Economics Letters 120 (2013) 14–17

Contents lists available at SciVerse ScienceDirect

Economics Letters

journal homepage: www.elsevier.com/locate/ecolet

Matching with quorums[☆]

Daniel Monte[a,*], Norovsambuu Tumennasan[b]

[a] Sao Paulo School of Economics - FGV, Rua Itapeva 474, 12 andar, Bela Vista, CEP 01332-000, São Paulo, Brazil

[b] Department of Economics and Business, Aarhus University, Aarhus, Denmark

HIGHLIGHTS

- We consider the problem of allocating agents to projects that have a minimum quorum.
- The serial dictatorship mechanism is not efficient or strategy proof.
- We propose a mechanism: serial dictatorship with project closures.
- The set of available projects evolves so that already-chosen projects are not closed.
- Our mechanism is strategy proof and efficient.

ARTICLE INFO

Article history:
Received 16 April 2012
Received in revised form
15 February 2013
Accepted 1 March 2013
Available online 13 March 2013

JEL classification:
C78
D61
D78
I20

Keywords:
Matching
Efficiency
Serial dictatorship

ABSTRACT

In the problem of allocating workers to different projects, where each project needs a minimum number of workers assigned to it, the serial dictatorship mechanism is neither strategy proof nor Pareto efficient. We therefore propose a strategy-proof and Pareto-efficient mechanism.

1. Introduction

We consider the problem of assigning agents to different projects that have a minimum quorum and a maximum capacity. Firms with multiple projects routinely face this problem: they must decide how to better allocate the workforce among different projects, and each worker must be allocated to at most one project. In addition, projects typically require a minimum number of workers in order to be completed successfully; hence, firms do not initiate a given project if the minimum quorum is not satisfied. This could be the case, for example, for projects that have a large fixed cost or that present economies of scale. At the same time, since allocating too many workers to a project is inefficient, the firms may require a maximal capacity for each project. Some educational institutions face a similar problem when assigning students to classes. Students must choose which classes to take in a given semester, during which there are many potential course offerings. Once all students registered for their classes – some of which are not mandatory – the courses will be offered only if a minimum quorum is satisfied.[1] At the same time, there is a limit on the class size due to physical space restrictions.

First we show that, in our setting, the well-known serial dictatorship mechanism is neither Pareto efficient nor strategy proof, regardless of how the agents are ordered. This is because agents who make the initial selections must consider the possibility that

☆ We thank the editor and an anonymous referee for insightful comments and many detailed suggestions.

* Corresponding author. Tel.: +55 1137993727.
E-mail addresses: daniel.monte@fgv.br (D. Monte), ntumennasan@econ.au.dk (N. Tumennasan).

[1] A director of graduate studies might find this problem to be a familiar one. In fact, as anecdotal evidence, some Ph.D. programs in the US regularly face this situation, in which a course will not be offered unless a predetermined minimal enrollment is met.

0165-1765/$ – see front matter © 2013 Published by Elsevier B.V.
http://dx.doi.org/10.1016/j.econlet.2013.03.007

the projects they select may never get opened due to insufficient enrollment. Consequently, agents might want to choose a less preferred project with a lower quorum. Motivated by this problem, we propose a strategy-proof and Pareto-efficient mechanism which we call *serial dictatorship with project closures*. Our mechanism is a stronger form of serial dictatorship in that the set of projects from which an agent can choose evolves so that already-chosen projects are opened.

The serial dictatorship mechanism in problems without any minimum quorum restriction satisfies many positive properties. In the house allocation setting, Svensson (1999) shows that it is the only deterministic mechanism that is strategy proof, nonbossy and neutral.[2] Abdulkadiroglu and Sönmez (1998) show that, for each Pareto-efficient allocation, there exists an ordering of the agents such that the serial dictatorship mechanism delivers the allocation. Furthermore, the core from random endowments is equivalent to the random serial dictatorship,[3] which provides an additional justification for the use of the random serial dictatorship in practice.

The study of matching problems with a minimum quorum is recent: Hamada et al. (2008) and Biró et al. (2010) study two-sided matching problems with a quorum in which both sides have well-defined preferences. They concentrate on stability, and show that stable matchings do not necessarily exist. The question of how to find a stable matching, if indeed it exists, is still under study. Meanwhile, the current paper studies efficiency and strategy-proofness.

Manea (2007) considers environments in which agents want to consume more than one object, and he studies a weak form of serial dictatorship: agents choose one object at a time according to an ordering in which any given agent could appear more than once. He shows that, in such environments, this weak version of the serial dictatorship mechanism fails both strategy-proofness and efficiency. In fact, Pápai (2001), Ehlers and Klaus (2003), and Hatfield (2009) establish a general result in such environments. The only strategy-proof Pareto-optimal and nonbossy mechanisms are the strong form of the sequential dictatorship: each agent chooses his/her favorite set of available objects according to a predefined ordering. In contrast, in our setting each agent is entitled to only one object, yet the serial dictatorship mechanism still fails in terms of efficiency and strategy-proofness.

2. Model

Finite set $I = \{1, \ldots, n\}$ is the set of agents/workers, and finite set $P = \{p_1, \ldots, p_m\}$ is the set of projects. Each project $p \in P$ has a maximum capacity k_p, with $1 \leq k_p \leq \infty$, and a minimum quorum $q_p \geq 1$. Both k_p and q_p are integer numbers, and throughout the paper we assume that $q_p \leq k_p$. This means that each project p cannot have more than k_p agents assigned to it. In addition, any project p with fewer than q_p agents assigned to it does not open. For convenience, we assume that $q_p \leq n$ for all $p \in P$; otherwise, project p would never open. Let $k = (k_p)_{p\in P}$ and $q = (q_p)_{p\in P}$.

Each agent $i \in I$ has a preference ordering $\succeq_i$ over the projects and the empty set $\emptyset$. The preference profile $(\succeq_i)_{i\in I}$ will be denoted by $\succeq$. Let $\succ_i$ and $\sim_i$ be the respective strict and indifference relations associated with $\succeq_i$. Throughout the paper, we maintain two assumptions about preferences: (1) each player's preference ordering $\succeq_i$ is strict, i.e., $p \succeq_i p'$ if either $p \succ_i p'$ or $p = p'$, and (2) $p \succ_i \emptyset$ for all $i \in I$ and $p \in P$; that is, every project is acceptable to any agent.

[2] The house allocation problem was first studied by Hylland and Zeckhauser (1979).

[3] This is a serial dictatorship in which the ordering is the outcome of a lottery.

A matching μ is a correspondence $\mu : I \cup P \to I \cup P$ such that (i) $\mu(i) \subseteq P$, for all $i \in I$; (ii) $\mu(p) \subseteq I$, for all $p \in P$; and (iii) $p \in \mu(i)$ if and only if $i \in \mu(p)$. If $\mu(i) = \emptyset$, we say that i is unmatched or unassigned at μ or that i is not assigned to any project at μ. Similarly, if $\mu(p) = \emptyset$, we say that p is unmatched at μ. For simplicity, we will write $\mu(i) = p$ instead of $\mu(i) = \{p\}$.

We will concentrate on matchings that assign each agent i to at most one project and each project p to at least q_p and at most k_p agents.

Definition 1 (*Feasible Matching*)**.** A matching μ is feasible if the following two conditions are satisfied:

(i) for all $p \in P$, either $q_p \leq |\mu(p)| \leq k_p$ or $|\mu(p)| = 0$; and
(ii) $|\mu(i)| \in \{0, 1\}$ for all $i \in I$.

According to our assumption that every project is acceptable to any agent, each feasible matching μ is individually rational; i.e., if $\mu(i) = p$, then $p \succ_i \emptyset$, for all $i \in I$. The definition of Pareto efficiency in our setting coincides with the standard one.

Definition 2 (*Pareto Efficiency*)**.** A *feasible* matching $\bar{\mu}$ Pareto dominates a feasible matching μ if

$$\bar{\mu}(i) \succ_i \mu(i) \quad \text{for at least one } i \in I \quad \text{and}$$
$$\bar{\mu}(j) \succeq_j \mu(j), \quad \text{for } \forall j \in I.$$

A matching μ is Pareto efficient if it is feasible and, in addition, there does not exist any feasible matching $\bar{\mu}$ that Pareto dominates μ.

A matching market M with quorums is given by the set of agents, the set of projects, the quorums and capacities of the projects, and the agents' preference profiles, i.e., $M = (I, P, q, k, \succeq)$. A mechanism φ for such markets is a mapping that assigns a feasible matching for each market.

A mechanism is Pareto efficient if it results in a Pareto- efficient matching for each market. Below, we define strategy-proofness.

Definition 3 (*Strategy-Proofness*)**.** Let φ be a mechanism for the set of matching markets with quorums. We say that φ is *manipulable* (individually) if there exist two markets $M = (I, P, k, q, \succeq)$ and $M' = (I, P, k, q, \succeq')$ and $i \in I$ such that (i) M' differs from M only in agent i's preference ordering (i.e., $\succeq_i \neq \succeq'_i$ and $\succeq_j = \succeq'_j$, for all $j \neq i$), and (ii) $\mu'(i) \succ_i \mu(i)$, where $\mu \equiv \varphi(M)$ and $\mu' \equiv \varphi(M')$. A mechanism φ is strategy proof if it is not manipulable.

3. Serial dictatorship

The algorithm known as serial dictatorship (SD) has been widely used in matching problems, both in theory and in practice. In environments without the minimum quorum restriction, the SD algorithm with respect to a given ordering of the agents in I is applied to market M as follows. Following the ordering of the agents, each agent is sequentially assigned to his/her most preferred project (with respect to his/her preferences in market M) among those projects that have not yet reached their maximum capacities. The algorithm terminates once the last agent in the ordering is assigned, or when all projects have reached their maximum capacities. The SD mechanism is the revelation mechanism which maps each market M to the matching produced by the SD algorithm for market M. This mechanism has been shown to be Pareto efficient and strategy proof (see, for example, Svensson, 1999).

In contrast, if we apply the SD algorithm to a market which has a minimum quorum restriction, the resulting matching may not be feasible. Specifically, there may be projects for which the number of agents assigned is lower than their minimum quorum. Therefore, for simplicity, we assume that, in the last step of the SD algorithm, all projects that do not satisfy the minimum quorum restriction are closed, and the agents who were tentatively matched to these projects are left unmatched. All other agents are allocated to the projects to which the SD algorithm assigned them.

Proposition 1 (*Failure of Pareto Efficiency and Strategy-Proofness*)**.** *The SD mechanism is neither strategy proof nor Pareto efficient.*

Proof. Consider a market M in which there are three agents, $I = \{i_1, i_2, i_3\}$, and five projects, $P = \{A, B, C, D, E\}$. Projects A, B, and C have a minimum quorum of 2 and no capacity restriction, while projects D and E have a minimum quorum of 1 and a maximum capacity of 1. Formally, $k_j = \infty$ and $q_j = 2$, for $j = A, B, C$; and $k_j = q_j = 1$, for $j = D, E$. Following Roth and Sotomayor (1990), we denote the preference ordering of agent i_k, for $k = 1, 2, 3$, by $P(i_k)$, which is listed below.

$P(i_1) : A, D, E, B, C, \emptyset$

$P(i_2) : B, D, E, A, C, \emptyset$

$P(i_3) : C, D, E, A, B, \emptyset.$

We will concentrate on the ordering in which each agent i's position is his/her index.

In market M, the SD mechanism results in the matching in which every agent is left unmatched. Because this matching is clearly Pareto dominated by any other feasible matching, the SD mechanism is not Pareto efficient. To see that the SD mechanism is not strategy proof, consider a market M' which differs from market M only in that project D is i_1's most preferred project in M'. In market M', the SD mechanism assigns i_1 to project D, which i_1 prefers to $\emptyset$ (under the preferences in market M).

We can perform a similar analysis for any ordering of the agents in M. In other words, regardless of how the agents are ordered, the SD mechanism does not result in an efficient matching in market M. Moreover, truth telling is not a dominant strategy in the game induced by the adoption of the SD mechanism for market M. □

4. Serial dictatorship with project closures

For the class of problems considered in this paper, we propose a mechanism that we call *serial dictatorship with project closures* (SDPC), which is a strong form of the SD mechanism. As we will show, this mechanism has many desirable properties: it produces a feasible matching, and it is both strategy proof and Pareto efficient.

Let $\{i_1, i_2, \ldots, i_n\}$ be an ordering of the agents in market $M = (I, P, k, q, \succeq)$. The SDPC algorithm for market M with respect to this ordering provides an assignment of the agents to the set of projects. The key notion is that of A_t, the set of active projects at step t, for all $t = 1, \ldots, n$.

Definition 4. Project p is active in step t (i.e., $p \in A_t$), for each $1 \leq t \leq n$, if

(i) whenever there are any steps before t, the number of agents assigned to p until step $t - 1$, inclusive, is less than or equal to its capacity k_p; and

(ii) in case i_t is assigned to p at step t then the remaining $n - t$ agents in the ordering are enough to fill p's quorum (if p has not yet completed its quorum) and the quorum of all projects to which some agent have already been assigned in the previous steps and which have not yet filled their quorum (if any such project exists).

Observe here that $A_1 = P$ (recall that $q_p \leq n$ for all $p \in P$), and $A_{t-1} \supseteq A_t$, for all $t = 2, \ldots, n$.

We are now ready to define the SDPC algorithm for market M with respect to the ordering $\{i_1, i_2, \ldots, i_n\}$. At any step t of the algorithm, for each $1 \leq t \leq n$, agent i_t is assigned to his/her most preferred project in A_t, if this set is nonempty, and is left unassigned if A_t is empty. In the latter case, $A_t = A_{t+1} = \cdots = A_n = \emptyset$ and the agents $i_t, i_{t+1}, \ldots, i_n$ are left unassigned.

We will use the following notation for the rest of the paper: P_t is the set of the projects that have had at least one agent assigned to them at some step $t' \leq t$ of the SDPC algorithm, and the notation $\theta_p(t)$ denotes the number of agents that have been allocated to project p at some step $t' \leq t$. Using this notation we can formalize the two conditions used in the definition of an active project as follows: a project p is active in step $t = 2, \ldots, n$ (recall that each $p \in P$ is active in step 1) if

(i) $\theta_p(t-1) < k_p$; and

(ii) $\max\{q_p - \theta_p(t-1) - 1, 0\} + \sum_{p' \in P_{t-1} \setminus \{p\}} \max\{q_{p'} - \theta_{p'}(t-1), 0\} \leq n - t$.

The SDPC mechanism is the revelation mechanism that maps each market M to the matching that the SDPC algorithm produces for M.

The following lemma plays an instrumental role for our main theorem.

Lemma 1. *Consider any $p \in P$ and $t \geq 2$. If $p \in P_{t-1}$ and $\theta_p(t-1) < q_p$, then $p \in A_t$.*

Proof. In contrast to the lemma, suppose that there exist $p \in P$ and $t \geq 2$ such that $p \in P_{t-1}$, $\theta_p(t-1) < q_p$, and $p \notin A_t$. Let τ be the last step prior to t (i.e., $\tau < t$) in which $A_\tau \neq \emptyset$. Clearly, $\tau \geq 1$, because $A_1 = P$. Observe here that $P_\tau = P_{t-1}$ because either (i) $\tau = t - 1$ or (ii) $\tau < t - 1$ and $A_k = \emptyset$, for all $k > \tau$. Consequently, $p \in P_\tau$. In addition, because $p \notin A_t$ and $A_{\tau+1}$ is either A_t or $\emptyset$, it must be that $p \notin A_{\tau+1}$. Because $A_\tau \neq \emptyset$, agent i_τ cannot be unmatched, due to the fact that every project is acceptable to every agent. Let p' be the project i_τ is matched. Clearly, $p' \in A_\tau$. Consequently, by the definition of active project, if i_τ is assigned to p' then the remaining $n - \tau$ agents are enough to complete the quorum requirements of all the projects in P_τ, in particular the quorum requirement of project p. Now, use Definition 4(ii) to see that p must be active at step $\tau + 1$, which contradicts the fact that $p \notin A_{\tau+1}$. □

Theorem 1 (*SDPC: Strategy-Proofness and Efficiency*)**.** *The SDPC mechanism is strategy proof and Pareto efficient.*

Proof. First, we show that the SDPC mechanism results in a feasible matching in each market M. Let the SDPC mechanism yield matching μ for market M. By the definition of the SDPC algorithm, it is clear that $|\mu(i)| \in \{0, 1\}$, for all $i \in I$. Moreover, in any step of the SDPC algorithm no agent is assigned to projects that are not active, and, by Definition 4(i), each project p is not active as soon as it reaches its capacity. Thus, $|\mu(p)| \leq k_p$. Hence, we are left to prove that, for all $p \in P$, either $|\mu(p)| = 0$ or $|\mu(p)| \geq q_p$. Then, let $p \in P$. By the definition of the SDPC algorithm it is clear that $|\mu(p)| = \theta_p(n)$.

In the last step of the algorithm there is only one agent i_n to be assigned. Suppose first that $p \notin A_n$. Then i_n cannot be assigned to p. If $p \notin P_{n-1}$, then $p \notin P_n$, and, consequently, $|\mu(p)| = 0$. If $p \in P_{n-1}$, Lemma 1 implies that $\theta_p(n-1) = \theta_p(n) \geq q_p$ (otherwise, $\theta_p(n-1) < q_p$, so $p \in A_n$, a contradiction). Suppose now that $p \in A_n$. The definition of A_n and the fact that there remains only one agent to be assigned in step n imply that, if p has not filled its quorum, then $A_n = \{p\}$; that is, i_n must be assigned to p in step n. In this case, the definition of active project implies that p will complete its quorum with i_n. Hence, at the end of the algorithm every project p will have filled its quorum. Consequently, $\theta_p(n) \geq q_p$. This completes the proof that the SDPC algorithm produces a feasible matching.

To show that the SDPC mechanism is strategy proof, fix any market M. Let $\bar{M}$ be a market which differs from M only in the preferences of some agent i_k. Suppose that the SDPC mechanism produces matchings μ and $\bar{\mu}$ in markets M and $\bar{M}$, respectively. The critical observation for the proof is that the definition of A_k does not use the preferences stated by i_k. Then, since the preferences of the agents other than i_k are the same in both markets, the set of active projects at step k is the same in both markets. Therefore, if $A_k = \emptyset$, we must have that i_k is unmatched in both markets; if $A_k \neq \emptyset$, then the definition of the SDPC algorithm implies that $\mu(i_k) \succeq_{i_k} \bar{\mu}(i_k)$, where $\succeq_{i_k}$ is agent i_k's preferences in market M. This completes the proof that the SPDC mechanism is strategy proof.

Now we show that the SDPC mechanism is Pareto efficient. Suppose by way of contradiction that the SDPC mechanism results in a non-Pareto efficient matching in some market M. Let the matching produced by the SDPC algorithm for market M be denoted by μ, and let μ be Pareto dominated by a feasible matching $\bar{\mu}$ in market M. Let i_k be the first agent in the algorithm, following the prespecified ordering, for whom $\bar{\mu}(i_k) \neq \mu(i_k)$. Then $\bar{\mu}(i_t) = \mu(i_t)$, for all $t < k$. Now the strictness of the preferences implies that $\bar{\mu}(i_k) \succ_{i_k} \mu(i_k)$; thus, i_k is matched to some project at $\bar{\mu}$, which is not active at step k of the algorithm. Let $\bar{\mu}(i_k) = \bar{p}$.

Since $\bar{\mu}\,(i_k) = \bar{p}$, the feasibility of $\bar{\mu}$ implies that the quorum of $\bar{p}$ has been completed at $\bar{\mu}$. On the other hand, all projects in P_{k-1} are open at $\bar{\mu}$. In fact, let $p' \in P_{k-1}$. Then p' has been assigned at μ to some agent i_t, with $t < k$. By definition of i_k it follows that $\mu(i_t) = \bar{\mu}(i_t) = p'$, and so p' is open at $\bar{\mu}$.

Now observe that, due to the fact that $\mu\,(i_t) = \bar{\mu}\,(i_t)$ for all $t < k$, the total number of agents i_t, with $t < k$, assigned to the projects of $\{\bar{p}\} \cup P_{k-1}$ under both matchings is $k-1$. Then, the number λ of agents, other than $i_1, \ldots, i_{k-1}$, that should be used to complete the quorum of the projects in $\{\bar{p}\} \cup P_{k-1}$ is the same in both matchings.

Clearly, $\bar{p}$ cannot have filled its capacity prior to step k; otherwise, $\bar{\mu}$ is not feasible. Thus, because $\bar{p} \notin A_k$, Definition 4 implies that, if i_k is assigned to $\bar{p}$ at step k of the algorithm, then the remaining $n-k$ agents are not enough to complete the quorum of $\bar{p}$ and of all projects in P_{k-1}. That is, $\lambda - 1 > n - k$. On the other hand, since the projects of $\{\bar{p}\} \cup P_{k-1}$ complete their quorum at $\bar{\mu}$, then $\lambda \leq n-k+1$, which is a contradiction. Hence, the proof is complete. □

Here we remark that in some markets the SD and the SDPC mechanisms (with respect to the same ordering of the agents) produce the same matchings. In such markets, Theorem 1 means that the SD mechanism results in a Pareto-efficient matching and, in addition, truth telling is a dominant strategy for every agent in the game induced by the adoption of the SD mechanism. In the proposition below, we identify a class of such markets.

Proposition 2. *Consider the market $M = (I, P, k, q, \succeq)$. If $\sum_{p \in P} k_p \leq n$, then the SD mechanism applied to M results in a Pareto-efficient matching. Furthermore, truth telling is a dominant strategy for every agent in M in the game induced by the adoption of the SD mechanism for market M.*

Proof. Because $\sum_{p \in P} k_p \leq n$ in market M, in each step of the SDPC algorithm each project which has not filled its capacity must be active. Thus, at any step $t \geq 1, \ldots, n$ the set of projects available to agent i_t is the same under both the SD and SDPC algorithms. Thus, the SD and SDPC algorithms produce the same matching for M. This and Theorem 1 complete the proof. □

In settings without the minimum quorum restriction, Abdulkadiroglu and Sönmez (1998) show that, in each market, and for each Pareto-optimal matching, there is a corresponding ordering of the agents such that the SD algorithm yields this matching. However, this is not the case for our SDPC algorithm, as seen in the following example.

Example (*Pareto-Efficient Matching Not Produced by the SDPC*)**.** Consider a market $M = (I, P, q, k, \succeq)$ in which $I = \{i_1, i_2, \ldots, i_n\}$, $P = \{\bar{p}, p_1, p_2, \ldots, p_n\}$, $q_p = n$, and $k_p = \infty$ for each p. The preference ordering of agent i_k, for $k = 1, \ldots, n$, is denoted by $P(i_k)$ and listed below.

$$P(i_1) : p_2, p_3, \ldots, p_n, \bar{p},\ p_1,\ \emptyset$$
$$P(i_k) : p_1, p_2, \ldots, p_{i-1}, p_{i+1}, \ldots, p_n, \bar{p}, p_i, \emptyset \quad \text{for all } 1 < k < n$$
$$P(i_n) : p_1, p_2, \ldots, p_{n-1}, \bar{p}, p_n, \emptyset.$$

In market M, the matching μ with $\mu\,(i) = \bar{p}$ for $\forall i \in \{i_1, i_2, \ldots, i_n\}$ is Pareto efficient because any other feasible matching makes at least one agent worse off. However, for any ordering of the agents that is considered, μ is not produced by the SDPC algorithm when it is applied to market M. In fact, for any ordering in which agent 1 is the first one to be assigned, the SDPC algorithm yields a matching in which all the agents are assigned to p_2. For any ordering in which agent 1 is not the first one to be assigned, the SDPC algorithm yields a matching in which all the agents are assigned to p_1.

Acknowledgment

We are very thankful to Marco Chiarandini who brought the matching problem with quorums to our attention.

References

Abdulkadiroglu, A., Sönmez, T., 1998. Random serial dictatorship and the core from random endowments in house allocation problems. Econometrica 66 (3), 689–702.

Biró, P., Fleiner, T., Irving, R.W., Manlove, D.F., 2010. The college admissions problem with lower and common quotas. Theoretical Computer Science 411 (34–36), 3136–3153.

Ehlers, L., Klaus, B., 2003. Coalitional strategy-proof and resource-monotonic solutions for multiple assignment problems. Social Choice and Welfare 21 (2).

Hamada, K., Iwama, K., Miyazaki, S., 2008. The hospitals/residents problem with quota lower bounds. in: Proc. MATCH-UP (satellite workshop of ICALP 2008). pp. 55–66.

Hatfield, J., 2009. Strategy-proof, efficient, and nonbossy quota allocations. Social Choice and Welfare 33 (3), 505–515.

Hylland, A., Zeckhauser, R., 1979. The efficient allocation of individuals to positions. Journal of Political Economy 87 (2), 293–314.

Manea, M., 2007. Serial dictatorship and Pareto optimality. Games and Economic Behavior 61 (2), 316–330.

Pápai, S., 2001. Strategyproof and nonbossy multiple assignments. Journal of Public Economic Theory 3 (3).

Roth, A., Sotomayor, M., 1990. Two-Sided Matching: A Study in Game-Theoretic Modeling and Analysis. In: Econometric Society Monographs, Cambridge University Press.

Svensson, L., 1999. Strategy-proof allocation of indivisible goods. Social Choice and Welfare 16 (4).