Economics Letters 121 (2013) 341–345

Contents lists available at ScienceDirect

Economics Letters

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

The computational complexity of random serial dictatorship

Haris Aziz [a,*], Felix Brandt [b,1], Markus Brill [b,1]

[a] NICTA and UNSW, 223 Anzac Parade, Sydney, NSW 2033, Australia

[b] Institut für Informatik, Technische Universität München, 85748 Garching, Germany

HIGHLIGHTS

- We study the computational complexity of random serial dictatorship (RSD).
- We show that computing the RSD lottery is #P-complete.
- We propose an efficient algorithm that computes the support of the RSD lottery.

ARTICLE INFO

Article history:
Received 12 June 2013
Received in revised form
26 August 2013
Accepted 6 September 2013
Available online 14 September 2013

JEL classification:
C63
C70
C71
C78

Keywords:
Social choice theory
Random serial dictatorship
Random priority
Computational complexity
Assignment problem

ABSTRACT

In social choice settings with linear preferences, *random dictatorship* is known to be the only social decision scheme satisfying strategyproofness and *ex post* efficiency. When also allowing indifferences, *random serial dictatorship* (*RSD*) is a well-known generalization of random dictatorship that retains both properties. *RSD* has been particularly successful in the special domain of random assignment where indifferences are unavoidable. While *executing RSD* is obviously feasible, we show that *computing* the resulting probabilities is #P-complete, and thus intractable, both in the context of voting and assignment.

1. Introduction

Social choice theory studies how a group of agents can make collective decisions based on the – possibly conflicting – preferences of its members. In the most general setting, there is a set of abstract *alternatives* over which each agent entertains *preferences*. A *social decision scheme* aggregates these preferences into a probability distribution (or *lottery*) over the alternatives.

Perhaps the most well-known social decision scheme is *random dictatorship*, in which one of the agents is uniformly chosen at random and then picks his most preferred alternative. Gibbard (1977) has shown that random dictatorship is the only social decision scheme that is strategyproof and *ex post* efficient, i.e., it never puts positive probability on Pareto dominated alternatives. Note that random dictatorship is only well-defined when there are no ties in the agents' preferences. However, ties are unavoidable in many important domains of social choice, such as assignment, matching, and coalition formation, since agents are assumed to be indifferent among all outcomes in which their assignment, match, or coalition is the same (e.g., Sönmez and Ünver, 2011).

In the presence of ties, random dictatorship is typically extended to *random serial dictatorship* (*RSD*), where dictators are invoked sequentially and ties between most-preferred alternatives are broken by subsequent dictators.[2] *RSD* retains the important properties of *ex post* efficiency and strategyproofness and is well-established in the context of random assignment (see e.g., Svensson, 1994; Abdulkadiroğlu and Sönmez, 1998; Bogomolnaia and Moulin, 2001; Crès and Moulin, 2001).

* Corresponding author. Tel.: +61 2 8306 0490.
E-mail addresses: haris.aziz@nicta.com.au (H. Aziz), brandtf@in.tum.de (F. Brandt), brill@in.tum.de (M. Brill).
1 Tel.: +49 89 289 17512.

2 *RSD* is referred to as *random priority* by Bogomolnaia and Moulin (2001).

0165-1765/$ – see front matter © 2013 Elsevier B.V. All rights reserved.
http://dx.doi.org/10.1016/j.econlet.2013.09.006

In this paper, we focus on two important domains of social choice: (1) the *voting setting*, where alternatives are candidates and agents' preferences are given by rankings over candidates, and (2) the aforementioned *assignment setting*, where each alternative corresponds to an assignment of houses to agents and agents' preferences are given by rankings over houses. Whereas agents' preferences over alternatives are listed explicitly in the voting setting, this is not the case in the assignment setting. However, preferences over houses can be easily extended to preferences over assignments by assuming that each agent only cares about the house assigned to himself and is indifferent between all assignments in which he is assigned the same house. As a consequence, the assignment setting is a special case of the voting setting. However, due to the different representations, *computational* statements do not carry over from one setting to the other.

In various settings, the probability that a social decision scheme assigns to an alternative is interpreted as the fraction of time or another resource that is allocated to the alternative (see, e.g., Moulin, 2003; Bogomolnaia et al., 2005). Similarly, in the assignment setting, the probability with which an agent is allocated a certain object is often viewed as the fraction of the object that this agent receives or the fraction of time that the agent is allowed to use the shared object. As a consequence, computing *RSD* lotteries is of great importance and has applications in domains such as scheduling of resources (see, e.g., Crès and Moulin, 2001).

We examine the computational complexity of *RSD* and show that computing the *RSD* lottery is #P-complete *both* in the voting setting and in the assignment setting. Loosely speaking, #P is the counting equivalent of NP—the class of decision problems whose solutions can be verified in polynomial time. #P-completeness is commonly seen as strong evidence that a problem cannot be solved in polynomial time.

As mentioned above, neither of the two results implies the other. We furthermore present a polynomial-time algorithm to compute the *support* of the *RSD* lottery in the voting setting. This is not possible in the assignment setting, because the support of the *RSD* lottery might be of exponential size. However, we can decide in polynomial time whether a given alternative (i.e., an assignment) is contained in the support or not.

2. Preliminaries

In the general social choice setting, there is a set $N = \{1, \dots, n\}$ of *agents*, who have preferences over a finite set A of *alternatives*. The preferences of agent $i \in N$ are represented by a complete and transitive *preference relation* $R_i \subseteq A \times A$. The interpretation of $(a, b) \in R_i$, usually denoted by $a \, R_i \, b$, is that agent i values alternative a at least as much as alternative b. In accordance with conventional notation, we write P_i for the strict part of R_i, i.e., $a \, P_i \, b$ if $a \, R_i \, b$ but not $b \, R_i \, a$, and I_i for the symmetric part of R_i, i.e., $a \, I_i \, b$ if $a \, R_i \, b$ and $b \, R_i \, a$. A *preference profile* $R = (R_1, \dots, R_n)$ is an n-tuple containing a preference relation R_i for every agent $i \in N$.

A preference relation R_i is *linear* if $a \, P_i \, b$ or $b \, P_i \, a$ for all distinct alternatives $a, b \in A$. A preference relation R_i is *dichotomous* if $a \, R_i \, b \, R_i \, c$ implies $a \, I_i \, b$ or $b \, I_i \, c$.

We let Π^N denote the set of all permutations of N and write a permutation $\pi \in \Pi^N$ as $\pi = \pi(1) \dots \pi(n)$. For $k \leq n$, we furthermore let $\pi|_k$ denote the prefix of π of length k, i.e., $\pi|_k = \pi(1) \dots \pi(k)$.

If R_i is a preference relation and $B \subseteq A$ a subset of alternatives, then $\max_{R_i}(B) = \{a \in B : a \, R_i \, b \text{ for all } b \in B\}$ is the set of most preferred alternatives from B according to R_i. Hence, $a \, I_i \, b$ for all $a, b \in \max_{R_i}(B)$ and $a \, P_i \, b$ for all $a \in \max_{R_i}(B)$, $b \in B \setminus \max_{R_i}(B)$.

In order to define the social decision scheme known as *random serial dictatorship* (*RSD*), let us first define its deterministic variant *serial dictatorship* (*SD*). For a given preference profile R and a permutation $\pi \in \Pi^N$, $SD(R, \pi)$ is defined via the following procedure. Agent $\pi(1)$ chooses the set of most preferred alternatives from A, $\pi(2)$ chooses his most preferred alternatives from the refined set and so on until all agents have been considered. The resulting set of alternatives is returned. Formally, $SD(R, \pi)$ is defined inductively via $SD(R, \pi|_0) = A$ and $SD(R, \pi|_i) = \max_{R_{\pi(i)}}(SD(R, \pi|_{i-1}))$.

Throughout this paper, we assume that the preferences of the agents are such that there is no pair $a, b \in A$ with $a \neq b$ and $a \, I_i \, b$ for all $i \in N$.[3] This assumption ensures that the set $SD(R, \pi)$ is always a singleton. We will usually write $SD(R, \pi) = a$ instead of $SD(R, \pi) = \{a\}$.

We are now ready to define *RSD*. For a given preference profile R, *RSD* returns $SD(R, \pi)$, where π is chosen uniformly at random from Π^N. The probability $RSD(R)(a)$ of alternative $a \in A$ is thus proportional to the number of permutations π for which $SD(R, \pi) = a$:

$$RSD(R)(a) = \frac{1}{n!} \left| \left\{ \pi \in \Pi^N : SD(R, \pi) = a \right\} \right| .$$

We refer to the probability $RSD(R)(a)$ as the *RSD probability* of alternative a and to the probability distribution $RSD(R)$ as the *RSD lottery*.

Our proofs leverage the fact that a certain matrix related to the Pascal triangle has a non-zero determinant.

Lemma 1 (*Bacher, 2002*)**.** *The $n \times n$ matrix $M = (m_{ij})_{i,j}$ given by $m_{ij} = (i + j - 2)!$ has a non-zero determinant. That is,*

$$\det \begin{pmatrix} 0! & 1! & \cdots & (n-1)! \\ 1! & 2! & \cdots & n! \\ \vdots & \vdots & \ddots & \vdots \\ (n-1)! & n! & \cdots & (2n-2)! \end{pmatrix} \neq 0.$$

3. Voting setting

A *voting problem* is given by a triple (N, A, R), where $N = \{1, \dots, n\}$ is a set of agents, A is a set of alternatives, and $R = (R_1, \dots, R_n)$ is a preference profile that contains, for each agent i, a preference relation on the set of alternatives. The goal is to choose an alternative that is socially acceptable according to the preferences of the agents.

If each agent has a unique most preferred alternative, the *RSD* lottery can be computed in linear time. Therefore, computational aspects of *RSD* only become interesting when at least some of the agents express indifferences among their most preferred alternatives. The straightforward approach to compute the *RSD* lottery involves the enumeration of permutations. This approach obviously takes exponential time. At first sight, it seems that even finding the support of the *RSD* lottery requires the enumeration of all permutations. However, we outline a surprisingly simple algorithm that checks in polynomial time whether a given alternative a is contained in the support (Algorithm 1).

The algorithm is based on a greedy approach and maintains a working set of alternatives A' and a working set of agents N', which are initialized as A and N, respectively. If no agent in N' has a as a most preferred alternative in A', then the algorithm returns "no". Otherwise let $i \in N'$ be the smallest index such that agent i has a as

[3] In the assignment setting, this assumption always holds if the agents have linear preferences over houses. *SD* (and *RSD*) can be defined without this assumption (see, e.g., Aziz et al., 2013).