

Approximating Random Serial Dictatorship: Hardness and Markov Chains

HARIS AZIZ, UNSW Sydney, Australia

BART DE KEIJZER, King's College London, United Kingdom

SEBASTIAN HOMRIGHAUSEN, Aarhus University, Denmark

ALI POURMIRI, Sydney, Australia

We consider a setting in which a randomised social choice rule selects one of the alternatives on the basis of preferences expressed by agents. Our focus is on *Random Serial Dictatorship* (RSD), a central rule in social choice theory. Computing RSD shares exactly is known to be $\#P$ -hard but the complexity of approximating it has been open for more than a decade. We resolve this question negatively: unless $NP \subseteq BPP$, there is no FPRAS for computing the RSD share of a given listed alternative, even for dichotomous preferences. We also develop a Markov Chain Monte Carlo framework for approximately counting the permutations that select a target alternative. The framework gives an FPRAS whenever the relevant suffix-restricted state spaces admit rapidly mixing uniform samplers. We analyse the natural Insertion Markov Chain and construct dichotomous instances on which it has exponentially small conductance. We then identify structured domains on which efficient approximation is possible. These include instances whose selecting permutations are linear extensions of a polynomial-time constructible partial order, single-plateaued preferences on a common axis, and dichotomous single-peaked (interval-approval) preferences. On the latter two domains, we prove rapid mixing of the Insertion Markov Chain on every suffix-restricted state space.

1 INTRODUCTION

Random Serial Dictatorship (RSD) is a mechanism of fundamental importance in social choice theory, because of its attractive axiomatic properties. The mechanism (or also *social choice rule*) applies universally to several important preference aggregation problems, including assignment and coalition formation. More formally, RSD applies to any setting that can be stated in the form of a set of n agents who express preferences over a set of m alternatives, and where the preferences of an agent are formed by a total order over a partition of the set of alternatives, where each set in the partition forms an equivalence (or indifference) class.

The RSD mechanism is relatively simple to state: It selects an alternative by iteratively reducing the set of alternatives under consideration. Starting from the set of all m alternatives, it restricts the current set of considered alternatives by restricting it to the set of most preferred alternatives (among the remaining ones) of an agent, iterating over the agents in a uniformly drawn random order. This procedure is clearly very easy to implement in an inherently computationally efficient way.

Random Serial Dictatorship satisfies anonymity, ex-post Pareto efficiency, and strategyproofness, a particularly compelling combination of axioms. In the assignment setting, it is uniquely characterised by these together with additional basic properties [Bade, 2020]. Considering the importance of the aforementioned three axioms, random serial dictatorship, as a social choice rule, has received ample attention in social choice theory. Furthermore, additional desirable properties of RSD have been identified, related to incentives for agents to participate in voting [Brandl et al., 2015] and to fairness [Airiau et al., 2019].

Our goal, in this paper, is to understand one of the most fundamental computational problems associated with the RSD mechanism: computing, with high precision, the probability with which each alternative is selected [Aziz et al., 2013, 2014]. Aziz et al. [2013] proved that checking whether the RSD share of a given alternative is positive can be solved in polynomial time, but computing an alternative’s exact RSD share is $\#P$ -hard even for dichotomous preferences. Equivalently, the time or budget share of an alternative is a computationally intractable problem. However this did not settle the complexity of approximation of RSD share.

In many social choice applications, RSD is used as a *deterministic* strategyproof and fair rule for *fractional selection*. In these applications, such as time or budget sharing [Aziz and Stursberg, 2014, Brandt, 2018], fair mixing [Aziz et al., 2020], portioning [Airiau et al., 2019], and seat apportionment [Tullock, 1967], the actual probability of RSD selecting an alternative is taken to be proportional to the “appropriate” time or budget share. Airiau et al. [2019] give several examples including a conference board deciding on how much time to assign to different conference activities, or company employees voting on which charities should receive a donation from the company. In these situations, computing accurate RSD shares is of critical importance, in the same way that computing the set of winners of a voting rule is. Computing these shares is the central problem that we study.

Contributions. The RSD lottery can be computed exactly by enumerating all agent permutations and averaging the corresponding serial-dictatorship outcomes. This immediately suggests sampling permutations uniformly at random. Such a method gives an additive estimate, but it can perform extremely poorly when the target alternative has exponentially small positive probability: with high probability, no sampled permutation selects the target, and the estimate is zero. We give a simple family illustrating this obstruction. We also examine a natural target-oriented sampling procedure adapted from the polynomial-time support algorithm of Aziz et al. [2013]. Although the procedure always returns a selecting permutation when one exists, it is biased and therefore cannot directly be used for approximate counting.

Our first main result resolves this model-specific, long-standing approximation question. Unless $NP \subseteq BPP$, no FPRAS exists for the RSD share of a given explicitly listed alternative. The result already holds for dichotomous preferences when the number of alternatives is linear in the number of agents, all agents who disapprove the target alternative have identical preferences, and the target is either the unique serial-dictatorship outcome or is not selected at all. NP is the class of problems whose solutions can be verified efficiently, and BPP is the class of problems solvable efficiently by randomized algorithms with bounded error. Since $NP \subseteq BPP$ is widely believed to be false, a result showing that an FPRAS for a problem implies $NP \subseteq BPP$ is standard evidence of inapproximability.

The negative result rules out a general-purpose FPRAS, but it does not make sampling methods irrelevant. We present a self-reduction framework that estimates the number of selecting permutations through a telescoping product of suffix-restriction ratios. At each stage, the algorithm approximately samples from the current state space, estimates simultaneously the probability of every agent occupying the final unfixed position, fixes an empirically frequent agent in that position, and continues with the resulting restricted state space. The simultaneous estimation is essential because the restricted state space is chosen adaptively from the samples.

We study the lazy Insertion Markov Chain as a natural uniform sampler for these state spaces. We prove a closure lemma showing that the leftmost-correction insertion between two selecting permutations remains selecting. This yields irreducibility and supplies canonical paths for the positive results. Nevertheless, the chain is not rapidly mixing in general: we give a family of dichotomous instances whose selecting permutations form two large regions connected through an exponentially smaller bottleneck.

Finally, we identify well-motivated domains on which the framework yields an FPRAS. If the selecting permutations are exactly the linear extensions of a polynomial-time constructible partial order, then the suffix-restricted spaces are linear-extension spaces and known rapidly mixing samplers apply. We give an explicit height-two dichotomous construction encoding precedence constraints. We then introduce c -prefix compressibility: every nonempty suffix-restricted space contains all permutations beginning with some ordered tuple of at most c agents. For fixed c , a direct canonical-path count proves rapid mixing of the Insertion Markov Chain on these spaces. Single-plateaued preferences and dichotomous single-peaked (interval-approval) preferences on a common axis [Elkind et al., 2022, Puppe and Slinko, 2026] are 2-prefix-compressible, so the chain supplies approximately uniform selecting permutations and an FPRAS through our framework.

Related Work. Random serial dictatorship has been examined in several preference aggregation domains including general social choice and the assignment problem. Bogomolnaia and Moulin [2001] studied random serial dictatorship and proposed an alternative mechanism called *probabilistic serial* that is not strategyproof but still has desirable efficiency and fairness properties. From a non-computational perspective, the assignment problem can be viewed as a special case of the social choice problem as each agent’s preferences over individual items naturally give rise to preferences over an exponential number of assignments.

Saban and Sethuraman [2015] considered random priority in the assignment problem. They proved that counting the agent permutations that assign a given object to a given agent is #P-complete and that the corresponding marginal-probability problem admits no FPRAS unless $NP = RP$. This is the relevant prior inapproximability result, but it concerns an implicitly represented matching space and an agent-object marginal, rather than the probability of one alternative in an explicitly represented social-choice instance. Our negative result therefore establishes the corresponding barrier for the voting/social-choice representation, already on dichotomous preferences and under the restrictions stated in Theorem 1.

More recently, Caragiannis and Homrighausen [2024] studied direct sampling for estimating the aggregate expected social welfare and social cost of RSD in assignment settings. Their estimands and guarantees are different from the multiplicative approximation of one alternative’s RSD share considered here: an exponentially rare target can defeat direct permutation sampling even when an aggregate expectation is accurately estimable from relatively few samples.

In general social choice with strict preferences, RSD is equivalent to *uniform random dictatorship*: it selects an agent uniformly at random and returns that agent’s top alternative. Gibbard’s characterization yields, under strategyproofness and ex-post Pareto efficiency, a random dictatorship with possibly non-uniform weights; anonymity forces the weights to be equal [Gibbard, 1977]. Under weak preferences, RSD is the natural extension preserving these axiomatic properties, and all other known rules violate one of anonymity, strategyproofness, and Pareto optimality [Aziz et al., 2018].

The central problem we consider is a counting problem at its core: we want to approximately count the permutations for which serial dictatorship selects a given alternative, since dividing this count by $n!$ yields the RSD share. The algorithmic techniques we use fall within the scope of Markov Chain Monte Carlo methods. Karp and Luby [Karp and Luby, 1983] pioneered Monte Carlo approximate counting for DNF satisfying assignments, and Jerrum and Sinclair [Jerrum and Sinclair, 1989] gave a fully polynomial randomised approximation scheme for the permanent on suitable classes of 0–1 matrices, using a general framework that turns a uniform sampler into an approximate counter for self-reducible structures [Jerrum et al., 1986]; it has since been applied to countless counting problems, such as graph colourings and independent sets.

Another closely related counting problem is that of counting the linear extensions of a partial order. Exact counting is #P-complete [Brightwell and Winkler, 1991]. The approximation and sampling history predates the fastest chain used here: Karzanov and Khachiyan [1991] analysed an order Markov chain, and Matthews [1991] gave a further polynomial-time almost-uniform generation method. Bubley and Dyer [1999] later introduced a faster linear-extension chain and proved rapid mixing by path coupling. We use that sampler for the poset-representable special case below. For general RSD selecting-permutation spaces, however, the structural coupling arguments for linear extensions do not transfer directly.

2 PRELIMINARIES

2.1 Social Choice and Random Serial Dictatorship

We first define the social choice setting and RSD formally. For $a \in \mathbb{N}$, we use $[a]$ to denote the set of natural numbers $\{1, \dots, a\}$. When S is a set, we write $\#S$ for its cardinality. A *social choice instance* consists of a finite set of *agents* N and a finite set of *alternatives* A . We typically take $N = [n]$ and $A = [m]$, and use n and m for the respective cardinalities when the instance is clear from context. Each agent $i \in N$ has a complete and transitive preference relation $\succsim_i$ over A . The preference profile is $\succsim = (\succsim_i)_{i \in N}$, and an instance is the triple $(N, A, \succsim)$. By $a \succsim_i b$, we mean that agent i regards alternative a as at least as desirable as alternative b . The strict part is denoted by $\succ_i$: $a \succ_i b$ if $a \succsim_i b$ but not $b \succsim_i a$. The symmetric part is denoted by $\sim_i$: $a \sim_i b$ if both $a \succsim_i b$ and $b \succsim_i a$. A preference relation is *strict* or *linear* if $a \succ_i b$ or $b \succ_i a$ for all distinct $a, b \in A$. It is *dichotomous* if it has at most two indifference classes; equivalently, $a \succsim_i b \succsim_i c$ implies $a \sim_i b$ or $b \sim_i c$ for all $a, b, c \in A$. We let Π^N denote the set of all permutations of N and write a permutation $\pi \in \Pi^N$ as $\pi = (\pi(1), \dots, \pi(n))$. Thus, we view π as a sequence of agents, and for $i \in [n]$, we treat $\pi(i)$ as the agent on the i -th position of this sequence. If $\succsim_i$ is a preference relation and $B \subseteq A$ a subset of alternatives, then $\max_{\succsim_i}(B) = \{a \in B : a \succsim_i b \text{ for all } b \in B\}$ is the set of most preferred alternatives from B according to $\succsim_i$.

Definition 1 (Random serial dictatorship). Let N be a set of agents, let A be a set of alternatives, and let $\succsim = (\succsim_i)_{i \in N}$ be a preference profile. For a permutation π of N , define $S_0(\pi) = A$ and, for every $t \in \{1, \dots, |N|\}$, $S_t(\pi) = \max_{\succsim_{\pi(t)}} S_{t-1}(\pi)$, where $\max_{\succsim_i} S$ denotes the set of alternatives that agent i most prefers among the alternatives in S .

Random serial dictatorship draws a permutation π uniformly at random and then selects an alternative uniformly at random from $S_{|N|}(\pi)$.

For any given permutation π of N , we define $SD(N, A, \succsim, \pi)$ as the outcome of applying serial dictatorship. The *RSD share* $s_j(N, A, \succsim)$ of alternative $j \in A$ is

$$\begin{aligned} s_j(N, A, \succsim) &= \Pr[SD(N, A, \succsim) = j] \\ &= \frac{1}{n!} \sum_{\pi \in \Pi^N} \frac{\mathbf{1}[j \in SD(N, A, \succsim, \pi)]}{\#SD(N, A, \succsim, \pi)}. \end{aligned}$$

The *RSD lottery* is $s(N, A, \succsim) = (s_1(N, A, \succsim), \dots, s_m(N, A, \succsim))$. Henceforth, we omit explicit references to $(N, A, \succsim)$ when they are clear from context.

We illustrate how RSD works with the aid of a simple example.

Example 1 (Illustration of RSD). Consider an instance with agents $\{1, 2, 3\}$, alternatives $\{a, b, c, d\}$ and the following preference profile.

$$1: a \sim_1 b \sim_1 c \succ_1 d \qquad 2: b \sim_2 d \succ_2 a \succ_2 c \qquad 3: c \succ_3 a \sim_3 b \sim_3 d$$

Then let us consider the *SD* outcomes for each permutation over the voters.

$$\begin{array}{lll} (1, 2, 3): \{b\} & (2, 1, 3): \{b\} & (3, 1, 2): \{c\} \\ (1, 3, 2): \{c\} & (2, 3, 1): \{b\} & (3, 2, 1): \{c\} \end{array}$$

Thus the RSD lottery is $s = (s_a, s_b, s_c, s_d) = (0, 1/2, 1/2, 0)$.

As seen by the formal definition as well as the example above, RSD probabilities can be computed by enumerating all the $n!$ permutations over agents. Aziz et al. [2013] proved that computing the RSD lottery (and equivalently, an alternative's RSD share) is #P-hard. We are interested in an FPRAS for computing the RSD share of an alternative.

2.2 FPRAS

A *randomised approximation scheme* for a non-negative function $f : \Sigma^* \rightarrow \mathbb{R}_{\geq 0}$ takes as input an instance x , an accuracy parameter $\varepsilon \in (0, 1)$, and a failure probability $\delta \in (0, 1/2)$, and outputs a rational number z such that

$$\Pr[(1 - \varepsilon)f(x) \leq z \leq (1 + \varepsilon)f(x)] \geq 1 - \delta. \quad (1)$$

It is a *fully polynomial randomised approximation scheme*, or *FPRAS*, if its running time is polynomial in the encoding length of x , ε^{-1} , and $\log(1/\delta)$. Equivalently, one may fix a constant success probability and amplify it by independent repetition and taking the median [Jerrum et al., 1986, Lemma 6.1].

The central question we study is whether an alternative's RSD share admits an FPRAS.

The RSD-SHARE problem. The input is an explicitly represented social-choice instance $(N, A, \succsim)$ and a target alternative $a \in A$. Each weak order is encoded by its ordered indifference classes (equivalently, by a rank for every alternative); on the dichotomous subdomain, the profile may be encoded by its $n \times m$ approval matrix. The exact problem asks for the rational number s_a , represented in binary. The approximation problem additionally takes $\varepsilon \in (0, 1)$ and $\delta \in (0, 1/2)$

and asks for an output satisfying (1). Thus, throughout the paper, an FPRAS for RSD-SHARE means an FPRAS for the non-negative rational-valued function $(N, A, \succ, a) \mapsto s_a$ under this explicit input encoding.

Two agents are said to be of the same *type* if they have identical preferences. Two alternatives are of the same *type* if every agent is indifferent between them.

OBSERVATION 1 (A SINGLE SURVIVING ALTERNATIVE TYPE). *For every serial-dictatorship permutation, all alternatives in the final surviving set have the same type.*

PROOF. Let b and c both survive to the end. Because neither is ever eliminated, both are present whenever any agent i acts. Since both are retained by that agent, they both belong to the same maximal indifference class of i in the current set, and hence $b \sim_i c$. Every agent acts once, so $b \sim_i c$ for every $i \in N$. Thus, b and c have the same alternative type. $\square$

The following lemma shows that, without loss of generality, we can focus on instances in which no two alternatives have the same type.

LEMMA 1. *If the RSD-share problem admits an FPRAS on instances in which no two alternatives have the same type, then it admits an FPRAS on general instances.*

PROOF. Partition A into type classes and retain one representative of every class, with the induced preferences. Alternatives in one class are retained or eliminated together under every serial-dictatorship order. By Observation 1, the original final surviving set is one entire type class, and the reduced instance ends with its representative. If the class of a target alternative a has size d and representative a' , then this class is final in exactly the permutations for which a' is selected in the reduced instance. Conditional on that event, RSD's uniform final tie-breaking assigns probability $1/d$ to each class member. Hence $s_a = s_{a'}/d$. Dividing an FPRAS estimate for $s_{a'}$ by the exactly known integer d preserves its relative-error guarantee. $\square$

From now on, unless otherwise stated, we work with *simplified* instances in which no two alternatives have the same type. Observation 1 then implies that every final survivor set is a singleton. Consequently, for every $j \in A$, $s_j = \frac{1}{n!} \# \{ \pi \in \Pi^N : SD(N, A, \succ, \pi) = \{j\} \}$. The associated integer-valued problem of counting selecting permutations is #P-complete [Aziz et al., 2013]. On simplified instances, exact RSD-share computation is polynomial-time equivalent to this count through multiplication or division by $n!$. Deleting alternatives preserves both single-plateauedness and interval-approval preferences on the restricted common axis. Thus, the reduction above also transfers the structured-domain results below to profiles containing duplicate alternative types.

OBSERVATION 2 (THE FIRST DISTINGUISHING AGENT). *In a simplified instance, a permutation selects a if and only if, for every $b \neq a$, the first agent in the permutation who distinguishes a and b prefers a to b .*

PROOF. If a survives, every agent indifferent between a and b retains both while both are present. Their first distinguishing agent must therefore prefer a to b . Conversely, suppose a is first eliminated by agent i , who retains some $b \succ_i a$. Both alternatives survived all preceding agents, so those agents were indifferent between them. Agent i is thus their first distinguishing agent, contradicting the stated condition. $\square$

3 LIMITATIONS OF NAIVE APPROACHES

In this section, we highlight the limitations of direct permutation sampling on unrestricted profiles.

3.1 Sampling Permutations Uniformly at Random

The lottery outcome of RSD can be computed exactly by enumerating all the permutations and averaging the outcomes of serial dictatorship. The approach suggests a natural approximation heuristic: sample permutations uniformly at random and then take the average of the outcomes of applying serial dictatorship with respect to the permutation. However, the naive randomised algorithm of drawing permutations uniformly at random performs poorly especially if there are very few permutations under which the alternative is chosen.

Next, we show that some alternatives may have a very low probability of being selected by the RSD mechanism, and therefore naively sampling permutations may result in attributing zero probability to the alternative with high probability. This implies that for this procedure, no multiplicative approximation guarantee can be provided. The difference between a zero RSD share and some small ε RSD share may determine the difference in deciding whether an alternative gets any representation (especially in a scenario where the RSD shares are used to make proportional selections in a discrete selection problem, such as apportioning the number of representatives of a party in a committee). In a participatory budgeting setting, a project that deserves \$100k based on its RSD share may instead get a zero share if the RSD shares are not estimated well.

Example 2. Consider the following instance with agent set $[n]$ and alternatives $a_1, \dots, a_n$. The agents' preferences are defined as follows. Here, for preferences $\succsim$, we use the notation $I \succ I'$ to denote that I and I' are indifference classes under $\succsim$ and that any alternative in I is strictly more desirable than any alternative in I' under $\succsim$.

$$\begin{array}{ll}
 1: A \setminus \{a_1\} \succ_1 a_1 & i: a_{i-1} \succ_i A \setminus \{a_{i-1}, a_i\} \succ_i a_i \\
 2: a_1 \succ_2 A \setminus \{a_1, a_2\} \succ_2 a_2 & \vdots \\
 3: a_2 \succ_3 A \setminus \{a_2, a_3\} \succ_3 a_3 & n-1: a_{n-2} \succ_{n-1} A \setminus \{a_{n-2}, a_{n-1}\} \succ_{n-1} a_{n-1} \\
 \vdots & n: a_{n-1} \succ_n a_n \succ_n A \setminus \{a_{n-1}, a_n\}
 \end{array}$$

This example is a general weak-preference illustration, not a dichotomous instance: agents $2, \dots, n$ have three indifference classes. The dichotomous hardness construction is given in Theorem 1.

The only permutation under which a_n is selected is $(1, 2, 3, \dots, n)$. Hence, it is possible that uniformly sampling permutations may miss the permutation that results in the selection of a_n .

We can extend this example to settings where the number of permutations where serial dictatorship yields a specific alternative a_n equals other small numbers. Consider introducing a series of x copies of agent n to the above instance. Now, there are $(n+x)!$ permutations, and serial dictatorship will result in alternative a_n only for those permutations that have $(1, \dots, n-1)$ as a prefix. Thus, in this modified instance there are $(x+1)!$ permutations on which serial dictatorship yields a_n , so that the RSD share of a_n would be $(x+1)!/(n+x)!$ which is inverse exponential in n for any fixed x .

3.2 Sampling Alternative-Oriented Permutations

The above drawback of the naive sampling procedure can be attempted to be fixed via a natural permutation selection process outlined as Algorithm 2 (see Appendix). Algorithm 2 constructs a permutation iteratively. Starting with $N' = N$ and $A' = A$, it chooses uniformly at random a remaining agent i for whom $a \in \max_{\succsim_i}(A')$, appends i to the permutation, and replaces A' by $\max_{\succsim_i}(A')$. If no such agent exists, it reports that no permutation selects a . Otherwise, it eventually returns an a -selecting permutation. Although every selecting permutation is generated with positive probability, these probabilities need not be equal, so the procedure is not a uniform sampler. The

process is an adaptation of the original algorithm of Aziz et al. [2013] to check whether an alternative is in the support of the RSD lottery or not.

It is well-known that for so-called *self-reducible* problems, approximate counting reduces to developing an *FPAUS*, i.e., a *fully polynomial almost uniform sampler*, for the problem. In our case, this means that we should have an algorithm that samples uniformly at random, in polynomial time, an agent permutation from the set of permutations on which serial dictatorship would select a target alternative a . As it is straightforward to see that Algorithm 2 selects each such permutation with positive probability, and has a seemingly natural formulation, one may wonder whether Algorithm 2 would indeed be the exact or approximate sampler that we are looking for. Unfortunately, the following example shows that the algorithm does not sample permutations uniformly at random.

Example 3. Consider a social choice instance with agents $\{1, 2, 3\}$ and alternatives $\{a, b, c\}$. The preferences are as follows.

$$1: \{a, c\} \succ_1 b \qquad 2: \{a, b\} \succ_2 c \qquad 3: b \succ_3 a \succ_3 c$$

There are three permutations that yield alternative a as the outcome: $(1, 2, 3)$, $(1, 3, 2)$, and $(2, 1, 3)$. The third permutation $(2, 1, 3)$ is sampled with probability $1/2$ with the algorithm, although it should be sampled with probability $1/3$.

4 INAPPROXIMABILITY OF RSD

For a dichotomous preference relation, let $\text{App}(i)$ denote the approval set of agent i , and put $B_t(\pi) = S_{t-1}(\pi) \cap \text{App}(\pi(t))$. The above update rule becomes

$$S_t(\pi) = \begin{cases} B_t(\pi), & \text{if } B_t(\pi) \neq \emptyset, \\ S_{t-1}(\pi), & \text{otherwise.} \end{cases}$$

Thus, if none of the currently surviving alternatives is approved by the acting agent, all surviving alternatives are tied in the agent's lower indifference class and the surviving set remains unchanged.

LEMMA 2 (SELECTING PERMUTATIONS FROM A GRAPH). *Let $G = (V, E)$ be a simple graph with $|V| = N$, and let $q \geq 1$ be an integer. Introduce a vertex agent x_v for each $v \in V$, test agents $y_1, \dots, y_q$, and alternatives $A = \{a\} \cup \{b_e : e \in E\}$, with*

$$\text{App}(x_v) = \{a\} \cup \{b_e : v \notin e\}, \qquad \text{App}(y_j) = \{b_e : e \in E\}.$$

For a permutation π , let $I(\pi)$ consist of the vertices whose agents occur after the first test agent. Then a is the unique final survivor if and only if $I(\pi)$ is independent; otherwise, a is eliminated. For each fixed independent set I of size s , exactly

$$w_s = (N - s)! q (q + s - 1)!$$

permutations satisfy $I(\pi) = I$. Consequently, the number of permutations selecting a is $Z(G) = \sum_{I \in \mathcal{I}(G)} w_{|I|}$, where $\mathcal{I}(G)$ is the collection of independent sets of G .

PROOF. Write $C = V \setminus I(\pi)$. Immediately before the first test agent acts, the surviving set is

$$\{a\} \cup \{b_e : e \cap C = \emptyset\}.$$

If $I(\pi)$ is independent, then C is a vertex cover, so this set is $\{a\}$ and no subsequent agent changes it. Otherwise, a surviving edge alternative is approved by the first test agent, who eliminates a .

For a fixed I of size s , order the $N - s$ agents outside I , choose the first test agent in q ways, and order the remaining $q + s - 1$ agents. These choices give the stated count and partition all selecting permutations. $\square$

THEOREM 1. *Unless $\text{NP} \subseteq \text{BPP}$, there is no fully polynomial randomised approximation scheme for computing the RSD share of a given alternative, even when all preferences are dichotomous.*

The result continues to hold under the following restrictions:

- (i) *the number of alternatives is linear in the number of agents;*
- (ii) *all agents who disapprove the target alternative have identical preferences; and*
- (iii) *for every permutation, the target alternative is either the unique serial-dictatorship outcome or is not contained in the outcome.*

PROOF. We reduce from a constant-gap version of MAXIMUM INDEPENDENT SET on graphs of maximum degree three. By Berman and Karpinski [1999, Theorem 3], for $\xi = 1/8$ it is NP-hard, given such a graph with $N = 284t$ vertices, to distinguish independence number greater than $(140 - \xi)t$ from independence number smaller than $(139 + \xi)t$. Since $140 - \xi > 559/4$ and $139 + \xi < 557/4$, it is therefore NP-hard to distinguish

$$\alpha(G) \geq \gamma_1 N \quad (\text{YES}) \quad \text{from} \quad \alpha(G) \leq \gamma_0 N \quad (\text{NO}),$$

where $\gamma_1 = 559/1136$ and $\gamma_0 = 557/1136$ are fixed constants. Set $\Delta = \gamma_1 - \gamma_0 > 0$, choose a fixed integer $C > 1$ with $C^\Delta > 2$, and put $q = CN$.

Construction and counting. Apply Lemma 2 to G with q test agents. The resulting instance has $n = N + q = (C + 1)N$ agents and $m = 1 + |E| \leq 1 + 3N/2 = O(n)$ alternatives, so its explicit approval matrix has polynomial size. All agents disapproving a are identical test agents, and the lemma also establishes the stated restriction on final survivors. The share of a is $p_a = Z(G)/(N + q)!$, with

$$Z(G) = \sum_{I \in \mathcal{I}(G)} w_{|I|}, \quad w_s = (N - s)! q (q + s - 1)!, \quad \frac{w_{s+1}}{w_s} = \frac{q + s}{N - s} \geq C$$

for $0 \leq s < N$. Thus the weights increase by at least a factor of C at each step.

Separating the gap cases. Let $s_1 = \lceil \gamma_1 N \rceil$ and $s_0 = \lfloor \gamma_0 N \rfloor$. A YES instance contains an independent set of size at least s_1 , whereas a NO instance has at most 2^N independent sets, all of size at most s_0 . Hence

$$\text{YES: } Z(G) \geq w_{s_1}, \quad \text{NO: } Z(G) \leq 2^N w_{s_0}.$$

Since $s_1 - s_0 \geq \Delta N$,

$$\frac{w_{s_1}}{2^N w_{s_0}} \geq \frac{C^{s_1 - s_0}}{2^N} \geq \left(\frac{C^\Delta}{2} \right)^N > 4$$

for all sufficiently large N . The finitely many smaller input sizes can be decided by brute force. Define

$$U = \frac{2^N w_{s_0}}{(N + q)!}.$$

We have $p_a \leq U$ in the NO case and $p_a > 4U$ in the YES case. An FPRAS run with $\varepsilon = 1/4$ and $\delta = 1/4$ would therefore, with probability at least $3/4$, return an estimate satisfying

$$\text{NO: } \hat{p}_a \leq \frac{5}{4}U < 2U, \quad \text{YES: } \hat{p}_a \geq \frac{3}{4}p_a > 3U > 2U.$$

Comparing the estimate with $2U$ distinguishes the two cases. This comparison is polynomial-time: because $q = CN$, the factorials and integers defining U have $O(N \log N)$ bits, and rational comparison uses exact cross-multiplication. We would obtain a bounded-error randomized polynomial-time algorithm for an NP-hard promise problem, implying $\text{NP} \subseteq \text{BPP}$. $\square$

Remark 1 (The amplification constant and a second parameterisation). Here

$$\Delta = \gamma_1 - \gamma_0 = \frac{2}{1136} = \frac{1}{568}.$$

The stronger sufficient condition $C^\Delta > 4$ would force

$$C > 4^{568} = 9.334 \dots \times 10^{341}.$$

The proof only needs $C^\Delta > 2$, because then $(C^\Delta/2)^N > 4$ for all sufficiently large N ; this is the condition used above. Even so, it requires

$$C > 2^{568} = 9.661 \dots \times 10^{170}.$$

Thus, C is a fixed but extremely large constant. This does not invalidate the polynomial-time reduction— C is independent of the input—but it makes the hidden constant in the linear agent blow-up enormous.

The large multiplier is the price of keeping the number of constructed agents linear in the source graph size and, on bounded-degree gap families with linearly many edges, keeping the numbers of agents and alternatives on the same proportional scale. A second, simpler parameterisation sets $q = N^2$. Then

$$\frac{w_{s+1}}{w_s} = \frac{q+s}{N-s} \geq N,$$

and hence

$$\frac{w_{s_1}}{w_{s_0}} \geq N^{\Delta N} > 4 \cdot 2^N$$

for all sufficiently large N . The proof then goes through without any large fixed multiplier. This version has $n = N + N^2 = \Theta(N^2)$ agents and $m = 1 + |E| = O(N) = O(\sqrt{n})$ alternatives; on the usual bounded-degree gap families with $|E| = \Theta(N)$, this is $m = \Theta(\sqrt{n})$. Thus, if restriction (i) is read as the upper bound $m = O(n)$, the second parameterisation still satisfies it and in fact strengthens it to $m = O(\sqrt{n})$; what it gives up is the proportional-scale relation $m = \Theta(n)$. Its explicit approval-matrix representation remains polynomial in the source-instance size.

Finally, replacing the crude upper bound 2^N by a sharp count of subsets of size at most $\gamma_0 N$ does not remove the phenomenon: the binary-entropy exponent is $H_2(\gamma_0) \approx 0.99973$. It merely changes the asymptotic threshold from $C^\Delta > 2$ to $C^\Delta > 2^{H_2(\gamma_0)}$, so the exponent of the required constant falls only from 568 to about 567.85. Thus, within this weight-separation argument, a base exponential in $1/\Delta$ remains unavoidable.

COROLLARY 1. *Unless $\text{NP} \subseteq \text{BPP}$, there is no FPRAS for computing RSD shares on the unrestricted domain of weak preferences.*

PROOF. Dichotomous preferences form a subdomain of the domain of weak preferences. An FPRAS for the unrestricted problem would therefore also be an FPRAS for the dichotomous restriction, contradicting the theorem. $\square$

Remark 2. The impossibility result relies essentially on indifferences. Under strict preferences, serial dictatorship selects the top-ranked alternative of the first agent in the permutation. Hence, RSD coincides with random dictatorship, and the RSD share of an alternative can be computed directly from the fraction of agents who rank it first.

5 USING MARKOV CHAINS

This section gives a general Markov-chain framework for approximating an RSD share. The framework is conditional: it becomes an FPRAS on a domain once we can sample almost uniformly, in polynomial time, from every state space obtained by fixing a suffix of a selecting permutation. We first recall the Markov-chain notions needed later and then state the telescoping estimator.

5.1 Markov-chain preliminaries

Let $\mathcal{M}$ be a finite Markov chain with state space Ω , transition matrix P , and stationary distribution τ . The chain is *irreducible* if every state can be reached from every other state with positive probability, and it is *aperiodic* if the greatest common divisor of all return times is one. A finite irreducible and aperiodic chain is ergodic and converges to a unique stationary distribution.

For distributions μ and ν on Ω , their total-variation distance is $d_{TV}(\mu, \nu) = \frac{1}{2} \sum_{x \in \Omega} |\mu(x) - \nu(x)|$. The η -mixing time is $t_{\text{mix}}(\eta) = \max_{x \in \Omega} \min\{t : d_{TV}(P^t(x, \cdot), \tau) \leq \eta\}$. The chain is reversible if $\tau(x)P(x, y) = \tau(y)P(y, x)$ for all $x, y \in \Omega$.

For a reversible chain, write $Q(x, y) = \tau(x)P(x, y)$ and, for $S \subseteq \Omega$, $Q(S, \bar{S}) = \sum_{x \in S, y \notin S} Q(x, y)$. The conductance is $\Phi = \min_{0 < \tau(S) \leq 1/2} \frac{Q(S, \bar{S})}{\tau(S)}$. For lazy reversible chains, we use canonical-path congestion to bound mixing from above and conductance to bound mixing from below. In particular, every finite irreducible lazy reversible chain satisfies $t_{\text{mix}}(1/4) \geq \frac{1}{4\Phi}$; see, for example, Sinclair [1992] and Levin et al. [2009].

For completeness, consider a finite irreducible lazy reversible chain and suppose that for every ordered pair (x, y) we specify a path $\gamma_{x,y}$ in the transition graph. Its congestion is $\rho = \max_e \frac{1}{Q(e)} \sum_{\gamma_{x,y} \ni e} \tau(x)\tau(y)$. If every path has length at most L , then $t_{\text{mix}}(\eta) \leq \rho L \log\left(\frac{1}{\eta \tau_{\min}}\right)$, where $\tau_{\min} = \min_{x \in \Omega} \tau(x)$; see Sinclair [1992].

5.2 Suffix-restricted selecting permutations

Fix a target alternative a . Under the simplifying convention from Section 2 that no two alternatives have the same type, every serial-dictatorship outcome is a singleton. Define

$$\Omega = \{\pi \in \Pi^N : SD(\pi) = \{a\}\}.$$

Then the RSD share of a is $s_a = \frac{|\Omega|}{n!}$. If $\Omega = \emptyset$, the support algorithm of Aziz et al. [2013] detects this in polynomial time and the correct output is zero. Otherwise, the same constructive procedure gives an initial permutation in Ω .

The counting algorithm fixes a suffix from right to left. At an iteration with k unfixed agents, let R_k be their set and let σ_k be the already fixed suffix. We write

$$\Omega_k = \{\rho\sigma_k : \rho \in \Pi_{R_k}, SD(\rho\sigma_k) = \{a\}\}.$$

The algorithm always stores a state $\pi_k^0 \in \Omega_k$. For every $u \in R_k$, define

$$p_{k,u} = \frac{|\{\pi \in \Omega_k : \pi(k) = u\}|}{|\Omega_k|}.$$

These probabilities sum to one, so $\max_{u \in R_k} p_{k,u} \geq \frac{1}{k}$. When $k = 1$, the state space is a singleton and sampling is trivial; all Markov-chain arguments below concern $k \geq 2$.

5.3 The telescoping estimator

The state space chosen for the next iteration is selected from the same sample used to estimate its size. Therefore, the positional probabilities must be estimated simultaneously for all remaining agents; treating the empirically most frequent agent as fixed in advance would be invalid.

Algorithm 1 Markov-chain estimator for the RSD share of a

Require: Instance $(N, A, \succ)$, target a , accuracy $\epsilon \in (0, 1/2)$, failure probability $\delta \in (0, 1/2)$, and a uniform-mixing chain for every nonempty suffix-restricted state space.

- 1: Use the support algorithm to test whether $\Omega = \emptyset$.
- 2: **if** $\Omega = \emptyset$ **then**
- 3: **return** 0.
- 4: **end if**
- 5: Find $\pi_0^n \in \Omega_n = \Omega$.
- 6: Set $\alpha = \epsilon/(12n^2)$, $\eta = \alpha/2$, and $s = \left\lceil \frac{2}{\alpha^2} \log \left(\frac{2n^2}{\delta} \right) \right\rceil$.
- 7: Set $\widehat{Z} = 1$.
- 8: **for** $k = n, n-1, \dots, 1$ **do**
- 9: Independently generate s samples $\pi^1, \dots, \pi^s$ whose distributions are within total-variation distance η of the uniform distribution on Ω_k .
- 10: **for each** $u \in R_k$ **do**
- 11: $\widehat{p}_{k,u} \leftarrow s^{-1} \sum_{\ell=1}^s \mathbf{1}[\pi^\ell(k) = u]$.
- 12: **end for**
- 13: Choose $u_k \in \arg \max_{u \in R_k} \widehat{p}_{k,u}$ and set $r_k = \widehat{p}_{k,u_k}$.
- 14: Set $\widehat{Z} \leftarrow \widehat{Z}/r_k$.
- 15: Fix u_k in position k , thereby defining $\Omega_{k-1} = \{\pi \in \Omega_k : \pi(k) = u_k\}$.
- 16: Choose, as π_0^{k-1} , any sampled permutation with $\pi^\ell(k) = u_k$.
- 17: **end for**
- 18: **return** $\widehat{Z}/n!$.

THEOREM 2. Suppose that, for every nonempty suffix-restricted state space Ω_k , an η -approximately uniform sample can be generated in time polynomial in the input size, n , and $\log(1/\eta)$. Then Algorithm 1 is an FPRAS for the RSD share of a .

PROOF. We expose the random choices iteration by iteration. Condition on the complete history before iteration k ; this fixes the random state space Ω_k and its remaining-agent set R_k . For a sampled permutation π^ℓ , let $q_{\ell,u} = \Pr[\pi^\ell(k) = u]$. Total-variation distance at most $\eta = \alpha/2$ gives

$$|q_{\ell,u} - p_{k,u}| \leq \eta.$$

The samples are independent. Hoeffding's inequality therefore yields

$$\begin{aligned} \Pr \left[\left| \widehat{p}_{k,u} - \frac{1}{s} \sum_{\ell=1}^s q_{\ell,u} \right| > \frac{\alpha}{2} \mid \text{the preceding history} \right] \\ \leq 2 \exp \left(-\frac{s\alpha^2}{2} \right) \leq \frac{\delta}{n^2}. \end{aligned}$$

Combining this with the total-variation bias shows that $|\widehat{p}_{k,u} - p_{k,u}| \leq \alpha$ except with conditional probability at most δ/n^2 . This bound is uniform over every possible preceding history. A union bound over the at most n^2 iteration-agent pairs therefore shows that, with probability at least $1 - \delta$, all these inequalities hold simultaneously, despite the adaptive choice of the state spaces.

The empirical frequencies sum to one, so their maximum satisfies $r_k \geq 1/k$ deterministically. Condition on the simultaneous accuracy event above. For the selected agent u_k ,

$$\left| \frac{p_{k,u_k}}{r_k} - 1 \right| \leq \frac{\alpha}{r_k} \leq k\alpha \leq \frac{\epsilon}{12n} =: \theta.$$

Also,

$$|\Omega_{k-1}| = p_{k,u_k} |\Omega_k|.$$

Because $|\Omega_0| = 1$, telescoping gives

$$|\Omega_n| = \prod_{k=1}^n p_{k,u_k}^{-1}, \quad \widehat{Z} = \prod_{k=1}^n r_k^{-1}.$$

Consequently,

$$\frac{\widehat{Z}}{|\Omega_n|} = \prod_{k=1}^n \frac{p_{k,u_k}}{r_k} \in [(1-\theta)^n, (1+\theta)^n].$$

Bernoulli's inequality gives $(1-\theta)^n \geq 1-n\theta \geq 1-\epsilon$. Also, $(1+\theta)^n \leq e^{n\theta} \leq e^{\epsilon/12} \leq 1+\epsilon$ for $\epsilon \in (0, 1/2)$. Thus

$$(1-\epsilon)|\Omega_n| \leq \widehat{Z} \leq (1+\epsilon)|\Omega_n|.$$

Division by $n!$ preserves this guarantee. The support test, the number of samples, the length of each run, and direct serial-dictatorship membership tests are all polynomial in the input size, ϵ^{-1} , and $\log(1/\delta)$ under the sampling assumption. Hence the algorithm is an FPRAS. $\square$

6 THE INSERTION MARKOV CHAIN

We now study a natural chain for sampling selecting permutations. An insertion removes one agent from its current position and inserts it in another position, shifting the intervening agents by one place.

For $k \geq 2$ and $i, j \in [k]$ with $i \neq j$, let $v(i, j)$ move the agent currently in position j to position i . On a nonempty suffix-restricted state space Ω_k , the lazy Insertion Markov Chain acts as follows. With probability $1/2$ it stays at the current state. Otherwise, it chooses an ordered pair (i, j) uniformly from the $k(k-1)$ pairs with $i \neq j$ and performs the insertion if the resulting permutation remains in Ω_k ; if not, it stays put. For $k = 1$, the chain consists of its unique state.

The following monotonicity observation is useful throughout the remainder of the paper. For a sequence ρ of agents and a set S of alternatives, write $SD_S(\rho)$ for the surviving set obtained by starting serial dictatorship from S rather than from all alternatives.

For each agent i , let $E_i(a) = \{b \in A : b \sim_i a\}$ be the indifference class containing a . Whenever i retains a from a set S , the update is $S \cap E_i(a)$.

LEMMA 3 (SURVIVAL UNDER RESTRICTION). *Let $T \subseteq S$ and suppose $a \in T$. If $a \in SD_S(\rho)$, then*

$$a \in SD_T(\rho) \quad \text{and} \quad SD_T(\rho) = T \cap SD_S(\rho).$$

PROOF. Since a survives from S , every acting agent i intersects the current set with $E_i(a)$. Starting from $T \subseteq S$ containing a applies the same intersections: inductively, a remains maximal in the smaller set whenever it is maximal in the larger one. Thus a also survives from T , and the resulting set is $T \cap SD_S(\rho)$. $\square$

LEMMA 4 (LEFTMOST INSERTION CLOSURE). *Let $\pi, \pi' \in \Omega_k$. Let i be their first position of disagreement and let $j > i$ satisfy $\pi(j) = \pi'(i)$. Then $v(i, j) \circ \pi \in \Omega_k$.*

The proof is given in Appendix B.

LEMMA 5. *On every nonempty suffix-restricted state space Ω_k , the lazy Insertion Markov Chain is irreducible, aperiodic, and reversible with respect to the uniform distribution.*

The proof is given in Appendix C.

6.1 An exponential mixing-time lower bound

Although the chain has the desired stationary distribution and is connected, it need not mix rapidly. Unlike Theorem 1, the next result is unconditional and specific to this chain: it does not rule out other approximation algorithms, but locates the obstruction in an explicit geometric conductance bottleneck.

THEOREM 3. *For every $r \geq 2$, there is a dichotomous social-choice instance with $n = 4r$ agents for which the lazy Insertion Markov Chain has conductance*

$$\Phi \leq \frac{1}{2^r - 1}.$$

Consequently,

$$t_{\text{mix}}(1/4) \geq \frac{2^r - 1}{4} = 2^{\Omega(n)}.$$

In particular, the Insertion Markov Chain is not rapidly mixing in general.

The proof is given in Appendix E.

7 FPRAS FOR SPECIAL CASES

The general hardness result and Theorem 3 leave open efficient approximation on structured domains. We first consider state spaces described by precedence constraints. We then give a general sufficient condition for rapid mixing of the Insertion Markov Chain.

7.1 Selecting permutations represented by a partial order

Definition 2 (Poset-representable target). Fix a preference profile and a target alternative a . The instance is *poset-representable for a* if there is a partial order $P_a = (N, \prec_a)$, computable in polynomial time, such that for every $\pi \in \Pi^N$, $SD(\pi) = \{a\} \iff \pi \in \mathcal{L}(P_a)$, where $\mathcal{L}(P_a)$ is the set of linear extensions of P_a .

THEOREM 4. *On instances that are poset-representable for the target alternative a , Algorithm 1, instantiated with a rapidly mixing linear-extension sampler, is an FPRAS for the RSD share of a .*

PROOF. Let $P = P_a$. Then $s_a = \frac{|\mathcal{L}(P)|}{n!}$. Suppose that Algorithm 1 has fixed a suffix $\sigma = (i_{k+1}, \dots, i_n)$ occurring in a linear extension. Let $N_k = N \setminus \{i_{k+1}, \dots, i_n\}$ and let P_k be the subposet induced by N_k . The permutations in the corresponding state space are precisely $\{(\rho, i_{k+1}, \dots, i_n) : \rho \in \mathcal{L}(P_k)\}$. Indeed, all constraints within N_k must be respected, and no member of the fixed suffix can be required to precede a remaining element because the suffix occurs in a linear extension.

Thus every suffix-restricted state space is in bijection with the linear extensions of an induced subposet. The rapidly mixing chain of Bubley and Dyer [1999] therefore supplies the approximately uniform samples required by Theorem 2. All order and transition tests are polynomial-time, so the result follows. $\square$

PROPOSITION 1 (HEIGHT-TWO DICHOTOMOUS PREFERENCES). *Let $N = L \dot{\cup} U$ and let $E \subseteq L \times U$. There is a dichotomous preference profile with target alternative a such that $SD(\pi) = \{a\}$ if and only if ℓ occurs before u for every $(\ell, u) \in E$. Consequently, exact RSD-share computation is #P-hard on this domain, while the RSD share of a admits an FPRAS.*

PROOF. Introduce an alternative $b_{\ell u}$ for every $(\ell, u) \in E$ and set $A = \{a\} \cup \{b_{\ell u} : (\ell, u) \in E\}$. Define

$$\text{App}(\ell) = \{a\} \cup \{b_{xy} : x \neq \ell\}, \quad \text{App}(u) = \{b_{\ell u} : (\ell, u) \in E\}.$$

For each blocker $b_{\ell u}$, only ℓ and u distinguish it from a : agent ℓ prefers a , whereas agent u prefers the blocker. By Observation 2, a is selected if and only if ℓ precedes u for every $(\ell, u) \in E$. Hence the selecting permutations are exactly the linear extensions of the height-two partial order P_E with covering relations E , and

$$n! s_a = |\mathcal{L}(P_E)|.$$

The construction is polynomial-time. Counting linear extensions is #P-complete even for height-two posets [Dittmer and Pak, 2020], which proves the exact-computation claim. This height-two result resolved a conjecture left open by Brightwell and Winkler [1991], whose original theorem concerned unrestricted posets. Theorem 4 supplies an FPRAS. $\square$

7.2 Prefix compressibility

For a set R of remaining agents and a fixed suffix σ of agents outside R , define

$$\Omega(R, \sigma) = \{\rho\sigma : \rho \in \Pi_R, SD(\rho\sigma) = \{a\}\}.$$

Definition 3 (Prefix compressibility). Fix a constant $c \geq 0$. A preference domain is *c-prefix-compressible* for a if, for every R , every fixed suffix σ , and every nonempty $\Omega(R, \sigma)$, there is an ordered tuple $B = (i_1, \dots, i_b)$, $b \leq c$, of distinct agents from R such that $SD(B\tau\sigma) = \{a\}$ for every ordering τ of $R \setminus B$.

The tuple B certifies that $|\Omega(R, \sigma)| \geq (|R| - b)!$.

We next prove that, on every c -prefix-compressible domain for fixed c , the Insertion Markov Chain mixes rapidly on every nonempty suffix-restricted space. This provides the approximately uniform samples required by Algorithm 1.

LEMMA 6 (CANONICAL-PATH COUNT). *Let Ω_k be a nonempty suffix-restricted state space on k remaining agents. For each ordered pair $(\pi, \pi') \in \Omega_k^2$, form a path by repeatedly moving the agent required by π' into the first position at which the current permutation and π' differ. These paths remain in Ω_k , have length at most $k - 1$, and for every directed transition e at most $k!$ ordered pairs have a path using e .*

PROOF. Path validity follows from Lemma 4, and each insertion increases the common-prefix length, giving at most $k - 1$ moves.

Fix a directed transition $e = (\gamma, \gamma')$, and let i be the first position where its endpoints differ. For any path using e , the first $i - 1$ destination agents have already been fixed. All other agents occur in the source permutation in the order $\gamma(i), \dots, \gamma(k)$: earlier moves only extracted agents into the fixed prefix. The source is therefore determined by the positions and ordering of those first $i - 1$ destination agents, giving at most $\binom{k}{i-1}(i-1)!$ possibilities. The destination's first i entries are fixed by γ' , leaving at most $(k - i)!$ possibilities for its other entries. Thus

$$|\text{cp}(e)| \leq \binom{k}{i-1}(i-1)!(k-i)! = \frac{k!}{k-i+1} \leq k!,$$

where $\text{cp}(e)$ denotes the ordered endpoint pairs whose canonical path uses e . The suffix outside these k positions is fixed throughout. $\square$

THEOREM 5. *For every fixed constant c , Algorithm 1 using the Insertion Markov Chain is an FPRAS on every c -prefix-compressible preference domain.*

The proof is given in Appendix F.

7.3 Single-plateaued preferences

Let $x \succ y$ abbreviate $y \triangleleft x$. Let the alternatives be arranged on a common linear axis $\triangleleft$. For each agent i , let $P_i = [\ell_i, r_i]$ be a nonempty interval of most-preferred alternatives. The agent is indifferent within P_i . On the left side, $x \triangleleft y \triangleleft \ell_i$ implies $y \succ_i x$; on the right side, $r_i \triangleleft x \triangleleft y$ implies $x \succ_i y$. Subject to completeness and transitivity, no additional restriction is imposed on comparisons between an alternative to the left of the plateau and one to its right. The results below are insensitive to these cross-side comparisons: whenever a current interval S is disjoint from P_i , the interval S lies entirely on one side of P_i , so only the within-side monotonicity is used. We call such preferences *single-plateaued*.

LEMMA 7. *Under single-plateaued preferences on a common axis, if a survives a sequence of agents, then the surviving set is an interval containing a .*

PROOF. The initial set is the whole axis. The set of most-preferred alternatives of a single-plateaued agent within an interval is itself an interval. If the update retains a , the new interval contains a . The result follows inductively. $\square$

THEOREM 6. *Single-plateaued preferences on a common axis are 2-prefix-compressible. Consequently, the RSD share of a target alternative admits an FPRAS on this domain.*

PROOF. Fix R and σ with $\Omega(R, \sigma) \neq \emptyset$, and let ρ be a successful ordering of R .

If every agent in R has a in their plateau, then, while a survives, agent i maps a current interval S to $S \cap P_i$. These intersections commute. Hence the surviving set immediately before σ is independent of the order of R , and the empty tuple witnesses prefix compressibility.

Otherwise, let y be the first agent in ρ whose plateau does not contain a . All preceding agents have plateaus containing a , so immediately before y acts the surviving set is $S = \bigcap_{x \text{ preceding } y} P_x$, an interval containing a . Suppose P_y lies strictly to the left of a . Since the ordering is successful, S cannot intersect P_y ; otherwise y would retain only alternatives strictly to the left of a . Hence y 's most-preferred alternative in S is its left endpoint. For a to survive, that endpoint must be a , and y reduces S to $\{a\}$. There must be at least one preceding agent: if y were first, then the whole axis would still contain alternatives in P_y strictly to the left of a , so y could not retain a . The left endpoint of the finite intersection S is the maximum of the preceding plateau left endpoints, so some preceding agent x has plateau left endpoint a . Placing x first and y second again makes y select a uniquely from P_x . The surviving set becomes $\{a\}$, after which the remaining agents and the fixed suffix cannot change it.

The case in which P_y lies strictly to the right of a is symmetric, using a preceding plateau whose right endpoint is a . Thus, an ordered tuple of at most two agents works in every nonempty restricted state space. Theorem 5 supplies the insertion-chain FPRAS. $\square$

7.4 Dichotomous single-peaked preferences

Suppose each agent has a dichotomous preference whose approval set is an interval $I_i = [\ell_i, r_i]$ on a common axis. Fix a target alternative a and let $X = \{i : a \in I_i\}$ be its approvers.

THEOREM 7. *Dichotomous single-peaked, or interval-approval, preferences on a common axis are 2-prefix-compressible. Consequently, the RSD share of a target alternative admits an FPRAS on this domain.*

PROOF. Fix a nonempty $\Omega(U, \sigma)$, and let $X_U = U \cap X$. Set

$$J = \bigcap_{i \in X_U} I_i,$$

interpreting the empty intersection as the whole alternative set. When $X_U \neq \emptyset$, every interval in this intersection contains a . Choose an approver with maximum left endpoint and one with minimum right endpoint. Their intervals already intersect in J . Place these agents first, using one agent if they coincide; when $X_U = \emptyset$, use the empty tuple. The surviving set after this prefix is J .

In any successful ordering of U , every disapprover leaves the current set unchanged, since a nonempty approval intersection would eliminate a . Every approver intersects it with an interval containing a . Consequently, the set at every such disapprover's turn contains J , and the set immediately before σ is exactly J . Each disapprover's approval interval is therefore disjoint from J .

After the chosen prefix, every remaining approver contains J and every remaining disapprover has approval interval disjoint from J . All remaining agents thus leave J unchanged in any order. The suffix σ starts from the same set as in the original successful ordering and still selects a . This proves 2-prefix compressibility, so Theorem 5 applies. $\square$

8 CONCLUSIONS

Table 1. Summary of the computational results.

Preference restriction or state-space structure	Computational conclusion
Strict preferences	Exact polynomial-time computation; RSD is random dictatorship.
Poset-representable selecting permutations	An FPRAS follows from rapidly mixing linear-extension samplers.
Single-plateaued preferences on a common axis	The Insertion Markov Chain mixes rapidly and gives an FPRAS via Algorithm 1.
Dichotomous single-peaked preferences	The Insertion Markov Chain mixes rapidly and gives an FPRAS via Algorithm 1.
Arbitrary dichotomous preferences	No FPRAS exists unless $\text{NP} \subseteq \text{BPP}$.

We studied the approximability of Random Serial Dictatorship shares. Our main complexity result closes the long-standing general approximation question: unless $\text{NP} \subseteq \text{BPP}$, there is no FPRAS, even on a highly restricted dichotomous domain. Thus, exact #P-hardness is accompanied by a strong barrier to multiplicative approximation.

The negative result does not eliminate the usefulness of Markov-chain methods. The telescoping framework isolates the algorithmic requirement: efficient approximate sampling from every suffix-restricted set of selecting permutations. The natural Insertion Markov Chain is connected and uniform on these spaces, but it can have exponentially small conductance. The same chain mixes rapidly whenever the domain is c -prefix-compressible for a fixed constant c , including single-plateaued preferences and dichotomous single-peaked preferences on a common axis. It therefore supplies approximately uniform selecting permutations on all suffix-restricted spaces and an FPRAS through the telescoping framework. Poset-representable selecting permutations provide a further tractable class through rapidly mixing samplers for linear extensions. Table 1 summarises the computational conclusions.

ACKNOWLEDGMENTS

The authors thank Catherine Greenhill for giving them pointers on the literature on rapidly mixing Markov chains. Sebastian Homrighausen was supported by Aarhus University for an extended research visit in 2024-2025.

Disclosure of AI assistance. This work was conceptualized, and its core technical content developed and written, by the authors prior to and independently of any use of large language models. In particular, the analysis of the limitations of existing approaches, the main hardness reduction, and the central Markov chain formulation and argument were devised and written by the authors. At a later stage of the project, the authors used OpenAI’s ChatGPT (GPT-5.6) to shortlist candidate tractable cases where our proposed Markov chain works well, to streamline the presentation of the inapproximability proof, and to audit proof details and examples.

REFERENCES

- S. Airiau, H. Aziz, I. Caragiannis, J. Kruger, J. Lang, and Dominik Peters. 2019. Portioning using Ordinal Preferences: Fairness and Efficiency. In *Proceedings of the 28th International Joint Conference on Artificial Intelligence (IJCAI)*. IJCAI.
- H. Aziz, A. Bogomolnaia, and H. Moulin. 2020. Fair Mixing: the Case of Dichotomous Preferences. *ACM Transactions on Economics and Computation (TEAC)* 8, 4 (2020), 18:1–18:27.
- H. Aziz, F. Brandl, F. Brandt, and M. Brill. 2018. On the tradeoff between efficiency and strategyproofness. *Games and Economic Behavior* 110 (2018), 1–18.
- H. Aziz, F. Brandt, and M. Brill. 2013. The Computational Complexity of Random Serial Dictatorship. *Economics Letters* 121, 3 (2013), 341–345.
- H. Aziz, F. Brandt, M. Brill, and J. Mestre. 2014. Computational Aspects of Random Serial Dictatorship. *ACM SIGecom Exchanges* 13, 2 (2014), 26–30.
- H. Aziz and P. Stursberg. 2014. A Generalization of Probabilistic Serial to Randomized Social Choice. In *Proceedings of the 28th AAAI Conference on Artificial Intelligence (AAAI)*. AAAI Press, 559–565.
- S. Bade. 2020. Random Serial Dictatorship: The One and Only. *Mathematics of Operations Research* 45, 1 (2020), 353–368.
- P. Berman and M. Karpinski. 1999. On some tighter inapproximability results. In *Proceedings of the 26th International Colloquium on Automata, Languages, and Programming (ICALP)*. Springer-Verlag, 200–209.
- A. Bogomolnaia and H. Moulin. 2001. A New Solution to the Random Assignment Problem. *Journal of Economic Theory* 100, 2 (2001), 295–328.
- F. Brandl, F. Brandt, and J. Hofbauer. 2015. Incentives for Participation and Abstention in Probabilistic Social Choice. In *Proceedings of the 14th International Conference on Autonomous Agents and Multiagent Systems (AAMAS)*. IFAAMAS, 1411–1419.
- F. Brandt. 2018. Collective Choice Lotteries: Dealing with Randomization in Economic Design. In *The Future of Economic Design*, J.-F. Laslier, H. Moulin, R. Sanver, and W. S. Zwicker (Eds.). Springer-Verlag. Forthcoming.
- Graham Brightwell and Peter Winkler. 1991. Counting Linear Extensions. *Order* 8, 3 (1991), 225–242. <https://doi.org/10.1007/BF00383444>
- Russ Bubley and Martin E. Dyer. 1999. Faster random generation of linear extensions. *Discret. Math.* 201, 1-3 (1999), 81–88. [https://doi.org/10.1016/S0012-365X\(98\)00333-1](https://doi.org/10.1016/S0012-365X(98)00333-1)
- I. Caragiannis and S. Homrighausen. 2024. Estimating the expected social welfare and cost of random serial dictatorship. In *Proceedings of the 17th International Symposium on Algorithmic Game Theory (SAGT)*. Springer-Verlag, 184–201.
- Samuel Dittmer and Igor Pak. 2020. Counting Linear Extensions of Restricted Posets. *The Electronic Journal of Combinatorics* 27, 4 (2020), P4.48. <https://doi.org/10.37236/8552>
- E. Elkind, M. Lackner, and D. Peters. 2022. Preference Restrictions in Computational Social Choice: A Survey. *CoRR* abs/2205.09092 (2022).
- A. Gibbard. 1977. Manipulation of schemes that mix voting with chance. *Econometrica* 45, 3 (1977), 665–681.
- Mark Jerrum and Alistair Sinclair. 1989. Approximating the Permanent. *SIAM J. Comput.* 18, 6 (1989), 1149–1178.
- Mark Jerrum, Leslie G. Valiant, and Vijay V. Vazirani. 1986. Random Generation of Combinatorial Structures from a Uniform Distribution. *Theor. Comput. Sci.* 43 (1986), 169–188.
- Richard M. Karp and Michael Luby. 1983. Monte-Carlo Algorithms for Enumeration and Reliability Problems. In *24th Annual Symposium on Foundations of Computer Science, Tucson, Arizona, USA, 7-9 November 1983*. IEEE Computer Society, 56–64.
- Alexander V. Karzanov and Leonid G. Khachiyan. 1991. On the Conductance of Order Markov Chains. *Order* 8, 1 (1991), 7–15. <https://doi.org/10.1007/BF00385809>

- David A. Levin, Yuval Peres, and Elizabeth L. Wilmer. 2009. *Markov Chains and Mixing Times*. American Mathematical Society, Providence, RI.
- Peter Matthews. 1991. Generating a Random Linear Extension of a Partial Order. *The Annals of Probability* 19, 3 (1991), 1367–1392. <https://doi.org/10.1214/aop/1176990349>
- C. Puppe and A. Slinko. 2026. Single-Peaked Domains and Generalisations. , 41-64 pages.
- Daniela Saban and Jay Sethuraman. 2015. The Complexity of Computing the Random Priority Allocation Matrix. *Mathematics of Operations Research* 40, 4 (2015), 1005–1014. <https://doi.org/10.1287/moor.2014.0707>
- Alistair Sinclair. 1992. Improved Bounds for Mixing Rates of Markov Chains and Multicommodity Flow. *Comb. Probab. Comput.* 1 (1992), 351–370.
- G. Tullock. 1967. *Towards a Mathematics of Politics*. University of Michigan Press.

A ALGORITHM IN SECTION 3.2

Algorithm 2 Sampler over permutations for which SD yields target alternative $a \in A$

```

1   $N' \leftarrow N$ 
2   $A' \leftarrow A$ 
3   $\pi^* \leftarrow$  empty list
4  while  $N' \neq \emptyset$  do
5    if  $a \notin \max_{\succsim_i}(A')$  for all  $i \in N'$  then
6      return “no permutations yielding  $a$ ”
7    else
8      Consider the set  $\{i \in N' : a \in \max_{\succsim_i}(A')\}$  and choose an  $i \in \{i \in N' : a \in \max_{\succsim_i}(A')\}$ 
        uniformly at random from the set.
9       $N' \leftarrow N' \setminus \{i\}$ 
10      $A' \leftarrow \max_{\succsim_i}(A')$ 
11     Append  $i$  to  $\pi^*$ 
12   end if
13 end while
14 return  $\pi^*$ 

```

B PROOF OF LEMMA 4

PROOF. Let S be the surviving set after the common prefix of length $i - 1$, let $x = \pi'(i)$, and let β be the block in positions $i, \dots, j - 1$ of π . Since π' selects a , agent x retains a from S , leaving $S \cap E_x(a)$. Since π selects a , both β and the following x retain a in their original order. By Lemma 3,

$$SD_{S \cap E_x(a)}(\beta) = E_x(a) \cap SD_S(\beta) = SD_S(\beta x).$$

Thus $x\beta$ and βx reach exactly the same surviving set. The remaining suffix therefore still selects a , and the fixed suffix defining Ω_k is unchanged. $\square$

C PROOF OF LEMMA 5

PROOF. Given $\pi, \pi' \in \Omega_k$, repeatedly apply the insertion from Lemma 4 at the first position where the current permutation differs from π' . Each move increases the common-prefix length, so at most $k - 1$ accepted insertions transform π into π' . This proves irreducibility. Laziness gives aperiodicity.

Each ordered insertion proposal has probability $1/(2k(k - 1))$. For every proposal that maps a state π to a distinct state π' , the reverse ordered proposal maps π' to π . (An adjacent swap can be generated by two ordered proposals in each direction.) Hence the total transition probabilities satisfy $P(\pi, \pi') = P(\pi', \pi)$. The transition matrix is symmetric, so the uniform distribution is stationary and the chain is reversible. $\square$

D ILLUSTRATING THE INSERTION CHAIN AND THE COUNTING FRAMEWORK

Example 4 (Illustrating the insertion chain and the counting framework). Consider three agents $N = \{1, 2, 3\}$, two alternatives $A = \{a, b\}$, and target alternative a . Let the preferences be dichotomous, with approval sets

$$\text{App}(1) = \{a\}, \quad \text{App}(2) = \{a, b\}, \quad \text{App}(3) = \{b\}.$$

Thus, agent 1 approves only a , agent 3 approves only b , and agent 2 is indifferent between the two alternatives.

Starting from the surviving set $\{a, b\}$, serial dictatorship selects a if and only if agent 1 appears before agent 3: if agent 1 acts first among $\{1, 3\}$, then the surviving set becomes $\{a\}$, whereas if agent 3 acts first among $\{1, 3\}$, then the surviving set becomes $\{b\}$. Consequently, the set of selecting permutations is

$$\Omega = \{\pi \in \Pi^N : SD(\pi) = \{a\}\} = \{(1, 2, 3), (1, 3, 2), (2, 1, 3)\}.$$

and therefore

$$|\Omega| = 3 \quad \text{and} \quad s_a = \frac{|\Omega|}{3!} = \frac{1}{2}.$$

We now build the lazy Insertion Markov Chain on Ω . The states are the three selecting permutations 123, 132, 213. From a current state, the chain stays put with probability $1/2$. Otherwise, it chooses an ordered pair $(i, j) \in [3] \times [3]$ with $i \neq j$ uniformly at random, removes the agent currently in position j , and reinserts that agent into position i , accepting the move only if the resulting permutation remains in Ω .

For example, from state 123, exactly two ordered insertion proposals lead to 132, namely $(i, j) = (2, 3)$ and $(3, 2)$, and exactly two ordered insertion proposals lead to 213, namely $(1, 2)$ and $(2, 1)$. Since there are $3 \cdot 2 = 6$ ordered proposals and the non-lazy step is taken with probability $1/2$, we obtain

$$\begin{aligned} P(123, 132) &= \frac{1}{2} \cdot \frac{2}{6} = \frac{1}{6}, \\ P(123, 213) &= \frac{1}{6}, \quad P(123, 123) = \frac{2}{3}. \end{aligned}$$

A similar calculation gives

$$\begin{aligned} P(132, 123) &= \frac{1}{6}, \quad P(132, 213) = \frac{1}{12}, \\ P(132, 132) &= \frac{3}{4}, \end{aligned}$$

and

$$\begin{aligned} P(213, 123) &= \frac{1}{6}, \quad P(213, 132) = \frac{1}{12}, \\ P(213, 213) &= \frac{3}{4}. \end{aligned}$$

Hence, in the state order (123, 132, 213), the transition matrix is

$$P = \begin{pmatrix} 2/3 & 1/6 & 1/6 \\ 1/6 & 3/4 & 1/12 \\ 1/6 & 1/12 & 3/4 \end{pmatrix}.$$

This example also illustrates the telescoping counting method of Section 5. In the state space Ω , the final position is occupied by agent 3 in two states (123 and 213) and by agent 2 in one state (132). Thus

$$R_{3,1} = 0, \quad R_{3,2} = \frac{1}{3}, \quad R_{3,3} = \frac{2}{3}.$$

Fixing the most frequent final agent 3 yields the suffix-restricted state space

$$\Omega_2 = \{12, 21\},$$

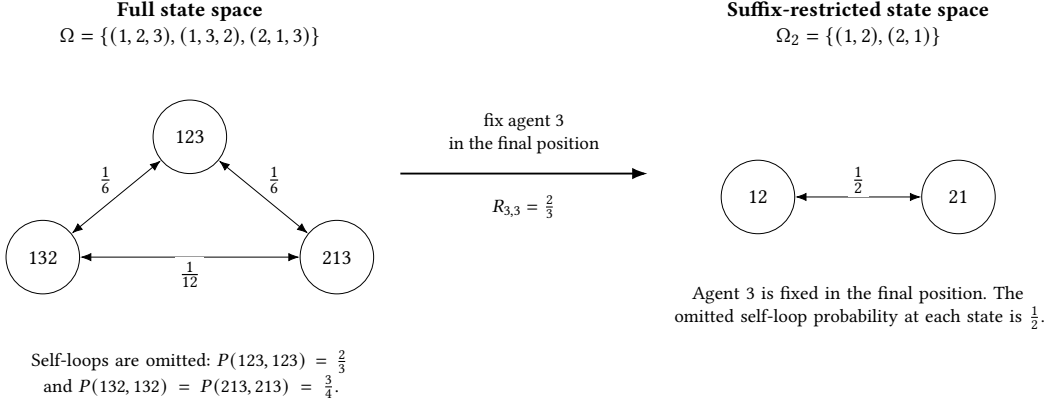


Fig. 1. Illustration of the lazy Insertion Markov Chain and the suffix-restricted counting procedure. The left panel gives the full chain on the permutations selecting a . Since agent 3 occupies the final position in two of the three states, fixing agent 3 last produces the two-state chain on the right. The telescoping calculation gives $|\Omega| = (2/3)^{-1}(1/2)^{-1} = 3$, and hence $s_a = 3/3! = 1/2$.

corresponding to the original states 123 and 213. In Ω_2 , each remaining agent appears in the final unfixed position with probability $1/2$. Therefore,

$$|\Omega| = \frac{1}{R_{3,3}} \cdot \frac{1}{R_{2,2}} = \frac{1}{2/3} \cdot \frac{1}{1/2} = 3.$$

Dividing by $3!$ recovers the RSD share

$$s_a = \frac{|\Omega|}{3!} = \frac{1}{2}.$$

Thus, this small example illustrates both how the insertion chain is constructed and how suffix-restricted sampling can be used to count the selecting permutations. Figure 1 illustrates the lazy Insertion Markov Chain on a tiny instance. The selecting permutations are $\Omega = \{123, 132, 213\}$, so the exact RSD share is $3/3! = 1/2$. The figure also visualises the suffix-restriction step used by Algorithm 1: since agent 3 occupies the final position in a $2/3$ -fraction of the states in Ω , fixing 3 last yields the reduced state space $\Omega_2 = \{12, 21\}$, from which the telescoping product recovers $|\Omega| = (2/3)^{-1}(1/2)^{-1} = 3$.

E PROOF OF THEOREM 3

PROOF. Apply Lemma 2 to the complete bipartite graph $G = K_{r,r}$ with sides V_L and V_R and with $q = 2r$ test agents. There are $n = 4r$ agents. Retain the lemma's notation $I(\pi)$ for the vertices occurring after the first test agent. Every independent set of $K_{r,r}$ lies entirely in one side, so

$$\Omega = \{\pi : I(\pi) \subseteq V_L\} \cup \{\pi : I(\pi) \subseteq V_R\}.$$

The intersection is $B = \{\pi : I(\pi) = \emptyset\}$, where all vertex agents precede all test agents. Set

$$S = \{\pi : \emptyset \neq I(\pi) \subseteq V_R\}.$$

By symmetry, $0 < \tau(S) \leq 1/2$ for the uniform stationary distribution τ .

Every accepted transition leaving S enters B . Indeed, moving a vertex agent changes only that vertex's membership in $I(\pi)$, so it cannot replace a nonempty subset of V_R by a nonempty subset

of V_L . Moving a test agent preserves the relative order of all vertex agents, so the old and new sets $I(\pi)$ are nested. This also precludes a transition between the two nonempty sides.

The counts in Lemma 2 now give

$$w_s = (2r - s)!(2r)(2r + s - 1)!, \quad |B| = w_0 = (2r)!(2r)!, \quad \frac{w_{s+1}}{w_s} = \frac{2r + s}{2r - s} \geq 1$$

for $0 \leq s < r$. Thus $w_s \geq w_0$ for $0 \leq s \leq r$, and

$$|S| = \sum_{s=1}^r \binom{r}{s} w_s \geq (2^r - 1)|B|.$$

By reversibility and the boundary property, $Q(S, \bar{S}) = Q(S, B) = Q(B, S) \leq \tau(B)$. Therefore,

$$\Phi \leq \frac{Q(S, \bar{S})}{\tau(S)} \leq \frac{|B|}{|S|} \leq \frac{1}{2^r - 1}.$$

The standard conductance bound for lazy reversible chains [Levin et al., 2009] yields

$$t_{\text{mix}}(1/4) \geq \frac{1}{4\Phi} \geq \frac{2^r - 1}{4}.$$

Since $n = 4r$, this is exponential in the number of agents. The instance has $r^2 + 1$ alternatives and an explicit approval matrix of polynomial size in r , so the lower bound is also superpolynomial in input length. $\square$

F PROOF OF THEOREM 5

PROOF. Consider a nonempty suffix-restricted state space $\Omega_k = \Omega(R, \sigma)$. A witnessing prefix of length $b \leq \min\{c, k\}$ gives $|\Omega_k| \geq (k - b)!$, and hence

$$\frac{k!}{|\Omega_k|} \leq \frac{k!}{(k - b)!} \leq k^b \leq k^c.$$

For $k = 1$ the state space is a singleton. Assume $k \geq 2$. By Lemma 5, the Insertion Markov Chain is lazy and has uniform stationary distribution. Each nontrivial canonical-path transition has probability at least $1/(2k(k - 1))$; adjacent swaps have two insertion proposals, which only increases this probability. Lemma 6 therefore gives

$$\rho \leq 2k(k - 1) \max_e \frac{|\text{cp}(e)|}{|\Omega_k|} \leq 2k(k - 1) \frac{k!}{|\Omega_k|} \leq 2k^{c+2}.$$

Every path has length at most $k - 1 \leq k$, and $\tau_{\min} = 1/|\Omega_k| \geq 1/k!$. The canonical-path bound yields

$$t_{\text{mix}}(\eta) \leq 2k^{c+3} \left(\log(k!) + \log \frac{1}{\eta} \right).$$

This is polynomial for fixed c . Each transition can be implemented in polynomial time by simulating serial dictatorship, so Theorem 2 applies. $\square$