr{R}$.

**Corollary 4.** *Fix any endowment system $\omega$ satisfying properties A1-A4. $F_{\omega}^{CO}$ is implementable in MSS via a finite rights structure.*

## Convergence In Matching

As a second application, we consider a two-sided, one-to-one matching model, namely the "marriage problem". A marriage problem is a market without transfers where the sides of the market are, for example, workers and firms (job matching), medical students and hospitals (matching of students to internships), students and advisors (matching of students to thesis advisors). The two-sided markets are referred to as "men" and "women," hence the name "marriage problem." An output of the model is termed a matching, which pairs each woman with at most one man and each man with at most one woman. Roughly speaking, a matching is stable when there is no blocking pair; no pair of agents is better off with each other than with their assigned partners. There are two prominent models describing the marriage problem: the Gale-Shapley model (Gale and Shapley, 1962) and the Knuth model (Knuth, 1976). The former studies stability for marriage problems in which agents can be singles. The latter is a pure matching model in which no agent can be single (thus, the number of men and women is assumed to be the same). Roth and Vande Vate (1990) show that the set of stable matchings in the Gale-Shapley model exhibits a convergence property; for any unstable matching, a myopic improvement path to a stable matching exists. On the contrary, no general convergence result exists for the Knuth model. For instance, Tamura (1993) shows that, under the usual matching rules, when there are at least four women, preferences exist such that agents cycle among unstable matchings. Our following result fills the gap. Indeed, since a stable matching in the marriage problem is monotonic and efficient, we establish, as a corollary of Theorem 1, that the set of stable matchings in the Knuth model is implementable in MSS, and so there exists a mechanism such that a converge property in the Knuth model is established.

**Corollary 5.** *The set of stable matchings in the Knuth model is implementable in MSS via a finite rights structure.*

Note that, under usual matching rules, Demuynck, Herings, Saulle and Seel (2019a) show that the MSS is a superset of the set of stable matchings. From this point of view, Corollary 5 further enlightens the relation between the MSS and the set of stable matchings. Moreover, it suggests that implementation by rights structures could represent a tool for refining the MSS whenever its prediction under canonical rules is too loose. Since this conjecture overcomes the aim of the present manuscript, we leave it for future research.

## A Non-Implementable Efficient SCR

**Example 5.** Let $F$ be the efficient SCR defined over $\mathscr{R}$. Suppose that there are three agents. Let the profiles $P, P', P''$ be defined as follows:

| $P$ | | | $P'$ | | | $P''$ | | |
|---|---|---|---|---|---|---|---|---|
| 1 | 2 | 3 | 1 | 2 | 3 | 1 | 2 | 3 |
| $j_1$ | $j_1$ | $j_2$ | $j_1$ | $j_1$ | $j_3$ | $j_1$ | $j_1$ | $j_2$ |
| $j_3$ | $j_2$ | $j_3$ | $j_3$ | $j_2$ | $j_2$ | $j_3$ | $j_3$ | $j_3$ |
| $j_2$ | $j_3$ | $j_1$ | $j_2$ | $j_3$ | $j_1$ | $j_2$ | $j_2$ | $j_1$ |

It can easily be checked that $F(R) = \{(j_3, j_1, j_2), (j_1, j_2, j_3), (j_1, j_3, j_2)\}$, $F(R') = \{(j_3, j_1, j_2), (j_1, j_2, j_3)\}$ and $F(R'') = \{(j_3, j_1, j_2), (j_1, j_3, j_2)\}$. $F$ is not implementable in rotation programs because it violates *rotation monotonicity*. To see it, assume, to the contrary, that $F$ satisfies *rotation monotonicity*. Then, the elements of $F(R)$ can be ordered as $x(1, R), x(2, R), x(3, R)$.

Let us consider $R''$. Select $i \in N$ such that $x(i, R) = (j_3, j_1, j_2)$. We show that $x(i+1, R) = (j_1, j_3, j_2)$. Since $x(i, R)$ has not fallen strictly in anyone's preference ordering because $R''$ is a monotonic transformation of $R$ at $(j_3, j_1, j_2) = x(i, R)$–$L_i((j_3, j_1, j_2), R) \subseteq L_i((j_3, j_1, j_2), R')$ for each agent $i$, it follows that we can only move to the next element of the ordered set, that is, to $x(i+1, R)$. Since the top-ranked job for agent 2 at $P''$ is $j_1$ and since, moreover, the top-ranked job for agent 3 at $P''$ is $j_2$, it follows that only agent 1 can move to $x(i+1, R)$ at $R''$, which implies that $x(i+1, R)$ must coincide with $(j_1, j_2, j_3)$, that is, we have that $x(i+1, R)\, P_1'' x(i, R)$ and $x(i+1, R) = (j_1, j_3, j_2)$.[11]

Let us now consider $R'$. Let us consider the allocation $x(i+1, R) = (j_1, j_2, j_3)$. Since $R'$ is a monotonic transformation of $R$ at $x(i+1, R)$, it follows that we can only move to the next element of the ordered set, that is, to $x(i+2, R)$. Note that the top-ranked job for agent 1 at $R'$ is $j_1$. Also, note that the top-ranked job for agent 3 at $R'$ is $j_3$. The preceding discussion implies that only agent 2 can move to $x(i+2, R)$, and so $x(i+2, R)$ must coincide with $(j_3, j_1, j_2) = x(i, R)$, which contradicts the assumption that the elements of $F(R)$ can be ordered as $x(1, R), x(2, R), x(3, R)$. Thus, $F$ does not satisfy *rotation monotonicity*.

# Appendix B

[11] It cannot be that $x(i+1, R) = (j_1, j_3, j_2)$ because this would lead to the contradiction that $x(i+2, R) = (j_3, j_1, j_2)$. The reason is that there cannot be any preference reversal around $(j_1, j_2, j_3)$ because $R''$ is a monotonic transformation of $R$ at $(j_1, j_3, j_2)$. Thus, we can only move to the next element of the ordered set. Since the top-ranked job for agent 1 at $P''$ is $j_1$ and since, moreover, the top-ranked job for agent 3 at $P''$ is $j_2$, the allocation $x(i+2, R)$ must coincide with $(j_3, j_1, j_2)$ because $(j_3, j_1, j_2)\, P_2''\, (j_1, j_3, j_2)$.

# Proofs

**Proof of Theorem 1.** The state space $S$ consists of $S = Gr(F) \cup Z$. Since $Z$ is finite, $S$ is also finite. The outcome function $h$ is defined such that $h(z, R) = z$ for all $(z, R) \in S$ and $h(z) = z$ for all $z \in Z$. The following five rules define the code of rights $\gamma$:

**RULE 1:** $\{i\} \in \gamma((z, R), (x, R))$ for all $R \in \mathscr{R}$, all $z, x \in F(R)$, and all $i \in N$,

**RULE 2:** $\{i\} \in \gamma((z, R), x)$ if $x \in L_i(z, R)$,

**RULE 3:** $\{i\} \in \gamma(x, (z, R))$ for all $x, (z, R) \in S$, and all $i \in N$,

**RULE 4:** $\{i\} \in \gamma(x, y)$ for all $x, y \in S$, and all $i \in N$, and

**RULE 5:** $\gamma(s, s') = \varnothing$ for any other $s, s' \in S$.

Let us show that the rights structure $\Gamma = (S, h, \gamma)$ defined above implements $F$ in MSS if $F$ is efficient and indirect monotonic. To this end, suppose that $F$ is efficient and indirect monotonic. The following lemmata will help to prove our result. To proceed with our lemmata, we need the following additional definitions. For each $R, R' \in \mathscr{R}$:

$$M(R) \equiv \{(z, R) \mid z \in F(R)\} \subseteq S \quad U(R) \equiv \{z \in Z \mid Z \subseteq L_i(z, R) \text{ for all } i \in N\};$$

$$Q(R, R') \equiv \left\{ (z', R') \in M(R') \;\middle|\; \begin{array}{l} \text{there does not exist any myopic improvement} \\ \text{path from } (z', R') \text{ to } M(R) \cup U(R) \text{ at } R \end{array} \right\};$$

$$Q(R) \equiv \bigcup_{R' \in \mathscr{R}} Q(R, R').$$

Since $S$ is finite, the property of the asymptotic external stability of Definition 5 is equivalent to the property of iterated external stability, defined in footnote 4.

Fix any profile $R$. The objective of the following lemmata is to show that

$$MSS(\Gamma, R) = M(R) \cup U(R) \cup Q(R) \text{ and } F(R) = h \circ (M(R) \cup U(R) \cup Q(R)).$$

**Lemma 1.** *There is a finite myopic improvement path to $M(R) \cup U(R)$ at $R$ from every state $s \in Z \backslash U(R)$.*

**Proof of Lemma 1.** Take any $s \in Z \backslash U(R)$. If $U(R) \neq \varnothing$, there is a one-step myopic improvement path from $s$ to $U(R)$, by Rule 4. Otherwise, suppose that $U(R) = \varnothing$. We divide the rest of the proof into two parts according to whether $s \notin F(R)$ or not.

**Case 1**: $s \notin F(R)$. Suppose that $sR_i h(s')$ for all $i \in N$ and all $s' \in M(R)$. Since $s' \in M(R)$ and $F$ satisfies efficiency, it holds that $sI_i h(s')$ for all $i \in N$. Since $R \in \mathscr{R}$, it follows that $s = h(s')$, and so $s \in F(R)$, which is a contradiction. Therefore, it must be the case that there exists an $s' \in M(R)$ such that $h(s')P_i s$ for some $i \in N$. Hence, by Rule 3, there exists a one-step improvement path from $s$ to $M(R)$ at $R$.

**Case 2**: $s \in F(R)$. Suppose that there exists an agent $i \in N$ such that $h(s')P_i s$ for some $s' \in M(R)$. By Rule 3, there exists a one step myopic improvement path from $s$ to $M(R)$ at $R$. Otherwise, suppose that $sR_i h(s')$ for all $s' \in M(R)$ and for all $i \in N$. Efficiency of $F$ implies that $h(s')I_N s$ for all $s' \in M(R)$, and so $h(s') = s$ because $R \in \mathscr{R}$. However, since $U(R) = \varnothing$, there exists $s'' \in Z$ and an agent $i \in N$ such that $s''P_i s$. Note that agent $i$ has the power to move from $s$ to $s'$ by Rule 4 and the incentive to do so since $s''P_i s$. Since $F$ satisfies efficiency and $s \in F(R)$, there must exist another agent $j \in N \backslash \{i\}$ such that $sP_j s''$. Since $s \in F(R)$, by assumption, it follows that $(s, R) \in M(R)$. By Rule 3, agent $j$ can move from $s''$ to $(s, R)$. Hence, we have established a two-step myopic improvement path at $R$ from $s$ to $(s, R) \in M(R)$—that is, $i \in \gamma(s, s'')$ and $s''P_i s$ and $j \in \gamma(s'', (s, R))$ and $h(s, R)P_j s''$. ■

**Lemma 2.** *For any $R' \in \mathscr{R}$, the set $Q(R, R')$ satisfies deterrence of external deviations and $h\left(Q(R, R')\right) = \{h(s) \in Z | s \in Q(R, R')\} \subseteq F(R)$.*

**Proof of Lemma 2.** Suppose that $Q(R, R') \neq \varnothing$ for some $R' \in \mathscr{R}$. Otherwise, there is nothing to be proved. Let us first prove that $h\left(Q(R, R')\right) \subseteq F(R)$. By definition, $Q(R, R') \subseteq M(R')$. Take any $(z', R') \in Q(R, R')$. Assume, to the contrary, that $h(z', R') = z' \notin F(R)$. Suppose that there exists an agent $i \in N$ such that $yP_i z'$ for some $y \in L_i(z', R')$. Then, by Rule 2, agent $i \in \gamma((z', R'), y)$ since $y \in L_i(z', R')$. An immediate contradiction is obtained if $y \in U(R)$ because there is a one-step myopic improvement from $Q(R, R')$ to $U(R)$. Suppose $y \in Z \backslash U(R)$. By Lemma 1, there is a finite myopic improvement path from $y$ to $M(R) \cup U(R)$ . Therefore, there exists a finite myopic improvement path from $(z', R')$ to $M(R) \cup U(R)$, which contradicts the definition of $Q(R, R')$. Thus, it has to be that $L_i(z', R') \subseteq L_i(z', R)$ for all $i \in N$.

Let us proceed according to whether $\{z\} = F(R')$ or not. Suppose that $\{z\} = F(R')$. Since $F$ satisfies *indirect monotonicity* and $L_i(z', R') \subseteq L_i(z', R)$ for all $i \in N$, it must be the case that $z \in F(R)$, which is a contradiction. Suppose that $\{z\} \neq F(R')$. Since $z' \in F(R') \backslash F(R)$ and since $L_i(z', R') \subseteq L_i(z', R)$ for all $i \in N$, *indirect monotonicity* implies that there exist a sequence of outcomes $\{z_1 \ldots, z_h\} \subseteq F(R')$ with $z' = z_1$ and $z \neq z_h$ a sequence of agents $i_1, \ldots, i_{h-1}$ such that (i) $z_{k+1} P_{i_k} z_k$ for all $k \in \{1, \ldots, h-1\}$ and (ii) $L_i(z_h, R') \nsubseteq L_i(z_h, R)$ for some $i \in N$.

By Rule 1, part (i) of *indirect monotonicity* implies that there exists a finite myopic improvement path from $(z', R')$ to $(z_h, R') \in M(R')$ at $R$. Part (ii) of *indirect monotonicity* implies that there exists a state $y \in L_i(z_h, R')$ such that $yP_i z_h$. By Rule 2, $\{i\} \in \gamma((z_h, R'), y)$. An immediate contradiction is obtained whenever $y \in U(R)$ because there is a finite myopic improvement path from $(z', R')$ to $U(R)$ at $R$. Suppose that $y \in Z \backslash U(R)$. Then, by Lemma 1, there exists a finite myopic improvement path from $y$ to $M(R) \cup U(R)$ at $R$. Therefore, there exists a finite myopic improvement path from $(z', R')$ to $M(R) \cup U(R)$ at $R$, which contradicts our initial supposition that $(z', R') \in Q(R, R')$. We conclude that $h(Q(R, R')) \subseteq F(R)$.

To complete the proof of Lemma 2, let us show that $Q(R, R') \subseteq M(R')$ satisfies deterrence of external deviations at $R$. The only way to get out of this set is to use either Rule 1 or Rule 2. Therefore, from any state of $Q(R, R')$, agents can only deviate to $M(R')\backslash Q(R, R')$ or $Z$. Note that if $M(R')\backslash Q(R, R') \neq \varnothing$, then there exists a myopic improvement path to $M(R) \cup U(R)$ at $R$, by the definition of $Q(R, R')$. Also, note that from any state in $Z\backslash U(R)$, there exists a finite myopic improvement path to $M(R) \cup U(R)$ at $R$, by Lemma 1. Hence, if an agent could benefit by deviating from a state $s \in Q(R, R')$ to a state outside of $Q(R, R')$ at $R$, there would exist a myopic improvement path from $s$ to $M(R) \cup U(R)$ at $R$, which would contradict the definition of $Q(R, R')$. ■

**Lemma 3.** *If $V$ is a nonempty subset of $S$ satisfying both deterrence of external deviations and iterated external stability at $(\Gamma, R)$, then $M(R) \subseteq V$.*

**Proof of Lemma 3.** Let $V$ be a nonempty subset of $S$ satisfying both deterrence of external deviations and iterated external stability at $(\Gamma, R)$. We show that $M(R) \subseteq V$. We proceed in two steps.

**Step 1**: $M(R) \cap V \neq \varnothing$. For the sake of contradiction, let $M(R) \cap V = \varnothing$. Then, by iterated external stability of $V$, there exists a sequence of states $s_1, \ldots, s_m$ with $s_1 \in M(R)$ and a collection of coalitions $K_1, \ldots, K_{m-1}$ such that, for $j = 1, \ldots, m-1$, $K_j \in \gamma(s_j, s_{j+1})$ and $h(s_{j+1}) P_{K_j} h(s_j)$. Moreover, $s_m \in V$. By definition of $\gamma$, by the fact that $s_1 \in M(R)$ and that $h(s_{j+1}) P_{K_j} h(s_j)$, we have that only Rule 1 applies, and so it has to be that $\{s_1, ..., s_m\} \subseteq M(R)$. Therefore, $s_m \in M(R) \cap V$ is a contradiction.

**Step 2:** $M(R) \subseteq V$. Take any $s \in M(R)$. Assume, to the contrary, that $s \notin V$. Since, by Step 1, $M(R) \cap V \neq \varnothing$, take any $s' \in M(R) \cap V$. Since $s, s' \in M(R)$, it must be the case that $h(s) \neq h(s')$. Suppose that for some $i \in N$, $h(s) P_i h(s')$. By Rule 1, agent $i$ can move from $s'$ to $s$, which contradicts the property of deterrence of external deviations of $V$. Therefore, it has to be that $h(s') R_N h(s)$. Since $R \in \mathscr{R}$ and $h(s) \neq h(s')$, it follows that $h(s') P_i h(s)$ for some $i \in N$. Since $F$ is efficient, it follows that $h(s) \notin F(R)$, and so $s \notin M(R)$, which is a contradiction. Since the choice of $s'$ is arbitrary and since, moreover, $s \in M(R)$, it follows that $M(R) \cap V = \varnothing$, which is a contradiction. Thus, it has to be that $M(R) \subseteq V$. ■

**Lemma 4.** *The set $M(R) \cup U(R) \cup Q(R)$ satisfies both deterrence of external deviations and iterated external stability at $(\Gamma, R)$. Moreover, $F(R) = h \circ (M(R) \cup U(R) \cup Q(R))$.*

**Proof of Lemma 4.** By definition of $\Gamma$, the set $M(R)$ satisfies deterrence of external deviations. By Lemma 2, the set $Q(R)$ satisfies deterrence of external deviations. By definition, the set $U(R)$ satisfies deterrence of external deviations. Deterrence of external deviations is therefore satisfied by $M(R) \cup U(R) \cup Q(R)$. By Lemma 1, there is a finite myopic improvement path from $Z \backslash U(R)$ to $M(R) \cup U(R)$ at $R$. For any $R' \in \mathscr{R}$, by the definition of $Q(R, R')$, there is a myopic improvement path from $M(R') \backslash Q(R, R')$ to $M(R) \cup U(R)$ at $R$. This implies that for any state outside of $M(R) \cup U(R) \cup Q(R)$ there is a myopic improvement path to $M(R) \cup U(R)$ at $R$, and so iterated external stability is satisfied by $M(R) \cup U(R) \cup Q(R)$. ■

**Lemma 5.** *If $V$ is a nonempty subset of $S$ satisfying both deterrence of external deviations and iterated external stability at $(\Gamma, R)$, then $M(R) \cup U(R) \cup Q(R) \subseteq V$.*

**Proof of Lemma 5.** By Lemma 3, we already know that $M(R) \subseteq V$. By iterated external stability of $V$, it has to be that $U(R) \subseteq V$—the reason is that no myopic improvement path can begin from a unanimously best outcome. We are left to show $Q(R) \subseteq V$. To this end, take any $R' \in \mathscr{R}$. Since $Q(R, R')$ satisfies deterrence of external deviations at $(\Gamma, R)$ by Lemma 2, it follows that $Q(R, R') \subseteq V$, otherwise, iterated external stability of $V$ is violated by the fact that $Q(R, R')$ satisfies deterrence of external deviations. Since $R'$ is arbitrary, we conclude that $Q(R) \subseteq V$. Thus, $M(R) \cup U(R) \cup Q(R) \subseteq V$. ■

**Lemma 6.** $M(R) \cup U(R) \cup Q(R) = MSS(\Gamma, R)$

**Proof of Lemma 6.** Lemma 4 implies that the set $M(R) \cup U(R) \cup Q(R)$ satisfies both deterrence of external deviations and iterated external stability at $(\Gamma, R)$. Lemma 5 implies that the set $M(R) \cup U(R) \cup Q(R)$ is the smallest nonempty set satisfying these two properties. Therefore, the unique MSS of $(\Gamma, R)$ consists of $M(R) \cup U(R) \cup Q(R)$. ■

**Lemma 7.** $F(R) = h \circ (M(R) \cup U(R) \cup Q(R))$.

**Proof of Lemma 7.** Let us show that $F(R) = h \circ M(R) \cup U(R) \cup Q(R)$. Clearly, $F(R) \subseteq h \circ M(R)$, and so $F(R) \subseteq h \circ M(R) \cup U(R) \cup Q(R)$. For the converse, Lemma 2 implies that $h \circ Q(R, R') \subseteq F(R)$ for all $R' \in \mathscr{R}$. Since $F$ is efficient, it follows that $U(R) \subseteq F(R)$. Moreover, by definition of $M(R)$, it follows that $h \circ M(R) \subseteq F(R)$. Therefore, $F(R) = h \circ M(R) \cup U(R) \cup Q(R)$. ■

**Proof of Corollary 4**. Fix any endowment system $\omega$ satisfying properties A1-A4. $F_{\omega}^{CO}$ is Pareto efficient because the direct exclusion core is efficient. In light of Corollary 1, we need only to show that $F_{\omega}^{CO}$ is monotonic. To this end, take any $\mu \in F_{\omega}^{CO}(R)$ for some $R \in \mathscr{R}$. Take any $R' \in \mathscr{R}$ such that $L_i(\mu, R) \subseteq L_i(\mu, R')$ for all $i$. Let us show that $\mu \in F_{\omega}^{CO}(R') = CO(R', \omega)$. Since $\mu \in CO(R, \omega)$, it follows that no coalition can directly exclusion block $\mu$ at $R$. That is, for all $K \in \mathcal{N}_0$ and all $\sigma \in \mathcal{M}$, $\mu(i) R_i \sigma(i)$ for some $i \in K$ or $[\mu(j) P_j \sigma(j)$ for some $j \in N \backslash K$ and $\mu(j) \notin \omega(K)]$. If $\mu(i) R_i \sigma(i)$ for some $i \in K$, it follows from the fact that $R'$ is a monotonic transformation of $R$ at $\mu$ that $\mu(i) R'_i \sigma(i)$ for some $i \in K$. If $\mu(j) P_j \sigma(j)$ for some $j \in N \backslash K$ and $\mu(j) \notin \omega(K)$, it follows from the the fact that $R'$ is a monotonic transformation of $R$ at $\mu$ and the fact that $R_j$ is a linear ordering that $\mu(j) P'_j \sigma(j)$ for some $j \in N \backslash K$ and $\mu(j) \notin \omega(K)$. We have that no coalition can directly exclusion block $\mu$ at $R'$. Thus, $F_{\omega}^{CO}$ is monotonic. ■

**Proof of Theorem 2.** Suppose that $\Gamma$ implements $F$ in rotation program. Fix any $R$. Then, the set $MSS(\Gamma, R)$ is partitioned in rotation programs $\{S_1, ..., S_m\}$ such that $h \circ S_i = F(R)$ for all $i = 1, ..., J$. Fix any rotation program $S_j = \{s_1, ..., s_m\}$ for some $m \in \mathbb{N}$. Let $x(i, R) = s_i = h(s_i)$ for all $s_i \in S_j$. Thus, $F(R)$ is an ordered set of $\#S_j = m \geqslant 1$ outcomes. Fix any $R'$ such that $F(R') \neq F(R)$. Suppose that either $\#F(R') > 1$ or $[\#F(R') = 1$ and $F(R') \notin F(R)]$. Fix any $s_i \in S_j$. We proceed according to whether $s_i \in MSS(\Gamma, R')$ or not.

**Case 1**: $s_i \in MSS(\Gamma, R')$ By the implementability of $F$, $h(s_i) \in F(R) \cap F(R')$. Since by the assumption that $F(R') \notin F(R)$ whenever $\#F(R') = 1$, it must be that $\#F(R') > 1$. Since $\Gamma$ implements $F$ in rotation program, the set $MSS(\Gamma, R')$ is partitioned in rotation programs $\{\bar{S}_1, ..., \bar{S}_m\}$ such that $h \circ \bar{S}_i = F(R')$ for all $i = 1, ..., m$. Then, there exists a unique $j$ such that $s_i \in \bar{S}_j$. Without loss of generality, let $s_i = s_1 \in \bar{S}_j$.

**Step 1:**. Since $\bar{S}_j$ is a rotation program and since $\#F(R') > 1$, it follows that there exist $s_2 \in \bar{S}_j \backslash \{s_1\}$ and a coalition $K_1$ such that $K_1 \in \gamma(s_1, s_2)$ and $h(s_2) P'_{K_1} h(s_1)$. Suppose that there exists $i_1 \in K_1$ such that $h(s_1) R_{i_1} h(s_2)$. Then, there exists $h(s_2) \in Z$ such that $h(s_2) P'_{i_1} h(s_1)$ and $h(s_1) R_{i_1} h(s_2)$, where $h(s_1) = h(s_i) = x(i, R)$. Otherwise, suppose that $h(s_2) P_{K_1} h(s_1)$. Since $S_j$ is a rotation program, it follows that $s_2 = s_{i+1} \in S_j$ and $h(s_{i+1}) = x(i+1, R)$.

The above Step 1 can be applied to $s_2 = s_{i+1} \in \bar{S}_j$ to derive a state $s_3 \in \bar{S}_j \backslash \{s_2\}$ and a coalition $K_2$ such that $K_2 \in \gamma(s_2, s_3)$ and $h(s_3) P'_{K_2} h(s_2)$ where $h(s_2) = x(i+1, R)$. Suppose that $s_3 = s_1$. Since $\bar{S}_j$ is a rotation program, it follows that $\bar{S}_j = \{s_1, s_2\}$. Since $F(R') \neq F(R)$, it follows that $s_3 = s_1 \neq s_{i+2} \in S_j$. It follows that there exists $i_2 \in K_2$ such that $h(s_1) P'_{i_2} h(s_2)$ and $h(s_2) R_{i_2} h(s_1)$. Thus, $zP'_{i_2} x(i+1, R) P'_{i_1} x(i, R)$ and $x(i+1, R) R_{i_2} z$ where $z = h(s_1) = x(i, R) \in Z$. Suppose that $s_3 \neq s_1$. Then, $s_3 \in \bar{S}_j \backslash \{s_1, s_2\}$. Suppose that there exists $i_2 \in K_2$ such that $h(s_2) R_{i_2} h(s_3)$. Thus, there exists $h(s_3) = z \in Z$ such that $h(s_3) P'_{i_2} h(s_2) P'_{i_1} h(s_1)$ and $h(s_2) R_{i_2} h(s_3)$, where $h(s_1) = h(s_i) = x(i, R)$ and $h(s_2) = h(s_{i+1}) = x(i+1, R)$. Otherwise, suppose that $h(s_3) P_{K_2} h(s_2)$. Since $S_j$ is a rotation program, it follows that $s_3 = s_{i+2} \in S_j$ and $h(s_{i+2}) = x(i+2, R)$. And, so on.

Since $\bar{S}_j \neq S_j$, after a finite number $1 \leqslant h \leqslant m$ of iterations, $s_1, s_2, ..., s_{h+1}$ states and $i_1, i_2, .., i_h$ agents can be derived such that $s_1, ..., s_h \in \bar{S}_j \cap S_j$, with $h(s_\ell) = h(s_{i+\ell-1}) = x(i+\ell-1, R)$ for all $\ell = 1, ..., h$, $s_{h+1} \in \bar{S}_j$, $h(s_{h+1}) = z \in Z$ and for all $\ell \in \{1, ..., h\}$,$h(s_{\ell+1}) P'_{i_\ell} h(s_\ell)$ and $h(s_h) R_{i_h} h(s_{h+1})$.

**Case 2**: $s_i \notin MSS(\Gamma, R')$. By iterated external stability of $MSS(\Gamma, R')$, there exists a finite myopic improvement path from $s_i$ to $t \in MSS(\Gamma, R')$; that is, there are coalitions $\{K_1, ..., K_{q-1}\}$ and states $\{s_i = t_1, t_2, ..., t_q = t\}$ such that for all $p = 1, ..., q-1$, $K_p \in \gamma(t_p, t_{p+1})$ and $h(t_{p+1}) P'_{K_p} h(t_p)$. Since $\Gamma$ implements $F$ in rotation program, the set $MSS(\Gamma, R')$ is partitioned in rotation programs $\{\bar{S}_1, ..., \bar{S}_m\}$ such that $h \circ \bar{S}_i = F(R')$ for all $i = 1, ..., m$. Then, there exists a unique $j$ such that $t_q \in \bar{S}_j$.

**Step 1:** Suppose that $t_2 \neq s_{i+1}$. Since $S_j$ is a rotation program and $s_i = t_1 \in S_j$, it follows that there exists $i_1 \in K_1$ such that $h(t_1) R_{i_1} h(t_2)$ where $h(t_1) = h(s_i) = x(i, R)$. Therefore, $h(t_2) P'_{i_1} h(t_1)$ and $h(t_1) R_{i_1} h(t_2)$, as we sought. Otherwise, suppose that $t_2 = s_{i+1} \in S_j$. If there exists $i_1 \in K_1$ such that $h(t_1) R_{i_1} h(t_2)$, then again $h(t_2) P'_{i_1} h(t_1)$ and $h(t_1) R_{i_1} h(t_2)$. Otherwise, suppose that $t_2 = s_{i+1} \in S_j$, $h(t_2) = x(i+1, R)$ and $h(t_2) P_{K_1} h(t_1)$.

The reasoning used in the above Step 1 can be applied to $t_3$ to conclude that either there exists $i_2 \in K_2$ such that $h(t_2) R_{i_2} h(t_3)$ for some $i_2 \in K_2$ or $h(t_3) P_{K_2} h(t_2)$ and $t_3 = s_{i+2} \in S_j$.

In the former case, we have that $h(t_3) P'_{i_2} h(t_2) P'_{i_1} h(t_1)$ and $h(t_2) R_{i_2} h(t_3)$, where $h(t_1) = x(i, R)$ and $h(t_2) = x(i+1, R)$. In the latter case, we have that $h(t_3) = x(i+2, R)$ and $h(t_3) P_{K_2} h(t_2)$.

Since the myopic improvement path from $s_i$ to $t \in MSS(\Gamma, R')$ is finite, after a finite number $1 \leqslant r \leqslant q-1$ of iterations, we have that $h(t_{p+1}) P'_{i_p} h(t_p)$ for all $p = 1, ..., r$, and either $[h(t_r) R_{i_r} h(t_{r+1})$ for some $i_r \in K_r]$ or $[r = q-1$, $h(t_{p+1}) P_{K_p} h(t_p)$ and $t_p = s_{i+p-1} \in S_j$ for all $p = 1, ..., r$, and $t_q \in S_j \cap \bar{S}_j]$. In the former case, we have that for all $p = 1, ..., r$, $h(t_{p+1}) P'_{i_p} h(t_p)$ and $h(t_r) R_{i_r} h(t_{r+1})$, where $h(t_p) = h(s_{i+p-1}) = x(i+p-1)$ for all $p = 1, ..., r$. In the latter case, since $t_q \in \bar{S}_j$, it follows that $t_q \in MSS(\Gamma, R')$. Case 1 above can be applied to the outcome $h(t_q) = h(s_{i+q-1}) = x(i+q-1) \in F(R)$ to complete the proof.

**Proof of Theorem 3.** The implementing rights structure is a variant of the rights structure constructed in the proof of Theorem 1. The only change concerns the definition of Rule 1. The state space is $S = Gr(F) \cup Z$. The outcome function is $h(x, R) = x$ for all $(x, R) \in Gr(F)$ and $h(x) = x$ for all $x \in Z$. The code of rights $\gamma$ is defined as follows. For all $i \in N$, all $R \in \mathscr{R}$ and all $s, t \in S$:

**RULE 1:** If $s = (x(k, R), R)$ and $t = (x(k+1, R), R)$ for some $1 \leqslant k \leqslant m$, then $\{i\} \in \gamma((x(k, R), R), (x(k+1, R), R))$, where the outcomes $x(k, R)$ are those specified by properties 1 and 2.

**RULE 2:** If $s = (z, R)$, $t = x$ and $x \in L_i(z, R)$, then $\{i\} \in \gamma((z, R), x)$.

**RULE 3:** If $s = x$ and $t = (z, R)$, then $\{i\} \in \gamma(x, (z, R))$.

**RULE 4:** If $s = z$ and $t = x$, then $\{i\} \in \gamma(s,t)$.

**RULE 5:** Otherwise, $\gamma(s,t) = \varnothing$.

Rule 1 allows agent $i$ to be effective only between two consecutive socially optimal outcomes at $R$, that is, between $(x(k,R), R)$ and $(x(k+1,R), R)$ for all $1 \leqslant k \leqslant m$. Fix any $R$. Let us show that $\Gamma$ implements $F$ in rotation programs. We first show that $F(R) = h \circ MSS(\Gamma, R)$ and then we show that $\Gamma$ partitions $MSS(\Gamma, R)$ in rotation programs such that for each rotation program $S$, it holds that $F(R) = h \circ S$. To show that $F(R) = h \circ MSS(\Gamma, R)$ and that $MSS(\Gamma, R) = M(R) \cup U(R) \cup Q(R)$, we need to show that Lemmata 1-7 still hold under the new rights structure $\Gamma$. The proofs of Lemma 2 and Lemma 3 need to be amended. As far as the proof of Lemma 3 is concerned, the arguments provided to prove Step 2 of Lemma 3 no longer hold. However, the statement of this step is still true under the new $\Gamma$. To show this, take any $s = (x(i,R), R) \in M(R) \cap V$, which exists by Step 1 of the proof of Lemma 3. We show that $M(R) \subseteq V$. Assume, to the contrary, there exists $s' = (x(i',R), R) \in M(R)$ such that $s' \notin V$. To complete the proof of Lemma 3, let us first show that $M(R)$ is a rotation program. Since $F$ is efficient and since $\mathscr{R}$ satisfies the restriction in (1), it follows that for all $1 \leqslant k \leqslant m$ and all $(x(k,R), R), (x(k+1,R), R) \in M(R)$, there exists $j \in N$ such that $x(k+1,R)\, P_j x(k,R)$. By definition of Rule 1, it follows that for each $1 \leqslant k \leqslant m$, there exists $j \in N$ such that $\{j\} \in \gamma((x(k,R), R), (x(k+1,R), R))$ and $x(k+1,R)\, P_j x(k,R)$. Moreover, by definition of $\gamma$, it follows that $M(R)$ is a rotation program because for each $(x(k,R), R)$, there do not exist any $K \in \mathcal{N}_0$ and any $s \in S$, with $s \neq (x(k,R), R)$ and $s \neq (x(k+1,R), R)$, such that $K \in \gamma((x(k,R), R), s)$ and $h(s)\, P_K x(k,R)$. Let us now complete the proof of Lemma 3. Since for each $1 \leqslant k \leqslant m$ there exists $j \in N$ such that $\{j\} \in \gamma((x(k,R), R), (x(k+1,R), R))$ and $x(k+1,R)\, P_j x(k,R)$, it follows that there exist $s_0, s_1, ..., s_{p-1}, s_p$, with $s_0 = s$ and $s_p = s'$, and $i_0, ..., i_{p-1}$ such that $i_h \in \gamma(s_h, s_{h+1})$ and $h(s_{h+1})\, P_{i_h} h(s_h)$ for all $h = 0, ..., p-1$, where $s_h \in M(R)$ for all $h = 0, 1, ..., p$. Since $s_0 \in M(R) \cap V$ and $s_p \in M(R) \backslash V$, there exists the smallest index $h^* \in \{0, ..., p-1\}$ such that $s_{h^*} \in M(R) \cap V$ and $s_{h^*+1} \in M(R) \backslash V$. Since

$i_{h^*} \in \gamma(s_{h^*}, s_{h^*+1})$ and $h(s_{h^*+1}) P_{i_{h^*}} h(s_{h^*})$, this contradicts our initial supposition that $V$ satisfies the property of deterrence of external deviations. Thus, we have that $M(R) \subseteq V$, and so Lemma 3 holds as well.

As far as the proof of Lemma 2 is concerned, it needs to be amended as follows. Fix any $R' \in \mathscr{R}$. The proof of Lemma 2 holds if $\#F(R) \neq 1$ or if $\#F(R) = 1$ and $F(R) \notin F(R')$. The reason is that in these cases *rotation monotonicity* implies *indirect monotonicity*. To complete the proof of Lemma 2, let us suppose that $\#F(R) = 1$ and $F(R) \in F(R')$. Suppose that $F(R) = \{a\} \neq F(R') = \{z(1, R'), ..., z(m, R')\}$. Without loss of generality, let $a = z(1, R')$. Suppose that *Property M* implies that for each $z(i, R') \in F(R') \setminus \{z(1, R')\}$, there exist $x \in Z$ and $i_1, ..., i_h$, with $1 \leqslant h \leqslant m$, such that:

$$z(i + \ell + 1, R') P_{\ell+1} z(i + \ell, R') \text{ for all } \ell \in \{0, ..., h-1\} \quad \text{and}$$

$$z(i + h, R') P_h x \text{ and } x R'_h z(i + h, R').$$

By definition of $\gamma$, we have that for each $z(i, R') \in F(R') \setminus \{z(1, R')\}$, there exists a finite myopic improvement path from $(z(i, R'), R')$ to $x$. Suppose that $U(R) \neq \varnothing$. Since $F$ is efficient and since, moreover, $\mathscr{R}$ satisfies the restriction in (1), it follows that $U(R) = \{z(1, R')\}$. Since by Rule 2 there exists a finite myopic improvement path from $x$ to $z(1, R')$, it follows that there exists a finite myopic improvement path from $z(i, R') \in F(R') \setminus \{z(1, R')\}$ to $M(R) \cup U(R)$. Suppose that $U(R) = \varnothing$. Since Lemma 1 implies that there exists a finite myopic improvement path from $x$ to $M(R) \cup U(R)$, we conclude that there exists a finite myopic improvement path from $z(i, R') \in F(R') \setminus \{z(1, R')\}$ to $M(R) \cup U(R)$. It follows from the definition of $Q(R, R') \subseteq M(R')$ that $Q(R, R') = \varnothing$ if there exists a finite myopic improvement path from $(z(1, R'), R')$ to $M(R) \cup U(R)$, otherwise, $Q(R, R') = \{(z(1, R'), R')\}$. In either case, we have that $h \circ Q(R, R') \subseteq F(R)$ and that $Q(R, R')$ satisfies the property of deterrence of external deviations. Note that $Q(R, R') = \{(z(1, R'), R')\}$ satisfies this property for the following two reasons: 1) Since every agent $i$ is effective in moving the state from $(z(1,R'),R')$ to $(z(2,R'),R')$, it cannot be that $z(2,R')\,P_i z(1,R')$ for some $i$, otherwise, since we have already shown that there exists a finite myopic improvement path from $(z(1,R'),R')$ to $M(R)\cup U(R)$, it follows that $Q(R,R')=\varnothing$, which is a contradiction; and 2) it cannot be that $xP_i z(1,R')$ for some $i$ and some $x\in L_i(z(1,R'),R')$, otherwise, since Rule 2 implies that $\{i\}\in\gamma((z(1,R'),R'),x)$ and $xP_i z(1,R')$ and since, moreover, Lemma 1 implies that there exists a finite myopic improvement path from $x$ to $M(R)\cup U(R)$, since we have already shown that there exists a finite myopic improvement path from $(z(1,R'),R')$ to $M(R)\cup U(R)$, it follows that $Q(R,R')=\varnothing$, which is a contradiction. Suppose that the above arguments do not hold for some $z(i,R')\in F(R')\setminus\{z(1,R')\}$. Clearly, for each $z(i,R')\in F(R')\setminus\{z(1,R')\}$ such that the above arguments hold, we have that there exists a finite myopic improvement path from $z(i,R')\in F(R')\setminus\{z(1,R')\}$ to $M(R)\cup U(R)$. *Property M* implies that $L_i(z(1,R'),R')\cup\{z(2,R')\}\subseteq L_i(z(1,R'),R)$ for all $i\in N$. For each $z(i,R')\in F(R')\setminus\{z(1,R')\}$ for which the above arguments do not hold, *Property M* implies that there exists a sequence of agents $i_1,...,i_\ell$ such that

$$z(1,R')\,P_{i_\ell}z(m,R')\,P_{i_{\ell-1}}\cdots P_{i_2}z(i+1,R')\,P_{i_1}z(i,R') \tag{2}$$

Since every agent $i$ can be effective in moving the state from $(z(1,R'),R')$ to $(z(2,R'),R')$, it follows that no agent has an incentive to do so because $z(2,R')\in L_i(z(1,R'),R)$ for all $i\in N$. Since, by Rule 1, each agent $i\in\{i_1,...,i_\ell\}$ is effective in moving between two consecutive states in $M(R')$, it follows from (2) that there exists a finite myopic improvement path from $(z(i,R'),R')$ to $(z(1,R'),R')$. We conclude that for each $z(i,R')\in F(R)\setminus\{z(1,R')\}$, there exists a finite myopic improvement path from $(z(i,R'),R')$ to either $M(R)\cup U(R)$ or to $\{(z(1,R'),R')\}$. It follows that $Q(R,R')\subseteq\{(z(1,R'),R')\}$. Again, $Q(R,R')=\varnothing$ if there exists a finite myopic improvement path from $(z(1,R'),R')$ to $M(R)\cup U(R)$, otherwise, $Q(R,R')=\{(z(1,R'),R')\}$. In either case, we have that $h\circ Q(R,R')\subseteq F(R)$ and that $Q(R,R')$ satisfies the property of deterrence of external deviations. Since the choice of $R'\in\mathscr{R}$ is arbitrary, it follows that Lemma 2 holds. Since Properties 1-2 imply that Lemmata 1-7 hold, it follows that $F(R) = h \circ MSS(\Gamma, R)$ and that $MSS(\Gamma, R) = M(R) \cup U(R) \cup Q(R)$.

To show that $\Gamma$ partitions $MSS(\Gamma, R)$ in rotation programs, we proceed according to whether $\#F(R) = 1$ or not. We have shown above that $M(R)$ is a rotation program.

**Case 1**: $\#F(R) \neq 1$. The set $U(R) = \varnothing$. To see it, suppose that there exists $x \in U(R)$. Since $F$ is efficient and since, moreover, $\mathscr{R}$ satisfies the restriction in (1), it follows that $F(R) = \{x\}$, which is a contradiction. Thus, $MSS(\Gamma, R) = M(R) \cup Q(R)$. We have shown above that $M(R)$ is a rotation program. Moreover, by its definition, it follows that $F(R) = h \circ M(R)$.

Fix any $R' \in \mathscr{R}$ such that $F(R') \neq F(R)$. We show that $Q(R, R') = \varnothing$. Fix any $z(i, R') \in F(R')$. *Rotation monotonicity* implies that there exist $x \in Z$ and a sequence of agents $i_1, ..., i_h$, with $1 \leqslant h \leqslant m$, such that:

$$z(i+\ell+1, R') P_{i_{\ell+1}} z(i+\ell, R') \text{ for all } \ell \in \{0, ..., h-1\} \quad \text{and}$$

$$z(i+h, R') R'_{i_h} x \text{ and } x P_{i_h} z(i+h, R').$$

Since, by Rule 1, for each $\ell \in \{0, ..., h-1\}$, $\{i_{\ell+1}\} \in \gamma(z(i+\ell, R'), z(i+\ell+1, R'))$ and since, moreover, by Rule 2, $\{i_h\} \in \gamma(z(i+h, R'), x)$, it follows that there exists a finite myopic improvement path from $(z(i, R'), R')$ to $x$. Since $U(R) = \varnothing$, Lemma 1 implies that there exists a finite myopic improvement path from $x$ to $M(R)$. Therefore, we have established that there exists a finite myopic improvement path from $(z(i, R'), R')$ to $M(R)$, and so $(z(i, R'), R') \notin Q(R, R')$. Since the choice of $z(i, R') \in F(R')$ is arbitrary, we have that $Q(R, R') = \varnothing$.

Fix any $R' \in \mathscr{R}$ such that $F(R') = F(R)$. Nothing has to be proved if $Q(R, R') = \varnothing$. Suppose that $Q(R, R') \neq \varnothing$. We show that $Q(R, R') = M(R')$ and that $Q(R, R')$ is a rotation program. Since $F$ is efficient and since $\mathscr{R}$ satisfies the restriction in (1), it follows that for all $(x(k, R'), R'), (x(k+1, R'), R') \in M(R')$, there exists $j \in N$ such that $x(k+1, R') P_j x(k, R')$. By definition of Rule 1, it follows that for each $1 \leqslant k \leqslant m$, there exists $j \in N$ such that $\{j\} \in$

$\gamma((x(k,R'),R'),(x(k+1,R'),R'))$ and $x(k+1,R')\,P_j x(k,R')$. If there exists a finite myopic improvement path from some $(x(i,R'),R') \in M(R')\backslash Q(R,R')$ to $M(R)\cup U(R)$, it follows that for each state in $M(R')$ there exists a finite myopic improvement path to $M(R)\cup U(R)$. This implies that $Q(R,R')=\varnothing$, which is a contradiction. Thus, $Q(R,R')=M(R')$. Since Lemma 2 implies that $Q(R,R')$ satisfies the property of deterrence of external deviations, it follows that $Q(R,R')$ is a rotation program. Since the choice of $R'\in\mathscr{R}$, with $F(R')=F(R)$, is arbitrary, it follows that $MSS(\Gamma,R)$ is the union of partitioned rotation programs because, for all $R',R''\in\mathscr{R}$ such that $F(R')=F(R'')=F(R)$, it holds that $h\circ M(R')=h\circ M(R'')$ and $M(R')\cap M(R'')=\varnothing$. Thus, $F$ is implementable in rotation programs.

**Case 2**: $\#F(R)=1$. Recall that $MSS(\Gamma,R)=M(R)\cup U(R)\cup Q(R)$. Let $F(R)=\{z(1,R)\}$. Note that $M(R)=(z(1,R),R)$. Also, note that if $U(R)\neq\varnothing$, it follows from the efficiency of $F$ and the restriction of $\mathscr{R}$ in (1) that $U(R)=\{z(1,R)\}$. Note that $M(R)$ and $U(R)$ are rotation programs such that $M(R)\cap U(R)=\varnothing$. To proof is complete if we show that for all $R'\in\mathscr{R}$, either $Q(R,R')=\varnothing$ or $Q(R,R')=\{(z(1,R),R')\}$. To this end, fix any $R'\in\mathscr{R}$. Suppose that $F(R)=\{z(1,R)\}\neq F(R')$. Let us proceed according whether $F(R)\in F(R')$ or not. Suppose that $F(R)\notin F(R')$. Fix any $z(i,R')\in F(R')$. By the same arguments provided in Case 1 above, it follows that there exists a finite myopic improvement path from $(z(i,R'),R')$ to $x$. If $U(R)\neq\varnothing$, then there exists a finite myopic improvement path from $(z(i,R'),R')$ to $z(1,R)\in U(R)$. Otherwise, if $U(R)=\varnothing$, Lemma 1 implies that there exists a finite myopic improvement path from $x$ to $M(R)$. Therefore, there exists a finite myopic improvement path from $(z(i,R'),R')$ to $M(R)\cup U(R)$, and so $(z(i,R'),R')\notin Q(R,R')$. Since the choice of $z(i,R')\in F(R')$ is arbitrary, we have that $Q(R,R')=\varnothing$. Suppose that $F(R)\in F(R')=\{z(1,R'),...,z(m,R')\}$. Without loss of generality, suppose that $z(1,R)=z(1,R')$. By arguing as we have done above in the completion of the proof of Lemma 2, we have that either $Q(R,R')=\varnothing$ or $Q(R,R')=\{(z(1,R'),R')\}$, as we sought. ■

**Proof of Theorem 4.** In light of Theorem 2, it suffices to show that $F$ satisfies properties 1 and 2. Since $\#F(R) > 1$ for all $R \in \bar{\mathscr{R}}$, it follows that *Property M* is vacuously satisfied. Therefore, let us show that $F$ satisfies *rotation monotonicity* as well. To this end, we need to introduce additional notation.

For all $R \in \bar{\mathscr{R}}$ and all $i \in N$, let $N_i(R)$ denote the set of efficient allocations at $R$ that assign $j_1^*$ to agent $i$, with $n_i(R)$ representing the number of elements in $N_i(R)$. Since $J$ is a finite set, it follows that $N_i(R)$ is a finite set. For all $R \in \bar{\mathcal{R}}$ and all $i \in N$, let $\tau_2(i, R)$ denote the second top-ranked job of agent $i$ at $R_i$. For all $x \in \bar{J}$ and all $R \in \bar{\mathscr{R}}$, let $\bar{x}(R)$ be a permutation of $x$ such that (i) the agent who obtains $j_1^*$ at $x$, let us say agent $i$, obtains his second top-ranked job $\tau_2(i, R)$ at $\bar{x}(R)$; (ii) the agent who obtains agent $i$'s second top-ranked job at $x$ obtains $j_1^*$ at $\bar{x}(R)$; whereas (iii) all other agents obtain the same job both at $x$ and at $\bar{x}(R)$. Formally, $\bar{x}_i(R) = \tau_2(i, R)$ if $x_i = j_1^*$, $\bar{x}_j(R) = j_1^*$ if $x_j = \tau_2(i, R)$, and $x_h = \bar{x}_h(R)$ for all $h \in N \backslash \{i, j\}$.

The proof that $F$ satisfies *rotation monotonicity* relies on the following lemmata.

**Lemma 8.** *For all $R \in \bar{\mathscr{R}}$ and all $i \in N$, $\sum_{j \in N \backslash \{i\}} n_j(R) \geqslant n_i(R)$.*

**Proof of Lemma 8:** The statement follows if we show that for all $R \in \bar{\mathscr{R}}$ and all $i \in N$, there exists an injective function $g_i^R$ from $N_i(R)$ to $\bigcup_{j \in N \backslash \{i\}} N_j(R)$, that is, if we show that for for all $R \in \bar{\mathscr{R}}$ and all $i \in N$, every two distinct elements of $N_i(R)$ have distinct images in $\bigcup_{j \in N \backslash \{i\}} N_j(R)$ under $g_i^R$. Let us define $g_i^R : N_i(R) \longrightarrow \bigcup_{j \in N \backslash \{i\}} N_j(R)$ by $g_i^R(x) = \bar{x}(R)$. Take any two distinct $x, y \in N_i(R)$. Then, $g_i^R(x) = \bar{x}(R)$ and $g_i^R(y) = \bar{y}(R)$. Suppose that $x_j = y_j = \tau_2(i, R)$ for some $j \in N \backslash \{i\}$. Since $x \neq y$, it follows that $x_h \neq y_h$ for some $h \in N \backslash \{i, j\}$. It follows that $\bar{x}(R) \neq \bar{y}(R)$. Suppose that $x_j = \tau_2(i, R)$ and $y_h = \tau_2(i, R)$ for some $h, j \in N \backslash \{i\}$ such that $h \neq j$. It follows that $\bar{x}(R) \neq \bar{y}(R)$. Thus, $g_i^R$ is an injective function. □

**Lemma 9.** *For all $R \in \bar{\mathscr{R}}$, elements of $F(R)$ can be ordered as $x(1, R), ..., x(m, R)$, with $m = \sum_{i \in N} n_i(R) > 1$, such that for all $k = 1, ..., m$ (mod $m$), if $x_i(k, R) = j_1^*$ for some $i \in N$, then $x_i(k+1, R) \neq j_1^*$.*

**Proof of Lemma 9:** Fix any $R \in \bar{\mathscr{R}}$. Without loss of generality, let us assume that$n_1(R) \geqslant n_2(R) \geqslant ... \geqslant n_{n-1}(R) \geqslant n_n(R)$. Let us apply the following procedure to arrange allocations of $F(R)$ in a way that the statement holds:

**Step 0:** If $n_1(R) - n_2(R) = 0$, then go to Step 1. If $n_1(R) - n_2(R) = k_0 > 0$, then take any $A \subseteq N_1(R)$ such that $\#A = k_0$. By Lemma 8, there exists $3 \leqslant h \leqslant n$ such that $\sum_{i=h}^{n} n_i(R) \geqslant k_0$ and $\sum_{i=h+1}^{n} n_i(R) < k_0$. Then, select any $B \subseteq N_h(R)$ such that $\sum_{i=h+1}^{n} n_i(R) + \#B = k_0$. List elements of the set $A$ and elements of the set $B \cup \left(\cup_{i=h+1}^{n} N_i(R)\right)$ in a way that no element of $A$ stands next to another element of set $A$. Start the list with an element of $A \subseteq N_1(R)$. By construction, no two consecutive allocations of the list allocate $j_1^*$ to the same agent.

**Step 1:** Then, $n_1(R) - k_0 - n_2(R) = 0$, with $k_0 - 0$ if $n_1(R) = n_2(R)$, and that $n_1(R) - k_0 = n_2(R) \geqslant ... \geqslant n_h(R) - \#B$, where $B = \varnothing$ and $h = n$ if $n_1(R) = n_2(R)$. Let $n_h(R) - \#B = k_1$. Construct a sequence $\{x_i\}_{i=1}^{h}$ of elements in $\bigcup_{i=1}^{h} N_i(R) \backslash (A \cup B)$ (of length equal to $h$) such that $x_i \in N_i(R)$ for all $i = 1, ..., h$. Thus, the sequence is constructed in a way that no element of $N_i(R)$ stands next to another element of $N_i(R)$, and the last element of the sequence belongs to $N_h(R)$. Since there are $k_1$ sequences of this type, list these sequences one after the other. By construction, no two consecutive allocations of this arrangement allocate $j_1^*$ to the same agent. Join this linear arrangement to the right end of the arrangement of Step 0. If $n_h(R) - \#B = n_1(R) - k_0$, then the derived linear arrangement can be transformed into a circular arrangement by joining its ends. Otherwise, move to Step 2. For each $i = 1, ..., h-1$, let $A_{1i}$ denote the set of elements of $N_i(R)$ used to construct the sequences. Thus, for each $i = 1, ..., h-1$, $\#A_{1i} = k_1$ and $N_i(R) \backslash A_{1i}$ is the set of allocations that still needs to be arranged.

**Step 2:** Then, $n_1(R) - k_0 - k_1 = n_2(R) - k_1 \geqslant ... \geqslant n_{h-1}(R) - k_1$. Let $n_{h-1}(R) - k_1 = k_2$. Construct a sequence $\{x_i\}_{i=1}^{h-1}$ of elements in

$$\bigcup_{i=1}^{h} N_i(R) \backslash \left( A \cup B \cup \left( \bigcup_{i=1}^{h-1} A_{1i} \right) \right)$$

(of length equal to $h-1$) such that $x_i \in N_i(R)$ for all $i = 1, ..., h-1$. Thus, the sequence is constructed in a way that no element of $N_i(R)$ stands next to another element of $N_i(R)$, and the last element of the sequence belongs to $N_{h-1}(R)$. Since there are $k_2$ sequences of this type, list these sequences one after the other. By construction, no two consecutive allocations of this arrangement allocate $j_1^*$ to the same agent. Join this linear arrangement to the right end of the arrangement of Step 1. If $n_{h-1}(R) - k_1 - k_2 = n_1(R) - k_0 - k_1 - k_2$, then the derived linear arrangement can be transformed into a circular arrangement by joining its ends. Otherwise, move to Step 4. For each $i = 1, ..., h-2$, Let $A_{2i}$ denote the set of elements of $N_i(R)$ used to construct the sequences. Thus, for each $i = 1, ..., h-2$, $\#A_{2i} = k_2$ and $N_i(R) \setminus (A_{1i} \cup A_{2i})$ is the set of allocations that still needs to be arranged.

⋮

**Step $\ell$:** Then, $n_1(R) - \sum_{i=0}^{\ell-1} k_i = n_2(R) - \sum_{i=1}^{\ell-1} k_i \geqslant ... \geqslant n_{h-(\ell-1)}(R) - \sum_{i=1}^{\ell-1} k_i$. Let $n_{h-(\ell-1)}(R) - \sum_{i=1}^{\ell-1} k_i = k_\ell$. Construct a sequence $\{x_i\}_{i=1}^{h-(\ell-1)}$ of elements in $\bigcup_{i=1}^{h-(\ell-1)} N_i(R) \setminus \left(A \cup B \cup \left(\bigcup_{i=1}^{h-(\ell-1)} \bigcup_{j=1}^{\ell-1} A_{ji}\right)\right)$ (of length equal to $h-(\ell-1)$) such that $x_i \in N_i(R)$ for all $i = 1, ..., h-(\ell-1)$. Thus, the sequence is constructed in a way that that no element of $N_i(R)$ stands next to another element of $N_i(R)$, and the last element of the sequence belongs to $N_{h-1}(R)$. Since there are $k_\ell$ sequences of this type, list these sequences one after the other. By construction, no two consecutive allocations of this arrangement allocate $j_1^*$ to the same agent. Join this linear arrangement to the right end of the arrangement of Step $\ell-1$. If $n_{h-(\ell-1)}(R) - \sum_{i=1}^{\ell-1} k_i = n_1(R) - \sum_{i=0}^{\ell-1} k_i$, then the derived linear arrangement can be transformed into a circular arrangement by joining its ends. Otherwise, move to Step $\ell+1$. For each $i = 1, ..., h-\ell$, Let $A_{\ell i}$ denote the set of elements of $N_i(R)$ used to construct the sequences. Thus, for each $i = 1, ..., h-\ell$, $\#A_{\ell i} = k_\ell$ and $N_i(R) \setminus \left(\bigcup_{j=1}^{\ell} A_{ji}\right)$ is the set of allocations that still needs to be arranged.

⋮

Since the set of allocations is finite, the above procedure is finite, and it produces a circular arrangement of elements of $F(R)$ such that no two consecutive allocations allocate $j_1^*$ to the same agent. □

For each $R \in \bar{\mathscr{R}}$, Lemma 9 implies that elements of $F(R)$ can be ordered as $x(1,R), ..., x(m,R)$ ,with $m = \sum_{i \in N} n_i(R) > 1$, such that for all $k = 1, ..., m$ (mod $m$), if $x_i(k,R) = j_1^*$ for some $i \in N$, then $x_i(k+1,R) \neq j_1^*$. Fix any $R' \in \bar{\mathscr{R}}$ such that $F(R) \neq F(R')$. We need to consider only the case that $\#F(R') > 1$. Suppose that for all $x(i,R) \in F(R)$, there do not exist any agent $\ell$ and any allocation $z \in \bar{J}$ such that $zP'_\ell x(i,R)$ and $x(i,R)R_\ell z$. This implies that for all $x(i,R) \in F(R)$, $L_\ell(x(i,R),R) \subseteq L_\ell(x(i,R),R')$ for all $\ell \in N$. Since $F$ is (Maskin) monotonic, it follows that $F(R) = F(R')$, which is a contradiction. Thus, for some $x(i,R) \in F(R)$, there exist an agent $\ell$ and an allocation $z \in \bar{J}$ such that $zP'_\ell x(i,R)$ and $x(i,R)R_\ell z$. Fix any of such $x(i,R) \in F(R)$. Since by construction of the set $\{x(1,R), ..., x(m,R)\}$ we have that for all $k = 1, ...m$, with $k \neq i$, it holds that $x(k+1,R)P'_j x(k,R)$ for some $j$, it follows that $x(i,R)$ can be reached via a myopic improvement path at $R'$ by any outcome in $x(k,R) \in \{x(1,R), ..., x(m,R)\} \setminus \{x(i,R)\}$. Thus, $F$ satisfies *rotation monotonicity*.

**Proof of Theorem 5.** Observe that $\#\phi(R) = 2m$, where $m$ is the number of such allocations at $R$ where all jobs except $\tau(R_1)$ are assigned to agents $N \backslash \{1,2\}$ in an efficient way (agent 2 getting the leftover). It follows that *Property M* is always satisfied by $\phi$ and Corollary 3 applies. Thus it suffices to prove that *rotation monotonicity* is satisfied.

Fix any $R \in \hat{\mathscr{R}}$ and any $x \in \phi(R)$. Let $\hat{x}$ be the allocation obtained from $x$ in which the job assigned to agent 1 under $x$ is assigned to agent 2 under $\hat{x}$, the job assigned to agent 2 under $x$ is assigned to agent 1 under $\hat{x}$, whereas all other assignments are unchanged. That is, $\hat{x}_1 = x_2$, $\hat{x}_2 = x_1$, and $\hat{x}_i = x_i$ for every agent $i \neq 1, 2$. Observe that $\hat{x} \in \phi(R)$ if and only if $x \in \phi(R)$. The following result shows that the efficient solution $\phi$ is implementable in rotation programs. This result is obtained by requiring that the ordered set

$$\phi(R) = \{x(1,R), x(2,R), ..., x(2n-1,R), x(2m,R)\}$$

satisfies the following properties for all $i \in \{1, ..., 2m\}$: (1) If $i$ is odd, then $x_1(i, R) = \tau(R_1)$. (2) If $i$ is even, then $x_2(i, R) = \tau(R_2)$. (3) If $x(i, R) = x$ and $i$ is odd, then $x(i+1, R) = \hat{x}$. $\phi(R)$ is implementable in rotation programs because we can devise a rights structure that allows agent 1 (agent 2) to be effective in moving from the outcome $x(i, R)$ to $x(i+1, R)$ provided that $i$ is even (odd). The reason is that agent 1 (agent 2) has incentive to move from $x(i, R)$ to his top-ranked outcome $x(i+1, R)$ when $i$ is odd (even). To see that *rotation monotonicity* is satisfied, fix any $R'$ such that $\phi(R) \neq \phi(R')$. This implies that at least one allocation $x(i, R) \in \phi(R)$ is Pareto dominated at $R'$, that is, there exists an allocation $z$ such that $zR'_j x(i, R)$ for each agent $j \in N$ and $zP'_j x(i, R)$ for some agent $j \in N$. We can proceed according to whether $\tau(R_1) \neq \tau(R'_1)$. Suppose that $\tau(R_1) \neq \tau(R'_1)$. This implies that $\tau(R_1) = \tau(R_2)$ has fallen strictly in agent $j = 1, 2$'s ranking when the profile moves from $R$ to $R'$. The preference reversal for both agent 1 and agent 2 guarantees that *rotation monotonicity* is satisfied for every $x(i, R) \in \phi(R)$. Suppose that $\tau(R_1) = \tau(R'_1)$. We have already observed that at $R$, it holds that $x(i+1, R) P_2 x(i, R)$ if $i$ is odd, and that $x(i+1, R) P_1 x(i, R)$ if $i$ is even. In other words, there is the following cycle among outcomes in $\phi(R)$:

$$x(1, R) P_1 x(2m, R) P_2 x(2n-1, R) \cdots x(3, R) P_1 x(2, R) P_2 x(1, R)$$

Since $\tau(R_j) = \tau\left(R'_j\right)$ for $j = 1, 2$, it follows that the above cycle also exists at $R'$. Since $\phi(R) \neq \phi(R')$, we already know that there is at least one allocation $x(i, R) \in \phi(R)$ that is Pareto dominated at $R'$. Since $x(i, R)$ is efficient at $R$, it follows that $x(i, R) \in \phi(R)$ has strictly fallen in the preference ranking of at least one agent $j \neq 1, 2$ when the profile moves from $R$ to $R'$. It follows that *rotation monotonicity* is satisfied. □