Games and Economic Behavior 157 (2026) 263–285

Contents lists available at ScienceDirect

Games and Economic Behavior

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

Sequential unanimity voting rules for binary social choice

Stergios Athanasoglou [a,*], Somouaoga Bonkoungou [b]

[a] Department of Economics, University of Milan - Bicocca, Milan, Italy
[b] Department of Economics, Université Norbert Zongo, Koudougou, Burkina Faso

ARTICLE INFO

Keywords:
Binary voting
M-winning coalition rule
Simple games
Sequential unanimity rule
Neutrality
Strategy-proofness

ABSTRACT

We consider a group of voters that needs to decide between two candidates. In this setting, M-winning coalition rules are characterized by neutrality and strategy-proofness (Moulin, 1983). Their practical implementation motivates the introduction of sequential unanimity rules. We develop algorithms that transform a given M-winning coalition rule into an equivalent sequential unanimity rule and vice versa. The sequential unanimity rules that are constructed present computational advantages compared to their M-winning counterparts. The analysis extends to the full preference domain. Since M-winning coalition rules are closely related to strong and proper simple games, the analysis is relevant to this strand of the game-theoretic literature as well.

1. Introduction

We study voting rules for settings featuring two alternatives, or candidates. Despite its simplicity, this framework has many practical applications. In politics, elections held in two-party systems (parliamentary or presidential), as well as referendums, determine the outcome of a binary choice. In criminal legal proceedings, juries must render a verdict on a defendant's guilt. The determination of monetary policy, in which a group of central bankers needs to decide between two or three alternatives (raise, lower, or keep steady interest rates), provides yet another example of this kind of discrete decision-making process (Riboni and Ruge-Murcia, 2010).

Selecting among voting rules is informed by the fairness, efficiency, and non-manipulability properties that they satisfy. Let us highlight some desirable properties in this regard. A rule is *strategy-proof* if it is not possible for any voter to profitably misrepresent his preferences. A rule is *neutral* if it treats all candidates equally, and thus does not systematically favor one over another. Neutrality is a salient fairness principle in contexts where candidates represent entities that are harmed by discriminatory practices (political candidates, job applicants, etc.). Finally, a rule is *anonymous* if it treats all voters equally, rendering all aspects of their identity irrelevant to the voting process.

In an important contribution, May (1952) showed that majority rule is characterized by anonymity, neutrality, and positive responsiveness, a straightforward monotonicity property that reduces to strategy-proofness when preferences are strict. This result, known as May's theorem, provided axiomatic foundations for majoritarian rules and had a major impact on the subsequent literature. However, despite the theoretical and intuitive appeal of majority rule, there are a number of settings in which voters are granted differential power. First, few actual political systems are consistent with pure majority rule. Any system that disaggregates the electorate into districts, whose elected representatives form a centralized body such as an assembly or parliament, will involve the violation of anonymity to some degree (Bartholdi et al., 2021). This is also true of presidential systems of government that decompose the electorate into blocks and then aggregate the block-level results. Second, in many contexts, seniority or expertise is a valuable dimension in the decision process that justifies differences in voting power among agents.

* Corresponding author.
E-mail addresses: stergios.athanasoglou@unimib.it (S. Athanasoglou), bkgsom@gmail.com (S. Bonkoungou).

https://doi.org/10.1016/j.geb.2026.02.002
Received 20 September 2024
Available online 14 February 2026

Motivated by the above observations, we relax anonymity and study strategy-proof and neutral rules. To begin, we focus on the strict-preference case and consider M-winning coalition rules. First introduced by Moulin (1983), this important family of rules represents the springboard for our analysis. A subset of voters C is a *winning coalition* of a rule if, whenever all voters in C prefer the same candidate, and all other voters prefer the other candidate, then the rule picks C's preference. An *M-winning coalition set* is a collection of winning coalitions that satisfies two properties: *minimality* and *Moulin's property*.[1] *M-winning coalition rules*, of which majority rule is a special case, are parameterized by an M-winning coalition set $\mathcal{C}$. For each profile of preferences, there is at least one element of $\mathcal{C}$ whose members are unanimous in their choice of preferred candidate and there can be no two elements of $\mathcal{C}$ who have different unanimous preferences. The M-winning coalition rule associated with $\mathcal{C}$ selects for each profile of preferences exactly that unique unanimous outcome. Minimality and Moulin's property ensure that the rule is well-defined, neutral and strategy-proof. It is important to note that M-winning coalition rules are identical to *constant-sum simple games*. Simple games have received considerable attention in the cooperative game-theoretic literature (Taylor and Zwicker, 1999).

The implementation of M-winning coalition rules requires searching for a winning coalition whose members are unanimous. A naive way of approaching this problem is to order winning coalitions randomly and, given a profile of voter preferences, go down the ordered list until a unanimous coalition is identified. At this point the procedure terminates and the winning candidate is determined. The structure of M-winning coalition sets ensures that this procedure is well-defined. However, depending on the preference profile and the structure of the sequence, this sequential check may be very time-consuming, where time is measured in the number of coalitions whose preferences we need to check. One of the main contributions of our work is to demonstrate that there are more efficient ways of performing this sequential check. The following example illustrates the key idea.

A concrete example. The following example is useful in illustrating the paper's framework and basic results. Suppose we have a panel of seven professors who need to decide between two job market candidates, Ann and Bob. Professors 1 and 2 are full professors, 3 and 4 are associate professors, and 5,6 and 7 are assistant professors. Reflecting differences in seniority and status, the opinion of a full professor carries twice the weight of that of an associate, and the opinion of an associate carries twice the weight of an assistant. The decision process is such that the candidate who enjoys the support of at least two full professors, or their weighted equivalent, will be selected.

While not immediately obvious, the above rule is an instance of a M-winning coalition rule having M-winning coalition set $\mathcal{C} = \{\{1,2\},\{1,3,4\},\{2,3,4\},\{1,3,5,6\},\{1,3,5,7\},\{1,3,6,7\},\{1,4,5,6\},\{1,4,5,7\},\{1,4,6,7\},\{2,3,5,6\},\{2,3,5,7\},\{2,3,6,7\},\{2,4,5,6\},\{2,4,5,7\},\{2,4,6,7\}\}$.[2] For instance, if professors 1 and 2 both prefer Ann to Bob, Ann is selected. Conversely, if professors 1,3,5, and 6 prefer Bob to Ann, then Bob receives support from 1 full professor, 1 associate professor, and two assistant professors. This adds up to the weighted equivalent of two full professors and thus Bob is selected.

One way to implement this rule would be to ask all the professors to assess the candidates, search for an element of $\mathcal{C}$ whose members have unanimous preferences, and select the corresponding candidate. However, candidate assessment is a time-consuming and cognitively demanding task. Asking for all professors to weigh in is not optimal: ideally, we would like to ask for a faculty member's opinion only if it is needed. With this in mind, an alternative, and arguably preferable, approach is to proceed sequentially in the following manner: first, solicit the opinion of professors 1 and 2. If they are unanimous, then the winning candidate is determined and we can stop there. If not, consult professors 3 and 4 and if *they* are unanimous, then once again the winner is determined (for she or he will have the support of one full and two associate professors, which add up to an equivalent of two full professors). Conversely, if professors 3 and 4 are not unanimous, then proceed with professors 5 and 6 and repeat: if they are unanimous, then once again the winner is determined. Otherwise, we move on and terminate the procedure by simply picking Professor7′s preferred candidate.

Contributions. The paper formalizes the idea developed in the example. We propose a new family of rules, *sequential unanimity rules*, that are equivalent to M-winning coalition rules but arguably simpler to implement. Sequential-unanimity rules are parameterized by a sequence $\mathrm{S} = (S_1, \ldots, S_K)$ of non-recurring voter subsets whose last element is a singleton. Briefly, here is how they work. Given a profile of preferences, a sequential unanimity rule examines the first subset in its associated sequence S. If the preferences of the voters composing it are unanimous, the rule stops and outputs this preferred candidate. Otherwise, it considers the next element in the sequence S and repeats the procedure until it encounters a group of voters who all agree on their preferred candidate. At that point, the process terminates and the rule outputs this candidate.

Taking as input an M-winning coalition rule, we construct a sequential unanimity rule with desirable properties. In particular, we develop an algorithm ("Algorithm 1") that transforms a given M-winning coalition rule into an equivalent sequential unanimity rule. This algorithm is a nontrivial procedure, in that it never results in a simple enumeration of all the elements of an M-winning coalition set plus a final backstop. The sequence produced by Algorithm 1 will consist of a number of subsets that is strictly smaller, oftentimes significantly so, than the cardinality of the original M-winning coalition set (see Corollary 1). This suggests that, from a naive worst-case perspective, sequential unanimity rules will be more computationally efficient than M-winning coalition rules. Let us be clear on what we mean by this statement. Suppose we find ourselves in a context in which the M-winning coalition set lacks any obvious structure that can be exploited in order to quickly determine the winning candidate. As mentioned earlier, one way we can

[1] A collection of winning coalitions is *minimal* if no winning coalition within it is a proper subset of another. A collection of winning coalitions satisfies *Moulin's property* if, for any subset of voters C, we have that C has a non-empty intersection with all elements of the collection if and only if it is a superset of an element of the collection. See Definition 4 for a clear formulation.

[2] Equivalently, this setting can be interpreted as a weighted constant-sum simple game with weight 4 for voters 1 and 2, weight 2 for voters 3 and 4, and weight 1 for voters 5, 6, and 7. The aggregate weight threshold defining a winning coalition is 8.

proceed is by ordering winning coalitions randomly and, given a profile of voter preferences, go down the ordered list one-by-one until we find one whose members are unanimous. A related but different way of approaching this problem is to conduct a sequential search via one of the sequential unanimity rules produced by Algorithm 1. Depending on the preference profile and the structure of the two sequences, one kind of sequential check will reach a conclusion sooner than the other or they might take the same amount of time, where time is measured in the number of coalitions whose preferences we need to check. Nevertheless, if we consider the worst case over the set of all possible preference profiles, the first process will take longer to terminate than the second (see Corollary 2).[3] An additional advantage of this approach is that the elements of the sequence produced by Algorithm 1 are themselves subsets, and sometimes proper subsets, of elements of the M-winning coalition set. Thus, not only do we need to check a smaller number of subsets, but the subsets that we do check will tend to be smaller than their counterparts in the M-winning coalition sets.

Motivated by similar efficiency concerns, we define and investigate a subclass of sequential unanimity rules that we refer to as *essential*. These sequential rules have the additional property that there exist no superfluous elements in their corresponding sequence: that is, removing any element of the sequence results in a different rule. This concern is not merely academic as we show that Algorithm 1 may produce sequences that are not essential. To address it, we identify an additional criterion that, when added to the requirements of Algorithm 1, provides necessary and sufficient conditions for the rules to be essential. Subsequently, we propose a different algorithm ("Algorithm 2") that transforms a given sequential unanimity rule into an equivalent M-winning coalition rule. We thus have a complete equivalence result between these two families of rules. In addition, upon slightly modifying the two algorithms, the equivalence carries over to the full preference domain, which allows for voters to be indifferent between candidates. We perform this analysis in section D of the Appendix.[4]

The above results are potentially relevant to the theory of constant-sum simple games, since they suggest a systematic sequential procedure for determining their outcome. This is particularly important for constant-sum simple games that are *not weighted*. In such games, it is impossible to assign weights to the voters and define winning coalitions as consisting of those subgroups whose aggregate weight exceeds some threshold (Taylor and Zwicker, 1999). While weighted schemes are considerably more intuitive and widespread, non-weighted systems are not a mere theoretical curiosity. Several actual voting procedures, including the United States Federal System and the European Union Council, correspond to non-weighted constant-sum simple games (see Examples 1.4-1.5-2.4 in Freixas and Molinero, 2009). In principle, our Algorithm 1 provides a promising tool for simplifying such games and computing their outcome in a faster way. We demonstrate its application in one such example in Section 3.1.

Related work. Our work is primarily relevant to the literature on binary social choice. As mentioned earlier, the canonical contribution in this area is May's Theorem (May, 1952), which provides a firm axiomatic foundation for majority rule. Subsequently, drawing on work by Fishburn and Gehrlein (1977), Moulin (1983) (page 64) showed that M-winning coalition rules uniquely satisfy strategy-proofness and neutrality. Along related lines, Larsson and Svensson (2006) axiomatized a different but similar family of rules, known as *voting by committee*, with strategy-proofness and ontoness.

There are a number of papers that examine the implications of anonymity for strategy-proof social choice in the two-candidate context. Lahiri and Pramanik (2020) generalized the characterization of Larsson and Svensson in a model with indifferences. In a series of recent papers, Basile et al. (2021, 2022c,b,a) provided alternative characterizations of strategy-proof anonymous rules that are simpler than Lahiri and Pramanik (2020) and admit explicit functional forms. To address the definitional complexity of the rules of Lahiri and Pramanik (2020), Basile et al. (2022c) offered an alternative, explicit characterization that is easier to work with. In a companion paper, Basile et al. (2021) extended this analysis to the case in which the voters must decide between two candidates but can indicate preferences over a wider set. In Basile et al. (2022b) they developed a unified framework that leads to a representation of various classes of strategy-proof rules, including anonymous ones.

Anonymity, while intuitive, is rarely satisfied in political systems based on some form of representative democracy. In such settings weaker standards of inter-voter fairness apply. Motivated by this observation, Bartholdi et al. (2021) proposed k-equity, a parametric relaxation of anonymity. This weaker requirement leads to the identification of a rich family of equitable rules that satisfy neutrality and positive responsiveness. Using insights from group theory, Bartholdi et al. (2021) were able to establish lower and upper bounds of magnitude $O(\sqrt{n})$ on the size of the winning coalitions defining such rules. Their work implies that consensus between a relatively small number of voters can be sufficient to determine the outcome of voting systems that meet reasonable standards of fairness. Working within a similar framework as Bartholdi et al. (2021), Kivinen (2023) studied the equity-manipulability tradeoff inherent in the design of voting rules. He introduced a slightly different version of equity, pairwise equity, and showed that for rules satisfying it anonymity is equivalent to various measures of group strategy-proofness.

As mentioned before, our analysis is also relevant to the literature on simple games (Taylor and Zwicker, 1999). Necessary and sufficient conditions for a simple game to be weighted have been the object of consistent study in the game-theoretic (Houy and Zwicker, 2014; Taylor and Zwicker, 1993, 1996) and discrete-mathematics (Freixas and Molinero, 2009; Taylor and Zwicker, 1992, 1995) literature. It is not immediately clear how these insights affect the analysis in our paper, in particular the impact they might

[3] We must stress that in making this comparison, we are focusing on the worst-case performance of a naive brute-force search of the M-winning coalition set. Clearly, more clever search algorithms may be devised and this is especially true if we assume further structure for the M-winning coalition set. At the same time, we contend that there is value in working behind a veil of ignorance and proposing a procedure that provides modest worst-case improvements across all possible M-winning coalition sets. Indeed, Algorithm 1 may well be viewed under this lens: as a systematic, albeit preliminary, attempt at designing a more efficient way of determining the winner of *any* kind of election.

[4] We stress that our results do not imply a characterization of these rules in the full domain case. Indeed, we are not aware of any characterization of neutral and strategy-proof rules when voters can be indifferent between the two candidates.

have on the structure and implementation of Algorithms 1 and 2. Getting a clearer picture of the connection between our analysis and the theory of simple games constitutes an interesting avenue of future research.

Finally, exploring the connections between static and sequential rules has been the object of study in other contributions to mechanism design (Gershkov and Szentes, 2009; Gershkov et al., 2017).

2. Model

2.1. M-winning coalition rules

Let N be a finite set of n voters and $A = \{a, b\}$ a set of two candidates. Voter i's preference relation over candidates is denoted by R_i so that $R_i = x$ for some $x \in A$ means that voter i prefers candidate x to candidate $y \neq x$. A **profile** R_N is an n-tuple of preferences and the space of profiles is denoted by $\mathcal{R}^N$. A **rule** is a function $f : \mathcal{R}^N \to A$ that assigns a candidate to each profile.

Definition 1. A rule f is **strategy-proof** if for all profiles $R_N \in \mathcal{R}^N$, each voter $i \in N$ and each $R_i' \neq R_i$, i does not prefer $f(R_i', R_{-i})$ to $f(R_N)$.

Given a permutation $\pi : A \to A$ of A and a profile R_N, define the profile πR_N as $\pi R_i = \pi(R_i)$ for all $i \in N$.

Definition 2. A rule f is **neutral** if for each permutation π of A, and all profiles $R_N \in \mathcal{R}^N$,

$$f(\pi R_N) = \pi(f(R_N)).$$

We now introduce the concept of a winning coalition of a rule.

Definition 3. A subset $C \subset N$ of voters[5] is a **winning coalition** of rule f if for all profiles $R_N \in \mathcal{R}^N$ satisfying $R_i = x$ for all $i \in C$ and $R_i \neq x$ for all $i \notin C$, we have $f(R_N) = x$.

The attentive reader will notice that the above definition of winning coalitions is consistent with (Moulin, 1983) and slightly different from Bartholdi et al. (2021).[6] The two definitions coincide under the added assumption of strategy-proofness or positive responsiveness.

Definition 4. A non-empty set $\mathcal{C}$ of non-empty subsets of N is an **M-winning coalition set** if it satisfies the following two properties:

P1. for all $C, C' \in \mathcal{C}$ we have $C \subset C' \Rightarrow C = C'$ [Minimality];
P2. for all $C \subset N$, we have $\left[C \supset C' \text{ for some } C' \in \mathcal{C}\right] \Leftrightarrow \left[C \cap C'' \neq \emptyset \text{ for all } C'' \in \mathcal{C}\right]$ [Moulin's Property].

Minimality is a straightforward property that allows us to focus only on coalitions that do not contain superfluous elements. Moulin's property however is a little more subtle. In words, it ensures that (i) all pairs of elements of $\mathcal{C}$ have a nonempty intersection and (ii) any subset having a nonempty intersection with all elements of $\mathcal{C}$ is a superset of some element of $\mathcal{C}$. Moulin's property is essential for M-winning coalition sets to systematically determine the winner of a binary election. In particular, part (i) of its definition ensures that there exist no two winning coalitions that yield conflicting results, while part (ii) ensures the existence of at least one winning coalition whose members are unanimous and thus determine the winner of the election. These facts will become clearer once we have defined M-winning coalition rules.

The structure of M-winning coalition sets can range from highly ordered and symmetric to apparently quite random. The following two examples make this clear.

1. Let $N = \{1, 2, \ldots, 9\}$ and $\mathcal{C}^1 = \{C \subset N : |C| = 5\}$ (i.e., majority rule with 9 voters) and
2. Let $N = \{1, 2, \ldots, 8\}$ and $\mathcal{C}^2 = \{\{1,2,3\}, \{1,2,4\}, \{1,2,7\}, \{2,3,4\}, \{1,3,5,6\}, \{1,3,5,7\}, \{1,3,6,7\}, \{1,4,5,6,8\}, \{2,3,5\}, \{2,3,6\}, \{2,3,8\}, \{2,5,7\}, \{2,6,7\}, \{2,4,5,6\}, \{3,4,7\}\}$

Both $\mathcal{C}^1$ and $\mathcal{C}^2$ are valid M-winning coalition sets. Let us see why. The set $\mathcal{C}^1$ clearly satisfies minimality since all of its elements have the same cardinality.

Let us now verify Moulin's property. Suppose that $C \subset N$ and $C \supset C'$ for some $C' \in \mathcal{C}^1$. This means $|C| \geq 5$. Let $C'' \in \mathcal{C}^1$, implying that $|C''| = 5$. Since $|N| = 9$ and $|C| \geq 5 = |C''|$, this implies $C \cap C'' \neq \emptyset$. Hence, $C \cap C'' \neq \emptyset$ for all $C'' \in \mathcal{C}^1$. Conversely, suppose that $C \subset N$ and $C \cap C'' \neq \emptyset$ for all $C'' \in \mathcal{C}^1$. If $|C| \leq 4$, then $|N \setminus C| \geq 5$. Hence, there exists $C' \subset N \setminus C$ such that $|C'| = 5 \Rightarrow C' \in \mathcal{C}^1$ and $C \cap C' = \emptyset$. This is a contradiction. Thus, $|C| \geq 5$, yielding $C \supset C'$ for some $C' \in \mathcal{C}^1$. The analysis of $\mathcal{C}^2$ is more involved and we refer the reader to Appendix A for a proof that it satisfies Minimality and Moulin's property.

Definition 5. Consider an M-winning coalition set $\mathcal{C}$. The **M-winning coalition rule** $\mathcal{C} : \mathcal{R}^N \mapsto A$, satisfies, for all $R_N \in \mathcal{R}^N$,

$$\mathcal{C}(R_N) = x \;\Leftrightarrow\; \exists C \in \mathcal{C} \text{s.t.} R_i = x \text{for all } i \in C.$$

[5] For clarity, we specify that in our paper the symbol $\subset$ denotes "subset of", while $\subsetneq$ denotes "proper subset of".
[6] They substitute the statement "$R_i = x$ for all $i \in C$ and $R_i \neq x$ for all $i \notin C$" with "$R_i = x$ for all $i \in C$".

First introduced by Moulin (1983), M-winning coalition rules are well-defined. We briefly demonstrate why this is true. First, we claim that for any profile R_N, there exists $C \in \mathcal{C}$ and $x \in A$ such that $R_i = x$ for all $i \in C$. To see this, let $N^a(R_N) = \{i \in N : R_i = a\}$ and $N^b(R_N) = \{i \in N : R_i = b\}$. We consider two cases. First, suppose that $N^a(R_N)$ has a non-empty intersection with every element of $\mathcal{C}$; that is, for all $C \in \mathcal{C}$, $N^a(R_N) \cap C \neq \emptyset$. By Moulin's property, there exists $C \in \mathcal{C}$ such that $N^a(R_N) \supset C$, proving that $\mathcal{C}(R_N) = a$. Second, suppose that there exists $C \in \mathcal{C}$ such that $N^a(R_N) \cap C = \emptyset$. Then $C \subset N^b(R_N)$ and for all $i \in C$ we have $R_i = b$. This implies that $\mathcal{C}(R_N) = b$. Finally, we claim that there are no $C, C' \in \mathcal{C}$ such that $R_i = a$ for all $i \in C$ and $R_j = b$ for all $j \in C'$. This holds because Moulin's property implies $C \cap C'' \neq \emptyset$ for all $C'' \in \mathcal{C}$ –including C'. In conclusion, for all profiles $R_N \in \mathcal{R}^N$, there exists $C \in \mathcal{C}$, and $x \in A$ such that $R_i = x$ for all $i \in C$ and there is no other $C' \in \mathcal{C}$ such that $R_j = y \neq x$ for all $j \in C'$.

As mentioned in the introduction, M-winning coalition rules are characterized by strategy-proofness and neutrality.

Theorem 1 (Moulin, 1983). *A rule is strategy-proof and neutral if and only if it is an M-winning coalition rule.*

2.2. Connection to simple games

M-winning coalition sets are closely linked to the concept of constant-sum (i.e., strong and proper) simple games in cooperative game theory (Taylor and Zwicker, 1999). This allows us to draw connections to the simple game literature. Some definitions are needed.

Definition 6. A **simple game** is a pair $(N, \mathcal{W})$, where N is a nonempty finite set and $\mathcal{W}$ is a collection of subsets of N satisfying the **monotonicity property**: if, for all X, $X \in \mathcal{W}$ and $X \subsetneq Y \subset N$, then $Y \in \mathcal{W}$. Moreover, $\mathcal{W} \notin \{\emptyset, 2^N\}$.

For our purposes, we need to further refine the simple game concept.

Definition 7. A simple game $(N, \mathcal{W})$ is

(i) **proper** if, for all X, $X \in \mathcal{W} \Rightarrow N \setminus X \notin \mathcal{W}$ and
(ii) **strong** if, for all X, $X \notin \mathcal{W} \Rightarrow N \setminus X \in \mathcal{W}$.

We are interested in simple games that are both strong and proper. Such simple games are referred to as *constant-sum*.

Definition 8. A simple game $(N, \mathcal{W})$ that is proper and strong is a **constant-sum** simple game.

Consider a simple game $(N, \mathcal{W})$. An element W of $\mathcal{W}$ is referred to as a **winning coalition**. A **minimal winning coalition** $W \in \mathcal{W}$ is a winning coalition that is not a proper subset of any other winning coalition in $\mathcal{W}$. More formally, if $W \in \mathcal{W}$ is such that for all $W' \subset N$,

$$\{W' \in \mathcal{W}, W' \subset W\} \Rightarrow W = W',$$

then W is a **minimal winning coalition.** Let $\mathcal{W}^m$ denote the **set of minimal winning coalitions** of simple game $(N, \mathcal{W})$. Because of monotonocity, any simple game $(N, \mathcal{W})$ is completely determined by its set of minimal winning coalitions. That is, given any two simple games $(N, \mathcal{W}_1)$ and $(N, \mathcal{W}_2)$ we have $\mathcal{W}_1 = \mathcal{W}_2$ if and only if $\mathcal{W}_1^m = \mathcal{W}_2^m$.

It is straightforward to show that M-winning coalition sets and constant-sum simple games are intimately connected. The following Proposition appears in Moulin (1983) as an exercise. For completeness, we provide a self-contained proof.

Proposition 1 (Moulin, 1983). *Consider a constant-sum simple game $(N, \mathcal{W})$. The corresponding set of minimal winning coalitions $\mathcal{W}^m$ satisfies minimality and Moulin's property. Hence, $\mathcal{W}^m$ is a well-defined M-winning coalition set.*

Proof. See Appendix C. □

A central issue in the simple-game literature is the property of weightedness. We say that a simple game $(N, \mathcal{W})$ is **weighted** if there exists a vector of weights $\boldsymbol{w} \in \mathfrak{R}^{|N|}$, where w_i is the weight of voter $i \in N$, and a threshold $q \in \mathfrak{R}$, such that $\mathcal{W} = \{W \subset N : \sum_{i \in C} w_i \geq q\}$.[7] A simple game that does not admit such a representation is **non-weighted**.

The job-market candidate example of the Introduction is an example of a weighted constant-sum simple game with weight vector $\boldsymbol{w} = (4, 4, 2, 2, 1, 1, 1)$ and threshold $q = 8$. In addition, the M-winning coalition set $\mathcal{C}^1$, right before Definition 5, is yet another example of such a game with $\boldsymbol{w} = (1, 1, 1, 1, 1, 1, 1, 1, 1)$ and $q = 5$.

Weighted constant-sum simple games constitute the canonical example of neutral and strategy-proof rules for binary social choice. But, not all constant-sum simple games are weighted and a considerable amount of research has been dedicated to ascertaining the exact conditions that render a game weighted or not (Freixas and Molinero, 2009; Houy and Zwicker, 2014; Taylor and Zwicker, 1992). A prominent necessary and sufficient condition for a simple game to be weighted is the property of *trade robustness*, proposed by Taylor and Zwicker (1992). Without giving a formal definition, the essence of trade robustness is that it is not possible to transform a set of winning coalitions into a set of losing coalitions via a sequence of inter-coalitional voter trades. The constant-sum simple game $\mathcal{C}^2$ mentioned earlier is not weighted because it fails exactly this criterion. To see this, consider the winning coalitions $C_1 = \{2, 3, 6\}$ and $C_2 = \{3, 4, 7\}$, which both belong to $\mathcal{C}^2$. Suppose voters 6 and 7 flip coalitions, resulting in the updated subsets $C_1' = \{2, 3, 7\}$ and $C_2' = \{3, 4, 6\}$. Since $C_1' \notin \mathcal{C}^2$ and $C_2' \notin \mathcal{C}^2$, the property of trade robustness is violated. Hence, the constant-sum simple game $(N, \mathcal{C}^2)$ is non-weighted.

[7] Note that $\boldsymbol{w}$ and q have to be carefully set, so as to avoid having two subsets $C, C' \in \mathcal{C}$ such that $C \cap C' = \emptyset$.

3. Sequential unanimity rules

The implementation of M-winning coalition rules requires searching for a winning coalition whose members are unanimous. How to conduct this search is not clear. A naive way of proceeding is to order the elements of the M-winning coalition set randomly and, given a profile of voter preferences, go down the ordered list until we find a coalition that yields a winning candidate. Though this process is guaranteed to produce a winner, it may be cumbersome. One of the main contributions of our work is to demonstrate that there are more efficient ways of performing this sequential check.

To this end, we introduce a family of rules that is parameterized by a sequence $\mathrm{S} = (S_1, \ldots, S_K)$ of non-empty, non-recurring subsets of N such that $|S_K| = 1$. We suppose, moreover, that the voter in the singleton S_K does not belong to any other set in the sequence. The **sequential unanimity rule** $\mathcal{S}$ considers each set in the sequence S in increasing order (from S_1 until S_K) and examines the preferences of the voters belonging to it. As soon as a set is encountered in which voter preferences are unanimous, the procedure stops and the outcome of rule $\mathcal{S}$ is exactly that candidate for which there is unanimous support. The following definition formalizes this idea.

Definition 9. Consider a sequence $\mathcal{S} = (S_1, \ldots, S_K)$ of non-empty, non-recurring subsets of N satisfying $|S_K| = 1$. The **sequential unanimity rule** $\mathcal{S} : \mathcal{R}^N \mapsto A$ is defined as follows: for all $R_N \in \mathcal{R}^N$,

$$\mathcal{S}(R_N) = x \;\Leftrightarrow\; \left[\exists\, k \in \{1, \ldots, K\}\; s.t.\; R_i = x\, for\, all\, i{\in}S_k and\; for\, all\; l < k there\, exist\; i, j \;\in\; S_l\; s.t.\; R_i \neq R_j.\right]$$

Proposition 2. *Sequential unanimity rules are neutral and strategy-proof.*

Proof. See Appendix C. □

3.1. From M-winning coalition rules to sequential unanimity rules

In this section, we show how an M-winning coalition rule can be transformed into a family of equivalent sequential unanimity rules. Our approach is constructive as we develop an algorithm that takes as input an M-winning coalition set and produces a sequence of voter subsets. We refer to this algorithm as Algorithm 1. The extension of this analysis to the full preference domain is relegated to Appendix D.

Before we begin our analysis, it is worth noting that M-winning coalition rules can be trivially reformulated as sequential unanimity rules. That is, given an M-winning coalition set $\mathcal{C} = \{C_1, \ldots, C_L\}$, we can define a sequential unanimity rule $\mathcal{S}$ having sequence $\mathrm{S} = (C_{j_1}, \ldots, C_{j_L}, \{i\})$, where $C_{j_1}, \ldots, C_{j_L}$ is an arbitrary ordering of $\{C_1, \ldots, C_L\}$ and $i \in N$. As the next section will make clear however, these sequences are needlessly long. They also fail an efficiency requirement that we introduce in Section 3.1.5.

3.1.1. General description of Algorithm 1

At a high level, Algorithm 1 works in the following way. Given an M-winning coalition set $\mathcal{C}$, it begins by choosing an arbitrary voter $i \in N$, which it designates as the "backstop" voter, and removes all elements of $\mathcal{C}$ that contain it. At each subsequent iteration, the algorithm further removes elements of $\mathcal{C}$. It does so by selecting a subset of voters N' that satisfies three specific criteria:

(i) it has cardinality at least 2,
(ii) it is a proper subset of some remaining element of $\mathcal{C}$, and
(iii) it has a nonempty intersection with all the previously discarded elements of $\mathcal{C}$.

Note that there may be more than subset $N' \subset N$ satisfying the above. In that case, the algorithm picks one arbitrarily, while respecting the following **proviso**: if N' satisfies (i)-(ii)-(iii) and there exists another N'' doing the same such that $N' \subsetneq N''$, then subset N'' cannot be selected. This condition is imposed in order to avoid producing a sequence $\mathrm{S} = (S_1, \ldots, S_K)$ such that $S_l \subsetneq S_k$ for some $l < k$. A sequence of this kind would be inefficient, as subset S_k is superfluous to the application of the rule. This is because the sequential unanimity rule associated with S can never reach an outcome immediately upon consulting the preferences of voters in S_k: (i) if voters in S_l are not unanimous, then also voters in S_k are not unanimous and (ii) if voters in S_l are unanimous, then the rule terminates before consulting subset S_k.[8]

When we can no longer identify a subset of N satisfying criteria (i)-(ii)-(iii), Algorithm 1 terminates. The output sequence S is constructed in the following way. In last place, we put the backstop voter. In the second to last place, we put subset N_2. Right before N_2 we put N_3, and so on, until the last such subset identified via the satisfaction of criteria (i)-(ii)-(iii). At that point, if the M-winning coalition set $\mathcal{C}$ has not been completely exhausted, then its remaining elements are placed, in any order, at the beginning of the sequence S.

What should be clear is that when applying Algorithm 1 different choices of backstop voter and/or subsequent voter subsets will yield different sequences. Indeed, we can associate an entire family of sequences $\mathcal{S}^{\mathcal{C}}$ to each M-winning coalition set $\mathcal{C}$.

We will show that the M-winning coalition rule $\mathcal{C}$ is equivalent to any sequential unanimity rule $\mathcal{S}$ having sequence $\mathrm{S} \in \mathcal{S}^{\mathcal{C}}$. Before doing so, and to generate some intuition for Algorithm 1, we illustrate it on the academic job market example discussed in the introduction.

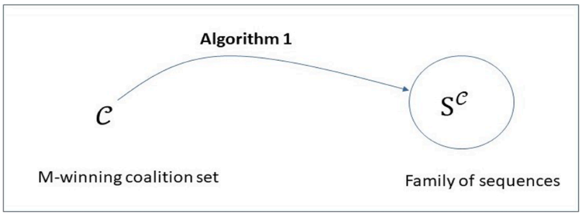

Fig. 1. Different applications of Algorithm 1 (i.e. different choices of backstop and N_k subsets during the course of the algorithm) associate to each M-winning coalition set $\mathcal{C}$ a family of sequences $S^{\mathcal{C}}$.

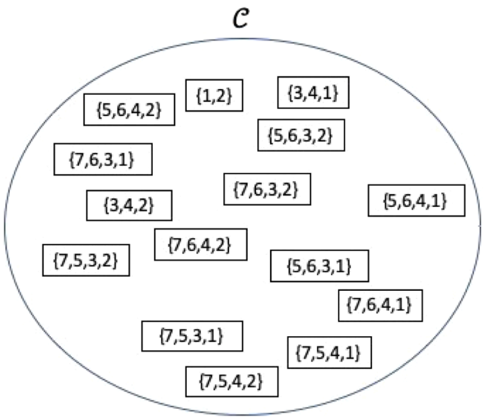

Fig. 2. M-winning coalition set $\mathcal{C}$.

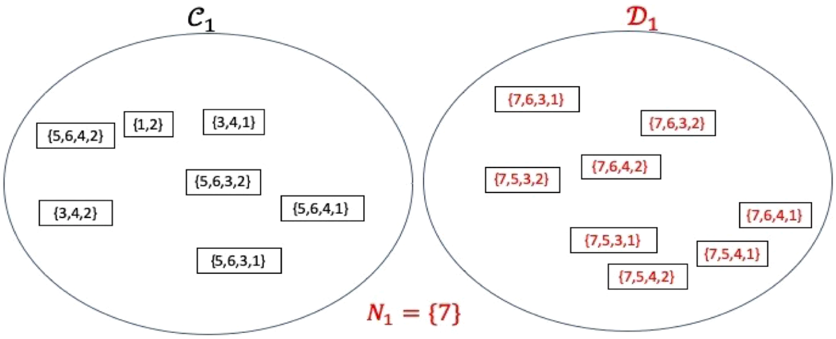

Fig. 3. Iteration 1: Set backstop voter $i = 7$ and remove all elements of $\mathcal{C}$ containing it. The remaining M-winning coalition set is $\mathcal{C}_1$ and the discarded M-winning coalition set is $\mathcal{D}_1$.

3.1.2. *An illustrative example*

We have $N = \{1, 2 \ldots, 7\}$ and the M-winning coalition set depicted in Fig. 2 below.

We begin applying Algorithm 1 by selecting the backstop voter. While we could pick any voter in N for this purpose, we want to be consistent with the introductory discussion and so we pick $N_1 = 7$. Subsequently, we update the M-winning coalition set by removing all its elements containing voter 7. For clarity, we refer to the *remaining* M-winning coalition set as $\mathcal{C}_1$ and the set of *discarded* winning coalitions of $\mathcal{C}$ as $\mathcal{D}_1$. We illustrate in Fig. 3.

We continue by searching for a subset of voters N_2 that satisfies the three criteria laid out in the previous subsection. There are various choices we could make at this point. Once again, for consistency with the introduction, we choose $N_2 = \{5, 6\}$. Fig. 4 highlights the elements of $\mathcal{C}_1$ that are proper supersets of N_2 and demonstrates why subset $\{5, 6\}$ is a is a valid choice. Fig. 5 displays the corresponding updates of the remaining M-winning coalition set $\mathcal{C}_1$ and the discarded M-winning coalition set $\mathcal{D}_1$, to $\mathcal{C}_2$ and $\mathcal{D}_2$ respectively. To be precise, $\mathcal{C}_2 = \mathcal{C}_1 \setminus \{\{5, 6, 3, 1\}, \{5, 6, 3, 2\}, \{5, 6, 4, 1\}, \{5, 6, 4, 2\}\}$ and $\mathcal{D}_2 = \mathcal{D}_1 \cup \{\{5, 6, 3, 1\}, \{5, 6, 3, 2\}, \{5, 6, 4, 1\}, \{5, 6, 4, 2\}\}$.

We continue by searching for a subset of voters N_3 that satisfies the three criteria laid out in the previous subsection. For similar reasons as before, we choose $N_3 = \{3, 4\}$. Fig. 6 highlights the elements of $\mathcal{C}_2$ that are proper supersets of N_3 and demonstrates why

[8] Note that, unless the proviso is imposed, Algorithm 1 might produce a sequence having this inefficient feature. See section 3.1.5.

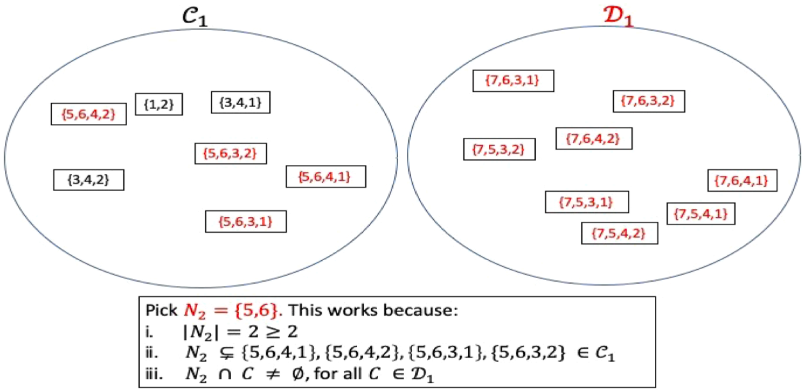

Fig. 4. Iteration 2: Selecting subset N_2.

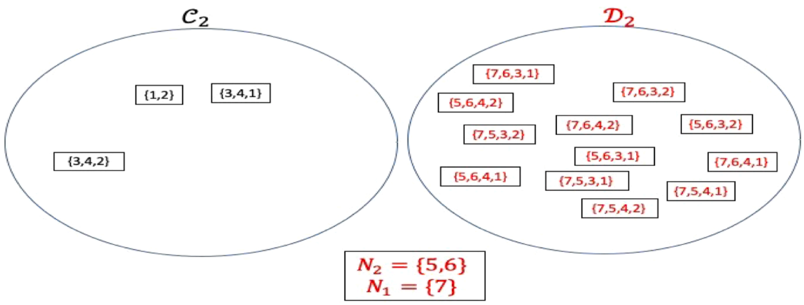

Fig. 5. Iteration 2: Updating sets $\mathcal{C}_1$ and $\mathcal{D}_1$ to $\mathcal{C}_2$ and $\mathcal{D}_2$.

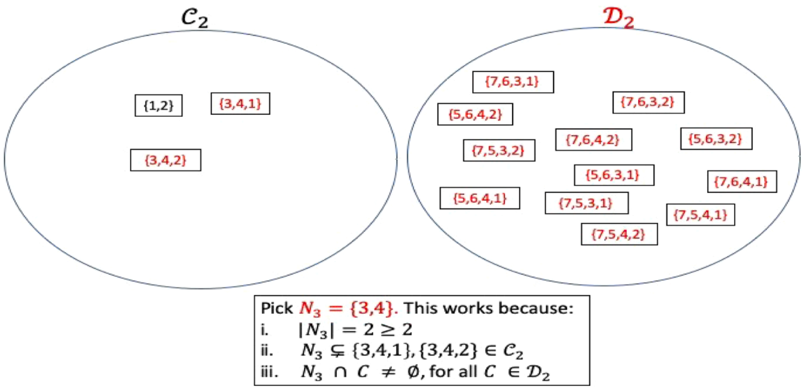

Fig. 6. Iteration 3: Selecting subset N_3.

subset $\{3,4\}$ is a is a valid choice. Fig. 7 displays the corresponding updates of the remaining M-winning coalition set $\mathcal{C}_2$ and the discarded M-winning coalition set $\mathcal{D}_2$, to $\mathcal{C}_3$ and $\mathcal{D}_3$ respectively.

At this point, there is a single element left in the remaining M-winning coalition set. Moreover, there is no way to satisfy all three criteria: in particular, criteria (i) and (ii) are mutually exclusive. Thus, Algorithm 1 terminates. The output sequence that it produces is the following:

$$\mathrm{S}^1 = (\{1,2\}, N_3, N_2, N_1) = (\{1,2\}, \{3,4\}, \{5,6\}, \{7\}).$$

Before moving on to the next subsection, we briefly comment on how different ways of applying Algorithm 1 can lead to very different output sequences. To this end, keeping the same coalition set $\mathcal{C}$ of Fig. 1 , we re-apply Algorithm 1 twice. For additional details on how the resulting sequences $\mathrm{S}^2, \mathrm{S}^3$ are obtained, please consult the paper's Appendix A.

First, we begin by selecting voter 4 as the backstop, so that $N_1 = \{4\}$. Subsequently, we set $N_2 = \{3,5,6\}, N_3 = \{3,6,7\}, N_4 = \{3,5,7\}$. At that point, only subset $\{1,2\}$ remains and the algorithm terminates with the following output sequence:

$$\mathrm{S}^2 = (\{1,2\}, N_4, N_3, N_2, N_1) = (\{1,2\}, \{3,5,7\}, \{3,6,7\}, \{3,5,6\}, \{4\}).$$

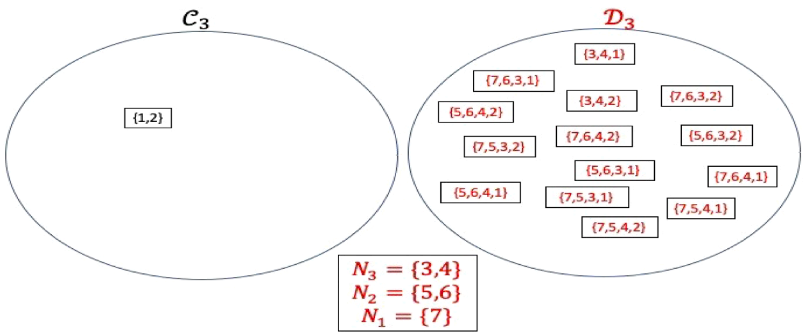

Fig. 7. Iteration 3: Updating sets $\mathcal{C}_2$ and $\mathcal{D}_2$ to $\mathcal{C}_3$ and $\mathcal{D}_3$..

To conclude, we demonstrate that the algorithm may occasionally terminate very quickly and with multiple remaining subsets. To wit, we select voter 2 as the backstop, so that $N_1 = \{2\}$. After this point, there is no subset that will satisfy the three criteria laid out in the algorithm description. Seven subsets remain, namely all of the elements of $\mathcal{C}$ containing voter 1 except subset $\{1, 2\}$. These can be ordered in an arbitrary fashion and placed in the first seven spots in the output sequence. Choosing one such ordering, we obtain the following sequence:

$$\mathsf{S}^3 = (\{7,5,3,1\}, \{5,6,3,1\}, \{7,5,4,1\}, \{5,6,4,1\}, \{7,6,3,1\}, \{3,4,1\}, \{7,6,4,1\}, \{2\}).$$

3.1.3. Algorithm 1 and its properties

We now give a formal description of Algorithm 1 and prove its properties.

Algorithm 1

Input: M-winning coalition set $\mathcal{C}$.
Output: Sequence S

1: Pick an arbitrary $i \in N$ and define $N_1 \equiv i$. Let

$$\mathcal{D}_1 = \{C \in \mathcal{C} : N_1 \subset C\}, \quad \mathcal{C}_1 = \mathcal{C} \setminus \mathcal{D}_1.$$

2: For $k = 2, 3, \ldots$, select $N_k \subset N$ such that the following three conditions hold:
(i) $|N_k| \geq 2$. (ii) $N_k \subsetneq C$ for some $C \in \mathcal{C}_{k-1}$ (iii) $N_k \cap C \neq \emptyset$ for all $C \in \mathcal{D}_{k-1}$.
If multiple subsets satisfy the above properties, choose among them arbitrarily with the following **proviso**: do **not** pick a subset N' if it is a proper superset of another subset that satisfies criteria (i)-(ii)-(iii). Define

$$\mathcal{D}_k = \mathcal{D}_{k-1} \cup \{C \in \mathcal{C}_{k-1} : N_k \subsetneq C\},$$
$$\mathcal{C}_k = \mathcal{C}_{k-1} \setminus \{C \in \mathcal{C}_{k-1} : N_k \subsetneq C\}.$$

3: If no such $N_k \subset N$ exists, STOP. Let $k^* \equiv k$ and $\mathcal{C}^* \equiv \mathcal{C}_{k^*-1}$. Suppose $\mathcal{C}^*$ has m elements and let $C_1^*, \ldots, C_m^*$ be an arbitrary ordering of them.

4: The output sequence is given by:

$$\mathsf{S} = \left(C_1^*, \ldots, C_m^*, N_{k^*-1}, N_{k^*-2}, \ldots, N_2, N_1\right).$$

Before we establish our main result, we prove a couple of technical properties related to Algorithm 1. First, we demonstrate that the sequences produced by Algorithm 1 must contain at least one element of the original M-winning coalition set. In formal terms, the set $\mathcal{C}^*$ will always be nonempty. Second, we show that the number of terms of the output sequence is strictly smaller than the cardinality of the input M-winning coalition set.

Corollary 1. *Consider an M-winning coalition rule C and any sequential unanimity rule $\mathcal{S}$, whose associated sequence S is an output of Algorithm 1 with M-winning coalition set $\mathcal{C}$ as input.*

(a) The set $\mathcal{C}^*$ is nonempty.
(b) Let $|S|$ denote the number of elements of the sequence S. Then, $|\mathcal{C}| > |S|$.

Proof. See Appendix C. □

We now proceed with the main result of this section.

Theorem 2. *Consider an M-winning coalition rule C and any sequential unanimity rule $\mathcal{S}$, whose associated sequence S is an output of Algorithm 1 with M-winning coalition set $\mathcal{C}$ as input. For all profiles $R_N \in \mathcal{R}^N$, we have $C(R_N) = \mathcal{S}(R_N)$.*

Proof. Consider a profile $R_N \in \mathcal{R}^N$ and suppose that $C \in \mathcal{C}$ is such that $R_i = x$ for all $i \in C$, for some $x \in A$. Hence, $\mathcal{C}(R_N) = x$.

Consider a sequence $\mathrm{S} = (S_1, \ldots, S_K)$ produced by Algorithm 1 with $\mathcal{C}$ as input. By construction, there exists $k \in \{1, \ldots, K\}$ such that $S_k \subset C$. We distinguish between two cases: (i) $C = S_k$ and (ii) $S_k \subsetneq C$. In case (i), it follows that $S_l \in \mathcal{C}$ for all $l = 1, \ldots, k-1$. Hence, Moulin's property implies $S_l \cap C \neq \emptyset$ for all $l = 1, \ldots, k-1$. In case (ii), for any $l = 1, \ldots, k-1$, either $S_l \in \mathcal{C}$, or $S_l \notin \mathcal{C}$. If $S_l \in \mathcal{C}$, then Moulin's property ensures $S_l \cap C \neq \emptyset$. If $S_l \notin \mathcal{C}$, then condition (iii) in Step 2 of Algorithm 1 implies $S_l \cap C \neq \emptyset$.

Thus, for all $l < k$, there exists $i_l \in S_l \cap C$. Moreover, recall that $R_i = x$ for all $i \in C \supset S_k$. Hence, $\mathcal{S}(R_N) = x$. □

Example 1. We exhibit an application of Algorithm 1 to an M-winning coalition rule that corresponds to a non-weighted constant-sum simple game. Recall the M-winning coalition set $\mathcal{C}^2$ in Section 2, i.e.

$$\mathcal{C}^2 = \{\{1,2,3\},\{1,2,4\},\{1,2,7\},\{2,3,4\},\{1,3,5,6\},\{1,3,5,7\},\{1,3,6,7\},\{1,4,5,6,8\} \\ \{2,3,5\},\{2,3,6\},\{2,3,8\},\{2,5,7\},\{2,6,7\},\{2,4,5,6\},\{3,4,7\}\},$$

which we demonstrated to correspond to a non-weighted constant-sum simple game. In applying Algorithm 1 to $\mathcal{C}^2$, we may begin by selecting voter 5 as the backstop, setting $N_1 = \{5\}$. Subsequently, the only subset satisfying the requisite three properties is $\{1,2\}$ and so we set $N_2 = \{1,2\}$. In the next iteration, the algorithm terminates. The remaining elements of $\mathcal{C}^2$ define the set $\mathcal{C}^*$:

$$\mathcal{C}^* = \big\{\{2,3,4\},\{1,3,6,7\},\{2,3,6\},\{2,3,8\},\{2,6,7\},\{3,4,7\}\big\}.$$

Ordering the elements of $\mathcal{C}^*$ arbitrarily, we obtain the following sequence:

$$\mathrm{S} = \big(\{1,3,6,7\},\{2,3,6\},\{2,3,4\},\{3,4,7\},\{2,3,8\},\{2,6,7\},\{1,2\},\{5\}\big).$$

Another application of Algorithm 1 to $\mathcal{C}^2$ begins with voter 8 as the backstop, setting $N_1 = \{8\}$. Subsequently, it selects $N_2 = \{1,3\}, N_3 = \{1,2\}, N_4 = \{2,5,6\}$. In the next (fifth) iteration, the algorithm terminates. The remaining elements of $\mathcal{C}^2$ define the set $\mathcal{C}^*$:

$$\mathcal{C}^* = \big\{\{2,3,4\},\{2,3,5\},\{2,3,6\},\{2,5,7\},\{2,6,7\},\{3,4,7\}\big\}.$$

Ordering the elements of $\mathcal{C}^*$ arbitrarily, we obtain the following sequence:

$$\mathrm{S}' = \big(\{2,3,6\},\{2,3,4\},\{3,4,7\},\{2,6,7\},\{2,3,5\},\{2,5,7\},\{2,5,6\},\{1,2\},\{1,3\},\{8\}\big).$$

The sequential unanimity rules associated with sequences S and S′ represent modest, but arguably nontrivial, ways of simplifying the constant-sum simple game $(N, \mathcal{C}^2)$.

3.1.4. Worst-case performance

An M-winning coalition rule associates to each preference profile the candidate that is unanimously preferred by at least one of the winning coalitions. However, this rule does not specify an explicit algorithm for identifying the winning coalition corresponding to a given profile. A straightforward search procedure consists of listing all coalitions in an arbitrary order and examining them sequentially until a unanimus winning coalition is found. An alternative approach, explored in the sequential-search literature (Gershkov and Szentes, 2009), is to begin with the smallest winning coalitions and proceed progressively to those of larger size. Sequential unanimity rules produced via Algorithm 1 do not necessarily search for unanimous winning coalitions in this way, but instead provide a structured procedure to determine the outcome implied by M-winning coalition rules.

Clearly, a systematic comparison of search procedures is challenging, as one procedure may terminate immediately while another may require a lot more time to determine the outcome (where time is measured in the number of coalitions whose preferences we need to check for unanimity), and the reverse may occur for a different preference profile. Moreover, the comparison of the average time it takes for the procedure to terminate across all preference profiles appears intractable. Thus, if we wish to perform a relative assessment of the two procedures we need to use a different benchmark. To this end, we adopt a worst-case perspective, based on the maximum number of coalition checks required to determine a winner across all preference profiles. Using this benchmark, the sequential unanimity rules of Algorithm 1 outperform their input M-winning coalition rules. This result is formalized in the following Corollary.

Corollary 2. *Consider an M-winning coalition rule $\mathcal{C}$ and any sequential unanimity rule $\mathcal{S}$ whose associated sequence $S = (S_1, \ldots, S_K)$ is an output of Algorithm 1 with M-winning coalition set $\mathcal{C}$ as input. Let $|\mathcal{C}| = L$ and assume an arbitrary sequence $(C_1, C_2, \ldots, C_L)$ of the elements of $\mathcal{C}$. Then,*

$$\max_{R_N \in \mathcal{R}^N} \min\{l : \ R_i = R_j \text{ for all } i,j \in C_l\} > \max_{R_N \in \mathcal{R}^N} \min\{k : \ R_i = R_j \text{ for all } i,j \in S_k\}.$$

Proof: See Appendix C. □

3.1.5. Essential sequential unanimity rules

In this section, we discuss a subfamily of sequential unanimity rules that are inspired by the proviso of Algorithm 1 and the discussion in the previous subsection.

Due to the proviso, we know that the sequences produced by Algorithm 1 will never include two subsets S_l, S_k such that $S_l \subsetneq S_k$ and $l < k$. By doing so, we exclude an obvious source of inefficiency in the implementation of the associated rule. Let us elaborate. If sequence S contained two such subsets S_l and S_k then, (i) if all voters in S_k are unanimous, then so are all voters in S_l and (ii) if there is disagreement among voters in S_l, then there is disagreements among those in S_k. Hence, subset S_k would be superfluous to the implementation of the corresponding sequential unanimity rule. More precisely, for any profile R_N the output of a sequential unanimity rule associated with S would be the same to the outcome of a similar rule in which subset S_k has been removed from the sequence.[9]

Hence, by imposing the proviso in Algorithm 1, we exclude an obvious kind of rule inefficiency due to two subsets in the sequence S containing each other. However, we do not know if the proviso guarantees efficiency in a broader sense. In particular, we do not know whether Algorithm 1 as specified will always produce sequences that contain no superfluous elements. Exploring this question is the motivation for the present subsection.

Some definitions and notation are now necessary. *Essential* sequential unanimity rules are distinguished by the fact that every term in the associated sequence $\mathrm{S} = (S_1, \ldots, S_K)$ matters. That is, if we take out any subset S_k from S and consider the resulting sequential unanimity rule, that rule is *not* identical to the original sequential unanimity rule. In other words, there are no superfluous,"dummy" subsets in the sequence S. Note that k can assume any value in $\{1, \ldots, K-1\}$. The value $k = K$ is excluded because taking out the backstop voter results in a rule that is not well-defined.

Definition 10. A sequential unanimity rule $\mathcal{S}$ associated with sequence $\mathrm{S} = (S_1, \ldots, S_K)$ is **essential** if, for all $k = 1, \ldots, K-1$, the sequential unanimity rule $\mathcal{S}_{-k}$ associated with sequence $\mathrm{S}_{-k} = (S_1, \ldots, S_{k-1}, S_{k+1}, \ldots, S_K)$ is such that there exists a profile $R_N \in \mathcal{R}^N$ such that $\mathcal{S}(R_N) \neq \mathcal{S}_{-k}(R_N)$.

Note that, given an M-winning coalition set $\mathcal{C} = \{C_1, \ldots, C_L\}$, a sequence $\mathrm{S} = (C_{j_1}, \ldots, C_{j_L}, \{i\})$, where $C_{j_1}, \ldots, C_{j_L}$ is an arbitrary ordering of $\{C_1, \ldots, C_L\}$ and $i \in N$, cannot be associated with an essential sequential unanimity rule. This is because we can safely take out from S any $C \in \mathcal{C}$ containing i and the resulting rule will be identical to the original one.

It turns out that Algorithm 1, as currently stated, may produce sequential unanimity rules that are not essential. The following example demonstrates how this can happen.

Example 2 (Recall the M-winning coalition $\mathcal{C}^2$ of Section 2:)**.**

$$\mathcal{C}^2 = \{\{1,2,3\},\{1,2,4\},\{1,2,7\},\{1,3,5,6\},\{1,3,5,7\},\{1,3,6,7\},\{1,4,5,6,8\},$$
$$\{2,3,4\},\{2,3,5\},\{2,3,6\},\{2,3,8\},\{2,5,7\},\{2,6,7\},\{2,4,5,6\},\{3,4,7\}\}.$$

Consider the following application of Algorithm 1 (for conciseness, we simply list the N_k sets at each iteration):

$$N_1 = \{8\}, N_2 = \{3,5\}, N_3 = \{3,6\}, N_4 = \{1,3\}, N_5 = \{1,2\}, N_6 = \{2,5,6\}.$$

At this point, after iteration $k = 6$, Algorithm 1 terminates.

Permuting randomly the remaining elements of $\mathcal{C}^2$, the output sequence is given by:

$$\mathrm{S} = (\{2,3,4\},\{2,5,7\},\{2,6,7\},\{3,4,7\},\{2,5,6\},\{1,2\},\{1,3\},\{3,6\},\{3,5\},\{8\}).$$

The corresponding sequential unanimity rule is denoted by $\mathcal{S}$. Now, let us drop subset $\{1,3\}$ from $\mathcal{S}$ and consider the sequential unanimity rule $\mathcal{S}_{-4}$ having sequence

$$\mathrm{S}_{-4} = (\{2,3,4\},\{2,5,7\},\{2,6,7\},\{3,4,7\},\{2,5,6\},\{1,2\},\{3,6\},\{3,5\},\{8\}).$$

We claim that $\mathcal{S}(R_N) = \mathcal{S}_{-4}(R_N)$ for all profiles $R_N \in \mathcal{R}^N$.

To see this, consider the profiles in which rule $\mathcal{S}$ produces its output exactly when it considers the preferences of agents belonging to subset $N_4 = \{1,3\}$. That is, profiles in which $R_1 = R_3$ and, for all

$$S_k \in \{\{2,3,4\},\{2,5,7\},\{2,6,7\},\{3,4,7\},\{2,5,6\},\{1,2\}\}$$

there exist $i, j \in S_k$ such that $R_i \neq R_j$. There are four categories of such profiles $R_N \in \mathcal{R}^N$:

1. Profiles such that $R_1 = R_3 = R_5 = R_6$, $R_2 = R_7 \neq R_1$ and the preferences of agents 4 and 8 are arbitrary.
2. Profiles such that $R_1 = R_3 = R_6 = R_7$, $R_2 = R_4 = R_5 \neq R_1$ and the preferences of agent 8 are arbitrary.
3. Profiles such that $R_1 = R_3 = R_5 = R_7$, $R_2 = R_4 = R_6 \neq R_1$ and the preferences of agent 8 are arbitrary.
4. Profiles such that $R_1 = R_3 = R_5 = R_6 = R_7$, $R_2 = R_4 \neq R_1$ and the preferences of agent 8 are arbitrary.

Clearly, for all profiles R_N belonging to categories 1-2-3-4, $\mathcal{S}(R_N) = R_1 = R_3$.

Now, suppose subset $\{1,3\}$ is removed from the sequence $\mathcal{S}$, leading to sequence $\mathcal{S}_{-4}$. Clearly, any profile R_N that does not belong to one of the categories 1-2-3-4 satisfies $\mathcal{S}(R_N) = \mathcal{S}_{-4}(R_N)$. For profiles R_N belonging to categories 1-2-4, $\mathcal{S}_{-4}(R_N) = R_3 = R_6$; hence, $\mathcal{S}_{-4}(R_N) = R_3 = \mathcal{S}(R_N)$. For profiles R_N belonging to category 3, $\mathcal{S}_{-4}(R_N) = R_3 = R_5$; hence, $\mathcal{S}_{-4}(R_N) = R_3 = \mathcal{S}(R_N)$.

We conclude that $\mathcal{S}(R_N) = \mathcal{S}_{-4}(R_N)$ for all profiles $R_N \in \mathcal{R}^N$.

[9] Note that this possibility is not purely theoretical. A valid application of Algorithm 1 to $\mathcal{C}^2$ of Section 3.1.3 without the proviso produces the subset selections $N_1 = \{8\}, N_2 = \{1,3,6\}, N_3 = \{1,3\}, N_4 = \{1,2\}$ and sequence

$$\mathrm{S} = \big(\{2,3,6\},\{2,3,4\},\{3,4,7\},\{2,6,7\},\{2,3,5\},\{2,5,7\},$$
$$\{2,5,6\},\{1,2\},\{1,3\},\{1,3,6\},\{8\}\big).$$

Here, the penultimate element of the sequence S is superfluous to the associated sequential unanimity rule.

The reason for the rule's failure to be essential can be traced back to the following property. At some iteration of the example, there exists a selected subset (in this case $\{1,3\}$ in iteration 4) such that *each* of the winning coalitions removed at this iteration is a superset of some subset that is selected in a later iteration. As a result, had the subset $\{1,3\}$ not been selected at iteration 4, then the unique winning coalition dropped at iteration 4 (i.e., $\{1,2,3\}$) would have been removed at the moment of subset $\{1,2\}$'s selection. Consequently, the same sequential rule would have been constructed by Algorithm 1, with the only difference that subset $\{1,3\}$ would have never been added to the output sequence.

Thus, the sequential unanimity rules produced by Algorithm 1 can fail to be essential. This shortcoming of Algorithm 1 can be corrected by adding a fourth condition to the criteria of step 2 of Algorithm 1. Before continuing, we need some further notation.

At each iteration $k = 1, 2, .$ of Algorithm 1, define the set (where $\mathcal{C}_0 \equiv \mathcal{C}$),

$$\mathcal{E}_k = \{C \in \mathcal{C}_{k-1} \ : \ N_k \subsetneq C\}.$$

Thus, $\mathcal{E}_k$ is the set of winning coalitions within $\mathcal{C}_{k-1}$ that are proper supersets of the newly selected subset N_k. As such, it coincides with the set of winning coalitions that is taken out of $\mathcal{C}_{k-1}$ at time k in order to update it to $\mathcal{C}_k$.

The additional condition to step 2 of Algorithm 1, which ensures that the produced sequential unanimity rules are essential, is as follows:

2(iv) For all $l = 2, 3, \dots k-1$ there exists $C_l^k \in \mathcal{E}_l$ such that $N_h \not\subset C_l^k$ for all $h = l+1, \dots, k$.

Clearly, the above condition 2 (iv) is only applicable for $k \geq 3$. As Algorithm 1 proceeds and k increases, it becomes harder to satisfy.

We are now ready to prove the main result of this subsection.

Proposition 3. *Consider an M-winning coalition rule $\mathcal{C}$ and any sequential unanimity rule $\mathcal{S}$, whose associated sequence S is an output of Algorithm 1 with M-winning coalition set $\mathcal{C}$ as input. The sequential unanimity rule $\mathcal{S}$ is essential if and only if condition 2(iv) is satisfied by Algorithm 1 at every iteration $k \geq 3$.*

Proof. See Appendix C. □

Thus, we have identified a necessary and sufficient condition for Algorithm 1 to produce an essential sequential unanimity rule. In light of this result, let us revisit Example 2 and see where Algorithm 1 makes a selection that fails criterion 2 (iv). To this end, it is useful to write down the subsets N_k together with the sets $\mathcal{E}_k$:

$$N_1 = \{8\}, \mathcal{E}_1 = \{\{2,3,8\}, \{1,4,5,6,8\}\}.$$
$$N_2 = \{3,5\}, \mathcal{E}_2 = \{\{2,3,5\}, \{1,3,5,6\}, \{1,3,5,7\}\}$$
$$N_3 = \{3,6\}, \mathcal{E}_3 = \{\{1,3,6,7\}, \{2,3,6\}\}$$
$$N_4 = \{1,3\}, \mathcal{E}_4 = \{\{1,2,3\}\}$$
$$N_5 = \{1,2\}, \mathcal{E}_5 = \{\{1,2,4\}, \{1,2,7\}\}$$
$$N_6 = \{2,5,6\}, \mathcal{E}_6 = \{\{2,4,5,6\}\}$$

Consider the fourth iteration of Algorithm 1 and the subset N_4. When $N_4 = \{1,3\}$ is selected, the only element of $\mathcal{C}_3$ that contains it is subset $\{1,2,3\}$. That is, $\mathcal{E}_4 = \{\{1,2,3\}\}$. Now, focus on the fifth iteration of the algorithm when subset $N_5 = \{1,2\}$ is selected. At this time, the subset N_5 is contained in $\{1,2,4\}$ and $\{1,2,7\}$, but we also have $N_5 \subsetneq \{1,2,3\}$. Thus, at iteration $k = 5$, a subset N_k is selected that leads to a violation of criterion 2 (iv): there exists $l = 4 < k = 5$ such that for all $C \in \mathcal{E}_l$ there exists $h > l$ satisfying $N_h \subsetneq C$. In this case, $\mathcal{E}_4$ is a singleton and its unique element contains N_5.

3.2. *From sequential unanimity rules to M-winning coalition rules*

In this section, we move in the reverse direction and demonstrate that a sequential unanimity rule can be transformed into an equivalent M-winning coalition rule. Our approach is once again constructive, and we refer to the algorithm that we develop and analyze as Algorithm 2. Once again, the extension of this analysis to the full preference domain is relegated to Appendix D.

3.2.1. *Algorithm 2: A simple special case*

To aid the reader, we begin by describing how Algorithm 2 works for sequential unanimity rules with associated sequences $S = (S_1, \dots, S_K)$ such that $S_l \cap S_k = \emptyset$, for all $k, l \in \{1, \dots, K\}$ where $k \neq l$. Thus, we focus on the special case in which each voter can appear at most once within the sequence S.

Algorithm 2 begins by defining a candidate winning coalition set $\mathcal{C}_1$ and initializing it to $\mathcal{C}_1 = S_1$. This implies that the sequential unanimity rule $\mathcal{S}$ and any M-winning coalition rule containing the winning coalition of $\mathcal{C}_1$ produce the same outcome for all profiles featuring unanimous agreement within subset S_1. Such profiles ensure that the sequential unanimity rule $\mathcal{S}$ reaches an outcome immediately after consulting the first element of the sequence S, i.e., S_1.

Subsequently, the algorithm considers subset S_2 and does the following. For every $i \in S_1$, it adds the subset $S_2 \cup \{i\}$ to winning coalition set $\mathcal{C}_1$, updating the latter to $\mathcal{C}_2$. This ensures that rule $\mathcal{S}$ and any M-winning coalition rule containing the winning coalitions of $\mathcal{C}_2$ produce the same outcome for all profiles featuring (i) unanimous agreement within subset S_1, or (ii) some voter disagreement

 Games and Economic Behavior 157 (2026) 263–285

within subset S_1, followed by unanimous agreement within subset S_2. These profiles are such that the rule $\mathcal{S}$ reaches an outcome after consulting at most the first two elements of sequence S.

Algorithm 2 continues iteratively in this fashion for all $k = 3, \dots, K$. That is, at each iteration k it does the following: For every $(i_1, i_2, \dots, i_{k-1}) \in S_1 \times S_2 \times \dots . S_{k-1}$, the subset $\{i_1, i_2, \dots, i_{k-1}\} \cup S_k$ is added to winning coalition set C_{k-1}, updating the latter to C_k. As a result, rule $\mathcal{S}$ and any M-winning coalition rule containing the winning coalitions of C_k produce the same outcome for all profiles featuring (i) unanimous agreement within subset S_1, or (ii) for some $l \in \{2, \dots, k\}$, some voter disagreement within each subset $S_1, S_2, \dots, S_{l-1}$, followed by unanimous agreement within subset S_l. These profiles are such that the rule $\mathcal{S}$ reaches an outcome after consulting at most the first k elements of sequence S.

Algorithm 2 terminates after iteration K, setting $C^S = C_K$. At that point, it is possible to show that the set C^S is an M-winning coalition set as per Definition 4. Moreover, the associated M-winning coalition rule C^S satisfies $C^S(R_N) = \mathcal{S}(R_N)$ for all $R_N \in \mathcal{R}_N$.

3.2.2. Paths between subsets

When the sequence S features overlapping subsets, the previous description of Algorithm 2 is very inefficient. This is because many subsets of the form $\{i_1, i_2, \dots, i_{k-1}\} \cup S_k$ will be proper supersets of other similar subsets. As a result, they will need to be removed from the final winning coalition set C^S to ensure that the latter is an M-winning coalition set. Considering that the number of subsets of the form $\{i_1, i_2, \dots, i_{k-1}\} \cup S_k$ grows exponentially in k, it would be very inefficient for our algorithm to indiscriminately add all such subsets into a candidate set of winning coalitions, only to remove a majority of them afterwards.

This concern motivates the introduction of a new concept. Given a sequence $S = (S_1, \dots, S_K)$ of subsets of N, we introduce the notion of a **path** between S_l and S_k, for $l, k \in \{1, \dots, K\}$ and $l < k$. Intuitively, a sequence of voters $(i_1, i_2, \dots)$ constitutes a **path** between S_l and S_k if it represents a parsimonious traversal of the sequence $(S_l, S_{l+1}, \dots, S_k)$. What do we mean by parsimonious? Essentially we mean that, when selecting voters sequentially from S_l to S_k, if a voter is chosen from some subset S_h, then all successive subsets of S in which that voter appears need not be considered en route to S_k. Thus, if voter $i_l \in S_l$ is selected, then it is as if all successive subsets containing voter i_l have already been visited, and can be disregarded as we make our way to from S_l to S_k. A formal definition follows.

Definition 11. Consider a sequence $(S_1, \dots, S_K)$ of non-empty subsets of N.[10] Given $l, k \in \{1, ., K\}$ such that $l < k$, a **path** p from S_l to S_k is a finite sequence $p = (i_1, i_2 \dots)$ of voters constructed via the following algorithm:

0. Let $S = (S_l, \dots, S_k)$. Let $S^0 = (S^0_1, S^0_2, \dots)$ denote the sequence obtained from S by removing all S_j such that $S_j \cap S_k \neq \emptyset$.
1. If sequence S^0 is empty (i.e., has length zero), STOP. There are no paths from S_l to S_k. Otherwise, select $i_1 \in S^0_1$. Delete all sets S_j from the sequence S^0 such that $i_1 \in S_j$. Denote the updated sequence $S^1 = (S^1_1, S^1_2, \dots)$.
2. For $h = 2, 3, \dots$
   a. If S^{h-1} is empty, STOP. The computed path is $p = (i_1, i_2, \dots, i_{h-1})$.
   b. Otherwise, pick $i_h \in S^{h-1}_1$. Delete all sets S^{h-1}_j from the sequence S^{h-1} such that $i_h \in S^{h-1}_j$. Denote the updated sequence $S^h = (S^h_1, S^h_2, \dots)$.

Given a sequence $(S_1, \dots, S_K)$ there will generally exist a number of paths between any two subsets S_l to S_k. Consider the following example.

Example 3. Suppose $S_1 = \{1, 2, 3\}, S_2 = \{6, 7\}, S_3 = \{1, 2, 4\}, S_4 = \{2, 5, 6\}, S_5 = \{2, 8\}, S_6 = \{8\}$. Let $(S_1, S_2, S_3, S_4, S_5, S_6)$ and suppose we want to identify all paths from S_1 to S_6. Applying Definition 11, we calculate 16 such paths from S_1 to S_6. Here they are: $(1,6), (1,7,2), (1,7,5), (1,7,6), (2,6), (2,7), (3,6,1), (3,6,2), (3,6,4), (3,7,1,2), (3,7,1,5), (3,7,1,6), (3,7,2), (3,7,4,2), (3,7,4,5), (3,7,4,6)$.

Fig. 8 illustrates the computation of path $(1,7,2)$. We initialize the process by deleting all elements of sequence S that have a non-empty intersection with S_6, i.e., that contain the voter 8. Thus, S_5 and S_6 are deleted and the resulting sequence is denoted $S^0 = (S^0_1, S^0_2, S^0_3, S^0_4) = (\{1,2,3\}, \{6,7\}, \{1,2,4\}, \{2,5,6\})$. In iteration 1, we select $i_1 = 1 \in S^0_1$. Subsequently, we delete all the terms of sequence S^0 that contain voter 1. This means we delete sets S^0_1, S^0_3. The resulting sequence is denoted $S^1 = (S^1_1, S^1_2) = (\{6,7\}, \{2,5,6\})$. In iteration 2, we select $i_2 = 7 \in S^1_1$. Subsequently, we delete all the terms of sequence S^1 that contain voter 7. This means we delete set S^1_1. The resulting sequence is denoted $S^2 = (S^2_1) = (\{2,5,6\})$. In iteration 3, we select voter $i_3 = 2 \in S^2_1$. Subsequently, we delete set S^2_1 and obtain an empty sequence, which we denote by S^3. In iteration 4, the algorithm terminates with the computed path $(i_1, i_2, i_3) = (1,7,2)$.

3.2.3. General description of Algorithm 2

At a high level, Algorithm 2 works in the following way. Given a sequence of subsets $S = (S_1, \dots, S_K)$, it begins by putting S_1 into a candidate winning coalition set C_1. At each subsequent iteration, the algorithm adds further elements to this set, each time updating it. Namely, at iteration $k = 2, \dots, K$, Algorithm 2 considers subset S_k of sequence S, and does the following:

(i) It computes all paths from S_1 to S_k; and
(ii) For each path p from S_1 to S_k, the subset $\{p\} \cup S_k$ is added to the candidate winning coalition set. We denote the *collection* of all such subsets by C^p_k, i.e. $C^p_k = \bigcup_p \left\{\{p\} \cup S_k\right\}$. Correspondingly, set C_{k-1} is updated to $C_k = C_{k-1} \cup C^p_k$. If necessary, set C_k is pruned in order to get rid of elements which are proper supersets of others.

[10] Note that, unlike in the sequences used for sequential unanimity rules, we make no restrictions on S.

 Games and Economic Behavior 157 (2026) 263–285

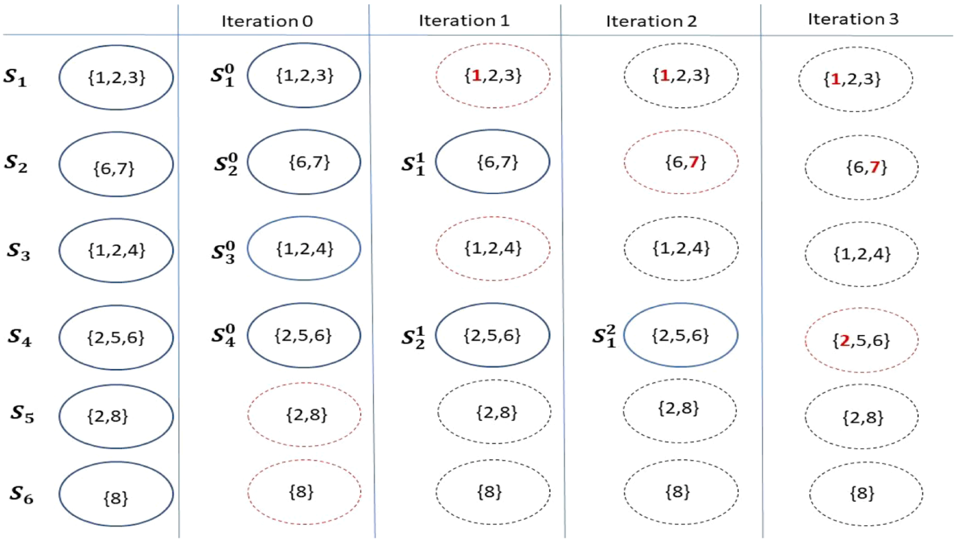

Fig. 8. Example 3: Computation of path (1,7,2). Deleted subsets appear in dashed circles. Newly deleted subsets appear in red dashed circles. Selected voters are indicated in red, bold fonts. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

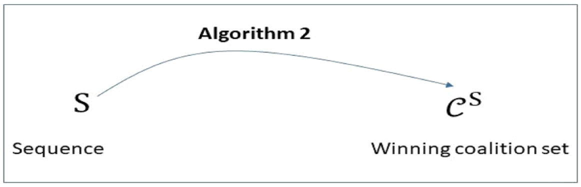

Fig. 9. Algorithm 2 associates to each sequence S a winning coalition set $\mathcal{C}^{\mathrm{S}}$.

After the last iteration $k = K$, the candidate winning coalition set is $\mathcal{C}_K$. At that point, the algorithm terminates and produces as output the M-winning coalition set $\mathcal{C}^{\mathrm{S}} = \mathcal{C}_K$.

At each iteration k, rule $\mathcal{S}$ and any M-winning coalition rule containing the winning coalitions in $\mathcal{C}_k$ produce the same outcome for all profiles featuring (i) unanimous agreement within subset S_1, or, (ii) for some $l \in \{2, \dots, k-1\}$ some voter disagreement within each subset $S_1, S_2, \dots S_{l-1}$, followed by unanimous agreement within subset S_l. These profiles are such that the sequential unanimity rule $\mathcal{S}$ reaches an outcome after consulting at most the first k elements of sequence S.

We will show two things: (i) if $\mathrm{S} = (S_1, \dots, S_K)$ is such that $|S_K| = 1$, then the winning coalition set $\mathcal{C}^{\mathrm{S}}$ produced by Algorithm 2 is an M-winning coalition set as per Definition 4, and (ii) the sequential unanimity rule $\mathcal{S}$ with associated sequence S, is equivalent to the M-winning coalition rule $\mathcal{C}^{\mathrm{S}}$. Before proceeding with these results, we illustrate Algorithm 2 on the usual job-market candidate example.

3.2.4. An illustrative example

Recall the winning coalition set $\mathcal{C}$ depicted in Fig. 2. In the previous section, we showed that by running Algorithm 1 with $\mathcal{C}$ as input in two different ways we obtained the sequences

$$\mathrm{S}^1 = (\{1,2\}, \{3,4\}, \{5,6\}, \{7\})$$

and

$$\mathrm{S}^2 = (\{1,2\}, \{3,5,7\}, \{3,6,7\}, \{3,5,6\}, \{4\}).$$

By Theorem 2 the respective sequential unanimity rules are equivalent to the M-winning coalition rule $\mathcal{C}$. For the results of this section to be consistent with Theorem 2, when applying Algorithm 2 to sequences S^1 and S^2 we need to obtain the same M-winning coalition set $\mathcal{C}$ of Fig. 2. We thus verify that this is the case.

First, consider S^1. Since this sequence features completely non-overlapping subsets, path calculations are very easy. Indeed, given $k \in \{2,3,4\}$, the set of paths from S_1 to S_k will consist of all sequences $(i_1, \dots, i_{k-1})$ that satisfy $i_1 \in S_1, \dots, i_{k-1} \in S_{k-1}$. The corresponding additions to the candidate winning coalition set are straightforward. Finally, after iteration K, we verify that $\mathcal{C}_K = \mathcal{C}^S$ equals the M-winning coalition set of Fig. 2.

		$k=1$	$k=2$	$k=3$	$k=4$
S_1	{1,2}	{1,2}	{1,2}	{1,2}	{1,2}
S_2	{3,4}	{3,4}	{3,4}	{3,4}	{3,4}
S_3	{5,6}	{5,6}	{5,6}	{5,6}	{5,6}
S_4	{7}	{7}	{7}	{7}	{7}
Paths from S_1 to S_k		∅	(1), (2)	(1,3), (1,4), (2,3), (2,4)	(1,3,5), (1,3,6), (1,4,5), (1,4,6), (2,3,5), (2,3,6), (2,4,5), (2,4,6)
Additions to candidate coalition set (i.e., set $\mathcal{C}_k^p$).		{1,2}	{1,3,4}, {2,3,4}	{1,3,5,6}, {1,4,5,6}, {2,3,5,6}, {2,4,5,6}	{1,3,5,7}, {1,3,6,7}, {1,4,5,7}, {1,4,6,7}, {2,3,5,7}, {2,3,6,7}, {2,4,5,7}, {2,4,6,7},

Fig. 10. Applying Algorithm 2 to $S^1 = (\{1,2\},\{3,4\},\{5,6\},\{7\})$. We verify that $C_4 = C^S$ equals the M-winning coalition set of Fig. 2.

		$k=1$	$k=2$	$k=3$	$k=4$	$k=5$
S_1	{1,2}	{1,2}	{1,2}	{1,2}	{1,2}	{1,2}
S_2	{3,5,7}	{3,5,7}	{3,5,7}	{3,5,7}	{3,5,7}	{3,5,7}
S_3	{3,6,7}	{3,6,7}	{3,6,7}	{3,6,7}	{3,6,7}	{3,6,7}
S_4	{3,5,6}	{3,5,6}	{3,5,6}	{3,5,6}	{3,5,6}	{3,5,6}
S_5	{4}	{4}	{4}	{4}	{4}	{4}
Paths from S_1 to S_k		∅	(1), (2)	(1), (2)	(1), (2)	(1,3), (2,3), (1,5,3), (1,5,6),(1,5,7) (2,5,3), (2,5,6),(2,5,7), (1,7,3), (1,7,5),(1,7,6) (2,7,3), (2,7,5),(2,7,6),
Additions to candidate coalition set (i.e., set $\mathcal{C}_k^p$).		{1,2}	{1,3,5,7}, {2,3,5,7}	{1,3,6,7}, {2,3,6,7}	{1,3,5,6}, {2,3,5,6}	{1,3,4}, {2,3,4}, ~~{1,5,3,4}~~,{1,5,6,4},{1,5,7,4}, ~~{2,5,3,4}~~,{2,5,6,4},{2,5,7,4}, ~~{1,7,3,4}~~,~~{1,7,5,4}~~,{1,7,6,4}, ~~{2,7,3,4}~~~~{2,7,5,4}~~,{2,7,6,4}

Fig. 11. Applying Algorithm 2 to $S^2 = (\{1,2\},\{3,5,7\},\{3,6,7\},\{3,5,6\},\{4\})$. Subsets which are removed during the pruning phase of the Algorithm, because they are strict supersets of others, are crossed out (e.g. $\{1,5,3,4\} \supsetneq \{1,3,4\}$). Duplicate subsets are crossed out as well (e.g. $\{2,7,5,4\}$ is identical to $\{2,5,7,4\}$). The resulting set $C_5 = C^S$ equals the M-winning coalition set of Fig. 2.

The application of Algorithm 2 is a little more involved for S^2. Here, there is overlap between subsets S_2, S_3, S_4 so that path calculations are a little trickier. Fig. 11 specifies at each iteration of the algorithm, the paths and additions to the candidate coalition set. At iteration 5, four subsets need to be removed from C_K because they are proper supersets of others.

3.2.5. *Algorithm 2 and its properties*

In this section we formally define Algorithm 2 and prove its desirable properties.

Lemma 1. *Let $S = (S_1, \ldots, S_K)$ be a sequence of subsets of N such that $|S_K| = 1$. The winning coalition set C^S computed by Algorithm 2 is an M-winning coalition set.*

Proof. See Appendix C. □

Theorem 3. *Consider a sequential unanimity rule $\mathcal{S}$ parametrized by $S = (S_1, \ldots, S_K)$. Suppose Algorithm 2 is applied to sequence S, producing the M-winning coalition set C^S, and consider the associated M-winning coalition rule $\mathcal{C}^S$. For all profiles $R_N \in \mathcal{R}^N$, we have $\mathcal{S}(R_N) = \mathcal{C}^S(R_N)$.*

Algorithm 2

Input: Sequence S $= (S_1, \ldots, S_K)$.
Output: winning coalition set $\mathcal{C}^{\mathrm{S}}$
1: Let $\mathcal{C}_1 = S_1$.
2: For $k = 2, \ldots, K$
  a. Consider the set S_k and the sequence $(S_1, \ldots., S_k)$.
  b. For every path p from S_1 to S_k, define $C_p \subset N$ such that $C_p = \{p\} \cup S_k$. Let $\mathcal{C}_k^p = \bigcup_p \{\{p\} \cup S_k\}$ be the collection of all such subsets C_p. If no paths from S_1 to S_k exist, set $\mathcal{C}_k^p = S_k$.
  c. Define $\mathcal{C}_k = \mathcal{C}_{k-1} \cup \mathcal{C}_k^p$. Update $\mathcal{C}_k$ by deleting all $\tilde{C} \in \mathcal{C}_k$ satisfying $\tilde{C} \supsetneq C$ for some $C \in \mathcal{C}_k$.
3: Define $\mathcal{C}^S \equiv \mathcal{C}_K$.

Proof. Consider a profile R_N. Suppose that at this profile set S_k is decisive. That is, (i) there exists $x \in A$ such that $R_i = x$ for all $i \in S_k$ and (ii) for all $l < k$, there exist $i, j, \in S_l$ such that $R_i \neq R_j$. Then $\mathcal{S}(R_N) = x$. We will show that $\mathcal{C}^S(R_N) = x$.

By Lemma 1, $\mathcal{C}^S$ is an M-winning coalition set. Thus, we need to prove that there exists $C \in \mathcal{C}^S$ such that $R_i = x$ for all $i \in C$. Recall k, the index of the decisive set. If $k = 1$, we immediately conclude $\mathcal{C}^S(R_N) = x$, since $S_1 \in \mathcal{C}^S$. If $k > 1$, then for all $l < k$, there exists $i_l \in S_l$ such that $R_i = x$. Therefore, there exists a path p^* from S_1 to S_k that is a subsequence of $(i_1, \ldots, i_{l-1})$. The corresponding set $S_k \cup \{p^*\}$ will either belong to $\mathcal{C}^S$, or step 2 iii of the algorithm will ensure that a strict subset of it does. Either way, $\mathcal{C}^S$ will include a set C' such that $R_i = x$ for all $i \in C'$. Thus, $\mathcal{C}^S(R_N) = x$. □

Corollary 3. *Suppose sequence S is obtained by applying Algorithm 1 on M-winning coalition set $\mathcal{C}$. If we apply Algorithm 2 to S, we obtain $\mathcal{C}^{\mathrm{S}} = \mathcal{C}$.*

Proof. By Theorem 2, the M-winning coalition rule $\mathcal{C}$ is equivalent to the sequential unanimity rule $\mathcal{S}$ associated with S. Theorem 3 implies that when applying Algorithm 2 to S, we obtain a M-winning coalition set $\mathcal{C}^{\mathrm{S}}$ whose associated M-winning coalition rule is equivalent to $\mathcal{C}$. Since a given neutral and SP rule is uniquely defined by its M-winning coalition set, we conclude $\mathcal{C}^{\mathrm{S}} = \mathcal{C}$. □

4. Conclusion

In this paper we introduced sequential unanimity rules for voting between two alternatives. Via a constructive approach, we demonstrated how these rules facilitate the implementation of the well-known M-winning coalition rules of Moulin (1983). We did so by developing algorithms that convert a given M-winning coalition rule into an equivalent sequential unanimity rule and vice versa. The sequential unanimity rules that are constructed present computational advantages compared to their M-winning coalition rule counterparts. Since constant-sum simple games are formally equivalent to M-winning coalition rules, these results are potentially of interest to the literature on simple games as well. Finally, we extended the reach of our analysis to cover the full preference domain which allows for indifferences.

Data availability

No data was used for the research described in the article.

Declaration of competing interest

none

Acknowledgments

The authors would like to thank Carlos Alos-Ferrer, Herve Moulin, and seminar participants at the University of Milan-Bicocca and the Lausanne Theory meeting for useful comments. Bonkoungou acknowledges financial support from the Swiss National Science Foundation, project number 100018-207722.

Appendix A. Proof that $\mathcal{C}^2$ is a valid M-winning coalition set

Recall that:

$$\mathcal{C}^2 = \{\{1,2,3\},\{1,2,4\},\{1,2,7\},\{1,3,5,6\},\{1,3,5,7\},\{1,3,6,7\},\{1,4,5,6,8\},\{2,3,4\},\{2,3,5\},\{2,3,6\},\{2,3,8\},\{2,5,7\},\{2,6,7\},\{2,4,5,6\},\{3,4,7\}\}.$$

It is not hard, though slightly time-consuming, to verify Minimality. For instance, consider the largest-cardinality element of $\mathcal{C}^2$, namely $\{1,4,5,6,8\}$. If we remove any of its elements, then the resulting version of $\mathcal{C}^2$ will violate Moulin's property: if we remove 1, then $\{4,5,6,8\} \cap \{1,2,3\} = \emptyset$; if we remove 4, then $\{1,5,6,8\} \cap \{3,4,7\} = \emptyset$; if we remove 5, then $\{1,4,6,8\} \cap \{2,5,7\} = \emptyset$; if we remove 6, then $\{1,4,5,8\} \cap \{2,3,6\} = \emptyset$; if we remove 8, then $\{1,4,5,6\} \cap \{2,3,8\} = \emptyset$. We can repeat a similar exercise for all the other elements of $\mathcal{C}^2$ and verify that replacing them with any one of their strict subsets violates Moulin's property.

As regards Moulin's property, the argument is a little more involved. First, notice that all $C', C'' \in \mathcal{C}^2$ are such that $C' \cap C'' \neq \emptyset$. Let $C \subset N$ be such that $C \supset C'$ for some $C' \in \mathcal{C}^2$. Since $C' \cap C'' \neq \emptyset$ for all $C'' \in \mathcal{C}^2$, we conclude that $C \cap C'' \neq \emptyset$ for all $C'' \in \mathcal{C}^2$. This establishes one direction of the equivalence stipulated in Moulin's property.

Let us now prove the other direction. Suppose $C \cap C'' \neq \emptyset$ for all $C'' \in \mathcal{C}^2$. We need to show that $C \supset C'$ for some $C' \in \mathcal{C}^2$. Let us consider four mutually exclusive and exhaustive cases and prove the result in each case separately.

(i) C does not contain 1 or 2. Since $C \cap \{1,2,3\} \neq \emptyset, C \cap \{1,2,4\} \neq \emptyset, C \cap \{1,2,7\} \neq \emptyset$, we must have $C \supset \{3,4,7\} \in \mathcal{C}^2$.

(ii) C contains both 1 and 2. Since $C \cap \{3,4,7\} \neq \emptyset$, we must have $3 \in C$ or $4 \in C$ or $7 \in C$. Thus, $C \supset \{1,2,3\} \in \mathcal{C}^2$ or $C \supset \{1,2,4\} \in \mathcal{C}^2$ or $C \supset \{1,2,7\} \in \mathcal{C}^2$.

(iii) C contains 1 but not 2. Since $C \cap \{3,4,7\} \neq \emptyset$, we must have $3 \in C$ or $4 \in C$ or $7 \in C$. If $3 \in C, 4 \in C$ and $7 \in C$, then $C \supset \{3,4,7\} \in \mathcal{C}^2$. If $3 \in C, 4 \notin C, 7 \notin C$, then $C \cap \{2,5,7\} \neq \emptyset$ and $C \cap \{2,6,7\} \neq \emptyset$ imply that $5 \in C$ and $6 \in C$. Hence $C \supset \{1,3,5,6\} \in \mathcal{C}^2$. If $3,4 \in C$ and $7 \notin C$, then for identical reasons as before $5 \in C$ and $6 \in C$. Hence $C \supset \{1,3,4,5,6\} \supset \{1,3,5,6\} \in \mathcal{C}^2$. If $3,7 \in C$ and $4 \notin C$, then $C \cap \{2,4,5,6\} \neq \emptyset$ implies that $5 \in C$ or $6 \in C$. Thus $C \supset \{1,3,5,7\} \in \mathcal{C}^2$ or $C \supset \{1,3,6,7\} \in \mathcal{C}^2$. Finally, if $3 \notin C$, then $C \cap \{2,3,x\} \neq \emptyset$ for $x = 4,5,6,8$ implies that $x \in C$ for all $x = 4,5,6,8$. As a result, $C \supset \{1,4,5,6,8\} \in \mathcal{C}^2$.

(iv) C contains 2 but not 1. Since $C \cap \{3,4,7\} \neq \emptyset$, we must have $3 \in C$ or $4 \in C$ or $7 \in C$. If $3 \in C, 4 \in C$ and $7 \in C$, then $C \supset \{3,4,7\} \in \mathcal{C}^2$. If $3 \in C, 4 \notin C, 7 \notin C$, then $C \cap \{1,4,5,6,8\} \neq \emptyset$ implies that $C \supset \{2,3,5\} \in \mathcal{C}^2$ or $C \supset \{2,3,6\} \in \mathcal{C}^2$ or $C \supset \{2,3,8\} \in \mathcal{C}^2$. If $3,4 \in C$ but $7 \notin C$, then we get $C \supset \{2,3,4\} \in \mathcal{C}^2$. If $3,7 \in C$ and $4 \notin C$, then $C \cap \{1,4,5,6,8\} \neq \emptyset$ implies that $5 \in C$ or $6 \in C$ or $8 \in C$. Thus $C \supset \{2,3,7,5\} \supset \{2,5,7\} \in \mathcal{C}^2$ or $C \supset \{2,3,7,6\} \supset \{2,6,7\} \in \mathcal{C}^2$ or $C \supset \{2,3,7,8\} \supset \{2,3,8\} \in \mathcal{C}^2$. Finally, if $3 \notin C$, then $C \cap \{3,4,7\} \neq \emptyset$ implies that either $4 \in C$ or $7 \in C$. If $4 \in C$ then $C \cap \{1,3,5,6\} \neq \emptyset, C \cap \{1,3,5,7\} \neq \emptyset, C \cap \{1,3,6,7\} \neq \emptyset$ together imply that (i) either $5,6 \in C$, in which case we get $C \supset \{2,4,5,6\} \in \mathcal{C}^2$, or (ii) $6,7 \in C$, in which case $C \supset \{2,4,6,7\} \supset \{2,6,7\} \in \mathcal{C}^2$ or (iii) $5,7 \in C$, in which case $C \supset \{2,4,5,7\} \supset \{2,5,7\} \in \mathcal{C}^2$. If $7 \in C$, then $C \cap \{1,3,5,6\} \neq \emptyset$ implies that $5 \in C$ or $6 \in C$. In the former case $C \supset \{2,5,7\} \in \mathcal{C}^2$, while in the latter $C \supset \{2,6,7\} \in \mathcal{C}^2$.

□

Appendix B. Computation of sequences S^2 and S^3 of Section 3.1.2

In this section, we demonstrate how Algorithm 1 produces sequences S^2 and S^3 cited in the main text.

We first address sequence S^2. We begin by selecting voter 4 as the backstop, so that $N_1 = \{4\}$. Subsequently, we update the M-winning coalition set by removing all its elements containing voter 4. Fig. B.1 illustrates.

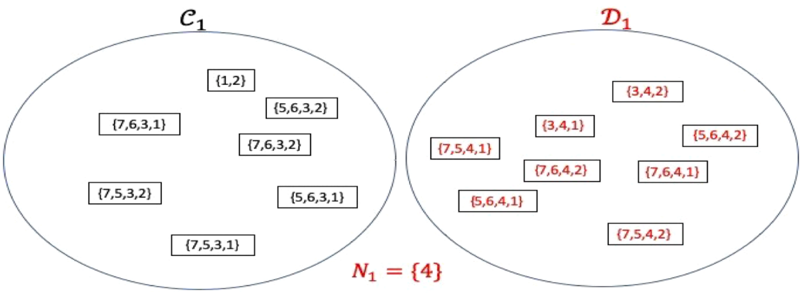

Fig. B.1. Iteration 1: Set backstop voter $i = 4$ and remove all elements of $\mathcal{C}$ containing it. The remaining M-winning coalition set is $\mathcal{C}_1$ and the discarded M-winning coalition set is $\mathcal{D}_1$..

We continue by searching for a subset of voters N_2 that satisfies the three criteria laid out in the previous subsection. There are various choices we could make at this point and we choose $N_2 = \{3,5,6\}$. Fig. B.2 highlights the elements of $\mathcal{C}_1$ that are proper supersets of N_2 and demonstrates why subset $\{3,5,6\}$ is a is a valid choice. For conciseness, we omit explicitly displaying the corresponding updates of the remaining M-winning coalition set $\mathcal{C}_1$ and the discarded M-winning coalition set $\mathcal{D}_1$, to $\mathcal{C}_2$ and $\mathcal{D}_2$ respectively.

We continue by choosing $N_3 = \{3,6,7\}$. Fig. B.3 illustrates why this is a valid selection.

We continue by choosing $N_4 = \{3,5,7\}$. Once again, Fig. B.4 illustrates why this is a valid selection.

After iteration 4, there is a single element left in the remaining M-winning coalition set $\mathcal{C}_4$. Moreover, there is no way to satisfy all three criteria: in particular, criteria (i) and (ii) are mutually exclusive. Thus, Algorithm 1 terminates. The output sequence that it produces is the following:

$$S^2 = (\{1,2\}, N_4, N_3, N_2, N_1) = (\{1,2\}, \{3,5,7\}, \{3,6,7\}, \{3,5,6\}, \{4\}).$$

Finally, we show what happens when the algorithm terminates with multiple remaining subsets. To make this point as stark as possible, we select voter 2 as the backstop voter, so that $N_1 = \{2\}$. Subsequently, we remove all elements of $\mathcal{C}$ that contain voter 2, and obtain $\mathcal{C}_1$ and $\mathcal{D}_1$ as depicted in Fig. B.5.

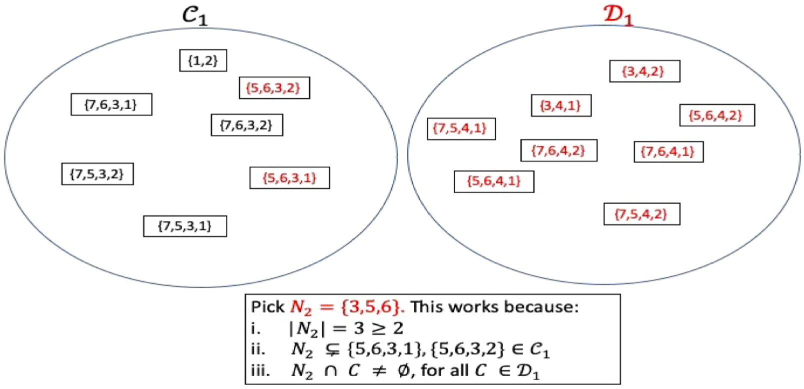

Fig. B.2. Iteration 2: Selecting subset N_2..

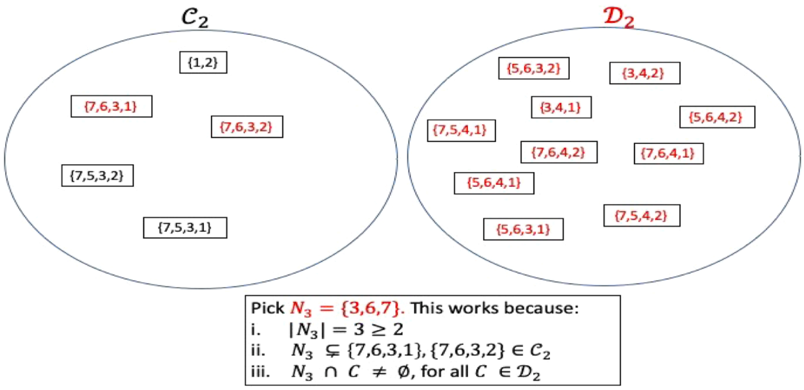

Fig. B.3. Iteration 3: Selecting subset N_3..

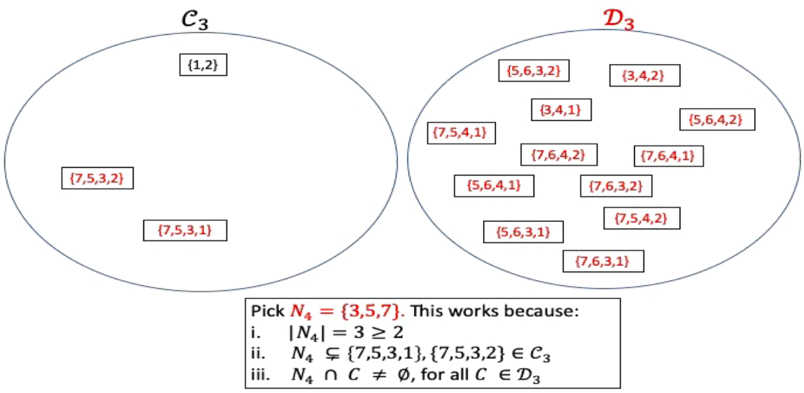

Fig. B.4. Iteration 4: Selecting subset N_4..

At this point there is no subset of N that satisfies criteria (i)-(ii)-(iii). For example, subset $\{5,6\}$ satisfies (i) and (ii) but fails (iii): we have $\{1,2\} \in \mathcal{D}_1$ and $\{5,6\} \cap \{1,2\} = \emptyset$. Or, similarly, subset $\{1,3,6\}$ satisfies (i) and (ii) but fails (iii): we have $\{7,5,4,2\} \in \mathcal{D}_1$ and $\{1,3,6\} \cap \{7,5,4,2\} = \emptyset$. Thus, the algorithm terminates, with seven elements of C_1 still remaining. These seven subsets can be ordered in an arbitrary fashion and placed in the first seven spots in the output sequence. Choosing one such ordering, we obtain the following sequence:

$$S^3 = (\{7,5,3,1\}, \{5,6,3,1\}, \{7,5,4,1\}, \{5,6,4,1\}, \{7,6,3,1\}, \{3,4,1\}, \{7,6,4,1\}, \{2\}).$$

Appendix C. Omitted proofs

Proposition 1. Minimality is satisfied by the definition of $\mathcal{W}^m$. So let us prove that $\mathcal{W}^m$ satisfies Moulin's property.

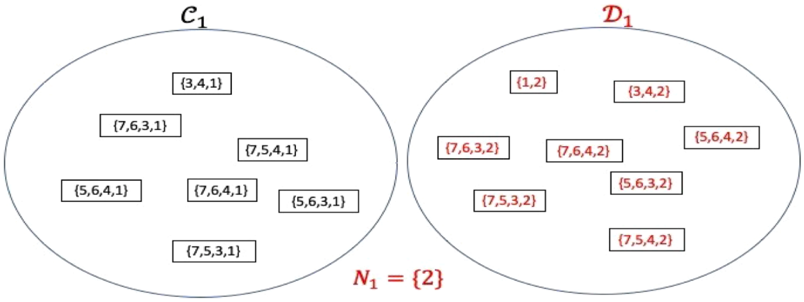

Fig. B.5. Iteration 2: After choosing voter 2 as the backstop, no subset N_2 satisfies the three criteria.

"⇒" Let $C \subset N$ and suppose that $C \supset C'$ for some $C' \in \mathcal{W}^m$. Consider $C_1, C_2 \in \mathcal{W}^m$ and assume $C_1 \cap C_2 = \emptyset$. Then, $C_2 \subset N \setminus C_1$, which by monotonicity implies $N \setminus C_1 \in \mathcal{W}$. But, since the simple game is constant-sum, $C_1 \in \mathcal{W} \Rightarrow N \setminus C_1 \notin \mathcal{W}$. This is a contradiction and we conclude $C_1 \cap C_2 \neq \emptyset$ for all $C_1, C_2 \in \mathcal{W}^m$. Thus, $C' \cap C'' \neq \emptyset$ for all $C'' \in \mathcal{W}^m$. As $C \supset C'$, this yields $C \cap C'' \neq \emptyset$ for all $C'' \in \mathcal{W}^m$.

"⇐" Let $C \subset N$ and suppose that $C \cap C'' \neq \emptyset$ for all $C'' \in \mathcal{W}^m$. Consider the following two cases.

1. If $C \in \mathcal{W}$, then there exists $C' \in \mathcal{W}^m$ such that $C \supset C'$, and we are done.
2. If $C \notin \mathcal{W}$, then since the game is constant-sum, $N \setminus C \in \mathcal{W}$. We again distinguish between two sub-cases: (i) If $N \setminus C \in \mathcal{W}^m$, then $C \cap (N \setminus C) = \emptyset$ contradicts $C \cap C'' \neq \emptyset$ for all $C'' \in \mathcal{W}^m$; (ii) If $N \setminus C \notin \mathcal{W}^m$, then there exists $C''' \in \mathcal{W}^m$ such that $C''' \subsetneq N \setminus C$. However $C \cap C''' = \emptyset$, again in contradiction to $C \cap C'' \neq \emptyset$ for all $C'' \in \mathcal{W}^m$. As a result, $C \in \mathcal{W}$ and the proof is complete. □

Proposition 2. It is easy to show that sequential unanimity rules are neutral. To establish their strategy-proofness, consider a sequential unanimity rule $\mathcal{S}$ with associated sequence $\mathrm{S} = (S_1, \ldots, S_K)$. Suppose that $\mathcal{S}(R_N) = x \in A$ and that subset S_l is the first element of sequence S to feature unanimous agreement. Now consider a voter $i \in N$ and suppose that $R_i \neq x$. This implies that $i \notin S_l$. Suppose this voter misreports his preferences, stating $R_i' = x$. We distinguish between two cases. If $i \notin S_k$ for all $k < l$, then voter i's preferences are not consulted before the rule reaches an outcome, implying $\mathcal{S}(R_i', R_{-i}) = \mathcal{S}(R_N) = x$. Conversely, let $\mathcal{K} = \{k < l : i \in S_k\}$. If there exists $k \in \mathcal{K}$ such that $R_j = x$ for all $j \in S_k$ such that $j \neq i$, then $\mathcal{S}(R_i', R_{-i}) = x$. If not, then S_l will once again be the first element of sequence S to feature unanimous agreement, implying $\mathcal{S}(R_i', R_{-i}) = x$. We conclude that $\mathcal{S}$ is strategy-proof. □

Corollary 1. We start with part (a). Suppose $\mathcal{C}^*$ is empty. Then, there exists k such that N_k satisfies conditions 2 (i)-(ii)-(iii) and such that $\mathcal{C}_k = \emptyset$. Thus, there exists $C \subset N$ such that $C \subsetneq C'$ for some $C' \in \mathcal{C}$ and $C \cap C'' \neq \emptyset$ for all $C'' \in \mathcal{C}$. By Moulin's property, there exists $C''' \in \mathcal{C}$ such that $C \supset C'''$. Since $C \subsetneq C'$ for some $C' \in \mathcal{C}$, we conclude that there exist $C', C''' \in \mathcal{C}$ such that $C' \subsetneq C'''$. This contradicts the Minimality of $\mathcal{C}$.

Now we prove part (b). We claim that the backstop voter must belong to at least two elements of $\mathcal{C}$, that is $|\mathcal{D}_1| \geq 2$. To derive a contradiction suppose that $|\mathcal{D}_1| = 1$. Let $S_K = i$ and $C \in \mathcal{D}_1$. Since $\mathcal{C}$ is an M-winning coalition set and $|\mathcal{D}_1| = 1$, $(C \setminus \{i\}) \cap C'' \neq \emptyset$ for all $C'' \in \mathcal{C}$. By Moulin's property, $C \setminus \{i\} \supset C'$ for some $C' \in \mathcal{C}$. Thus, there exist $C, C' \in \mathcal{C}$ such that $C' \subsetneq C$. This contradicts the Minimality of $\mathcal{C}$. Therefore, $|\mathcal{D}_1| \geq 2$. Consequently,

$$|\mathrm{S}| \leq 1 + |\mathcal{C}| - |\mathcal{D}_1| \leq |\mathcal{C}| - 1.$$

The upper bound is attained if and only if $|\mathcal{D}_1| = 2$ and $k^* = 2$. □

Corollary 2. Let R_N be a profile such that $R_i = a$ for all $i \in C_L$ and $R_i = b$ for all $i \notin C_L$. Since $\mathcal{C}$ is an M-winning coalition set, $C_l \not\subset C_L$ and $C_l \cap C_L \neq \emptyset$ for all $l = 1, 2, \ldots, L-1$. Hence

$$\min\{l : R_i = R_j \text{ for all } i, j \in C_l\} = L.$$

Since $|\mathcal{C}| = L$, the above implies

$$\max_{R_N \in \mathcal{R}^N} \min\{l : R_i = R_j \text{ for all } i, j \in C_l\} = L. \tag{C.1}$$

Conversely,

$$|\mathrm{S}| = K \geq \max_{R_N \in \mathcal{R}^N} \min\{k : R_i = R_j \text{ for all } i, j \in S_k\}. \tag{C.2}$$

By Corollary 1, $L > K$. This fact, together with Eq. (C.1)–(C.2), implies:

$$\max_{R_N \in \mathcal{R}^N} \min\{l : R_i = R_j \text{ for all } i, j \in C_l\} > \max_{R_N \in \mathcal{R}^N} \min\{k : R_i = R_j \text{ for all } i, j \in S_k\}.$$

□

Proposition 3. Before we begin, it is useful to make a general observation. Suppose Algorithm 1 is implemented with condition 2 (iv) and produces the sequence $S = (S_1, S_2, \ldots, S_K)$. For $k = 1, \ldots, K$ define:

$$\mathcal{L}_k = \begin{cases} S_k, & \text{if } S_k \in \mathcal{C} \\ \mathcal{E}_{l(k)}, & \text{if} l(k) = \{j : N_j = S_k\} \text{ is well-defined.} \end{cases}$$

Thus, $\mathcal{L}_k$ is either equal to S_k if S_k is an element of $\mathcal{C}$, or it is equal to the set of winning coalitions that strictly contained S_k when it was selected by step 2 of Algorithm 1. For example, Corollary 1 implies that $\mathcal{L}_1 = S_1$, while $\mathcal{L}_K$ equals the set of elements of $\mathcal{C}$ that contain the backstop voter S_K (i.e., $\mathcal{L}_K = \mathcal{D}_1$).

The following Lemma will prove useful later on.

Lemma 2. *Suppose Algorithm 1 is implemented with condition 2 (iv) and produces the sequence $S = (S_1, S_2, \ldots, S_K)$ with $\mathcal{C}$ as input. For all $k \in \{1, \ldots, K-1\}$, there exists $C_k \in \mathcal{L}_k$ such that $S_k \subset C_k$ and $S_l \not\subset C_k$ for all $l \neq k$.*

Proof: Let $k \in \{1, \ldots, K-1\}$ and $l \neq k$.

If $\mathcal{L}_k = S_k$, then let $C_k = S_k$. Let $l < k$. By the construction of Algorithm 1, $S_l \in \mathcal{C}$. Since $\mathcal{C}$ is an M-winning coalition set, $S_l \not\subset C_k$. Let $l > k$. If $S_l \subsetneq C_k$, then $C_k \in \mathcal{L}_l$, which is a contradiction. Thus, $C_k \in \mathcal{L}_k$ is such that $S_l \not\subset C_k$ for all $l \neq k$.

If $\mathcal{L}_k \neq S_k$, then condition 2 (iv) implies that there exists $C_k \in \mathcal{L}_k$ such that $S_l \not\subset C_k$ for all $l < k$. Now consider $l > k$. If $S_l \subsetneq C_k$, then $C_k \in \mathcal{L}_l$, which is a contradiction. Thus, $C_k \in \mathcal{L}_k$ is such that $S_l \not\subset C_k$ for all $l \neq k$. □

We proceed with the proof of the Proposition.

"⇒" First, we prove sufficiency. Let $\mathrm{S} = (S_1, \ldots, S_K)$ be an output of Algorithm 1 with $\mathcal{C}$ as input. We assume that $K > 1$ and that, at each iteration of Algorithm 1, condition 2 (iv) is satisfied. Let $\mathcal{S}$ denote the corresponding sequential unanimity rule. Let $k = 1, \ldots, K-1$ and suppose subset S_k is removed from sequence S. The updated sequential unanimity rule $\mathcal{S}_{-k}$ is associated with sequence $(S_1, \ldots, S_{k-1}, S_{k+1}, \ldots, S_K)$.

By Lemma 2, there exists $C_k \in \mathcal{L}_k$ such that $S_k \subset C_k$ and $S_l \not\subset C_k$ for all $l \neq k$. We consider two cases:

Case 1: $k = 1$. Let R_N be a profile such that $R_i = a$ if and only if $i \in C_k$. By Theorem 2, $\mathcal{S}(R_N) = a$. By Lemma 2, for all $l = 2, \ldots, K-1$ there exists $i_l \in S_l \setminus C_k$, implying $R_{i_l} = b$. Since S_K is a singleton, by Definition 9, $\mathcal{S}_{-k}(R_N) = b$.

If $K = 2$ we are done. So suppose that $K > 2$ and consider the second case.

Case 2: $k = 2, \ldots, K-1$. Let R_N be a profile such that $R_i = a$ if and only if $i \in C_k$. By Theorem 2, $\mathcal{S}(R_N) = a$. Let $l = 1, \ldots, k-1$. By Lemma 2 there exists $i_l \in S_l \setminus C_k$, implying $R_{i_l} = b$. Moreover, by condition 2(iii) of Algorithm 1, there exists $j_l \in S_l \cap C_k$. Thus, $R_{j_l} = a$. Let $l = k+1, \ldots, K$. By a similar argument as in Case 1, there exists $i_l \in S_l \setminus C_k$, implying $R_{i_l} = b$. Thus, for all $l = 1, \ldots, K$ such that $l \neq k$, there exists $i_l \in S_l$ such that $R_{i_l} = b$. Since S_K is a singleton, by Definition 9, $\mathcal{S}_{-k}(R_N) = b$.

Hence, there exists a profile $R_N \in \mathcal{R}^N$ such that $\mathcal{S}(R_N) \neq \mathcal{S}_{-k}(R_N)$. This establishes sufficiency.

"⇐" We now prove necessity. Suppose condition 2(iv) is violated during the application of Algorithm 1. Thus, there exists some $k = 2, \ldots, K-1$ such that, for all $C \in \mathcal{L}_k$ there exists $h < k$ such that $S_h \subsetneq C$.

We will show that $\mathcal{S}(R_N) = \mathcal{S}_{-k}(R_N)$ for all $R_N \in \mathcal{R}^N$.

Let $R_N \in \mathcal{R}^N$. We consider two cases:

Case 1: There exist $i, j \in S_k$ such that $R_i \neq R_j$. Then, by Definition 9, $\mathcal{S}(R_N) = \mathcal{S}_{-k}(R_N)$.

Case 2: For all $i, j \in S_k$, $R_i = R_j$. Without loss of generality, suppose $R_i = a$ for all $i \in S_k$. We consider two sub-cases:

(a) There exists $C \in \mathcal{C}$ such that $S_k \subset C$ and $R_i = a$ for all $i \in C$. By Theorem 2, $\mathcal{S}(R_N) = a$. By Definition 9, there exists $l \leq k$ such that (i) $R_i = a$ for all $i \in S_l$ and (ii) for all $j = 1, \ldots, l-1$, there exists $i_j \in S_j$ such that $R_{i_j} = a$. Let l^* be the minimum such index.
If $l^* < k$, then $\mathcal{S}_{-k}(R_N) = a$.
If $l^* = k$ then we need to further examine the set C. If $C \in \mathcal{L}_k$, by assumption there exists $h < k$ such that $S_h \subsetneq C$. But this contradicts the minimality of l^*. Thus, $C \notin \mathcal{L}_k$, which, combined with $S_k \subset C$ and Step 2 of Algorithm 1, implies that $S_l \subsetneq C$ for some $l > k$. Hence, $C \in \mathcal{L}_l$ for some $l > k$. Moreover, $R_i = a$ for all $i \in S_l$. By step 2 (iii) of Algorithm 1, $S_j \cap C \neq \emptyset$ for all $j = 1, 2, \ldots l-1$. Hence, for all $j = 1, \ldots, l-1$, there exists $i_j \in S_j$ such that $R_{i_j} = a$. Thus, $\mathcal{S}_{-k}(R_N) = a$.

(b) For all $C \in \mathcal{C}$ such that $S_k \subset C$ there exists $i \in C$ such that $R_i = b$.
Let $C' \in \mathcal{C}$ be such that $R_i = x$ for all $i \in C'$ for some $x \in \{a, b\}$. (This set is guaranteed to exist and will necessarily satisfy $S_k \not\subset C'$.) By Theorem 2, $\mathcal{S}(R_N) = x$.
Let l^* be the minimum index l such that $S_l \subset C'$. Clearly, $l^* \neq k$. By step 2 (iii) of Algorithm 1, $S_l \cap C' \neq \emptyset$ for all $l = 1, \ldots, l^*-1$. Hence, for all $j = 1, \ldots, l^*-1$ there exists $i_j \in S_j$ such that $R_{i_j} = x$. Since $k \neq l^*$, $\mathcal{S}(R_N) = \mathcal{S}_{-k}(R_N) = x$. □

Lemma 1. By Step 2c of the algorithm, the set $\mathcal{C}^S$ satisfies property P1 (Minimality). We show that $\mathcal{C}^S$ satisfies property P2 (Moulin's property). First, we show that for all $C, C' \in \mathcal{C}^S$, $C \cap C' \neq \emptyset$. Let $C, C' \in \mathcal{C}^S$. By construction, there exist $k, l \in \{1, \ldots, K\}$ and two paths p and p' from S_1 to S_k, from S_1 to S_ℓ, respectively, such that $C = S_k \cup \{p\}$ and $C' = S_\ell \cup \{p'\}$. If $k = \ell$, then $S_k = S_\ell$, and thus $C \cap C' \neq \emptyset$. Suppose that $\ell < k$. If $S_l \cap S_k \neq \emptyset$, then $C \cap C' \neq \emptyset$. If $S_l \cap S_k = \emptyset$, by construction there is an element i of the path p such that $i \in S_\ell$. Hence, $\{p\} \cap S_\ell \neq \emptyset$. Therefore, $C \cap C' \neq \emptyset$.

Finally, let N' be a subset of N such that for all $C \in \mathcal{C}^S$, $N' \cap C \neq \emptyset$. We show that there exists $C' \in \mathcal{C}^S$ such that $C' \subset N'$. Let us first prove a claim.

Claim 1. There exists $k = 1, \ldots, K$ such that $S_k \subset N'$.

Proof: We proceed by contradiction. Suppose that for all $k = 1, \ldots, K$, there is $i_k \in S_k$ such that $i_k \notin N'$. By construction, there exists a path p from S_1 to S_K that is a subsequence of $(i_1, \ldots, i_{K-1})$. Thus, $N' \cap p = \emptyset$. Moreover, since S_K is a singleton, then $S_K \cap N' = \emptyset$. By construction, the set $S_K \cup \{p\}$, or a proper subset of it, will belong to $\mathcal{C}^S$. Let us call this subset C'. Since $N' \cap S_K = \emptyset$ and $N' \cap p = \emptyset$, we have identified a subset $C' \in \mathcal{C}^S$ such that $N' \cap C' = \emptyset$. This conclusion contradicts our assumption that for all $C \in \mathcal{C}^S$, $N' \cap C \neq \emptyset$. This concludes the proof of Claim 1.

Thus, let us consider the index $k \in \{1, \dots, K\}$ such that $S_k \subset N'$. If $k = 1$, then $S_1 \subset N'$ and we are done since $S_1 \in \mathcal{C}^S$. So, suppose that $k > 1$. We prove the following claim.

Claim 2.. Let k be the minimum index such that $S_k \subset N'$. For all $\ell < k$, $S_\ell \cap N' \neq \emptyset$.

Proof: We proceed by contradiction. Suppose that there is $\ell < k$ such that $S_\ell \cap N' = \emptyset$. If $\ell = 1$, then we have $S_1 \in \mathcal{C}^S$ and $N' \cap S_1 = \emptyset$. This conclusion contradicts our assumption that for all $C \in \mathcal{C}^S$, $N' \cap C \neq \emptyset$. Suppose that $\ell > 1$. Since k is chosen to be minimum, for all $\ell' = 1, \dots, \ell - 1$, there is $i_{\ell'} \in S_{\ell'}$ such that $i_{\ell'} \notin N'$. Thus, there exists a path p from S_1 to S_ℓ that is a subsequence of $(i_1, \dots, i_{\ell-1})$. Thus, $N' \cap p = \emptyset$. By construction, the set $S_l \cup \{p\}$, or a proper subset of it, will belong to $\mathcal{C}^S$. Let us call this subset C'. Since $S_\ell \cap N' = \emptyset$ and $\{p\} \cap N' = \emptyset$, we have identified a subset $C' \in \mathcal{C}^S$ such that $N' \cap C' = \emptyset$. This conclusion contradicts our assumption that for all $C \in \mathcal{C}^S$, $N' \cap C \neq \emptyset$. This concludes the proof of Claim 2.

In light of Claim 2, for all $\ell = 1, \dots, k-1$, let $i_\ell \in S_\ell \cap N'$. Thus, there exists a path p from S_1 to S_k that is a subsequence of $(i_1, \dots, i_{k-1})$. By construction, the set $S_k \cup \{p\}$, or a proper subset of it, belongs to $\mathcal{C}^S$. We call this subset C'. Thus, we have identified $C' \in \mathcal{C}^S$ such that $C' \subset N'$. □

Appendix D. Extending Algorithms 1 and 2 to the full preference domain

We now show how our results extend to the full preference domain, which allows voters to be indifferent between alternatives. The candidate set A is expanded to include 0, indicating a tie between a and b. Voter i's preference relation over candidates is denoted by $\tilde{R}_i \in \{a, b, 0\}$. A preference of $\tilde{R}_i = x$, with $x \in \{a, b\}$, means that voter i has a strict preference for candidate x, whereas $\tilde{R}_i = 0$ means that i is indifferent between a and b. A voter with a strict preference for $x \in \{a, b\}$ is assumed to prefer the null outcome 0 to an outcome $y \in \{a, b\}$ such that $y \neq x$.

The property of strategy-proofness ensures that it is not possible for a voter to misreport his preferences and obtain an outcome that he prefers to that obtained under truthfulness. Since preferences in the full domain are a little more involved, the following definition clarifies what we mean.

Definition 12. A rule f is **strategy-proof** if for all profiles $\tilde{R}_N \in \tilde{\mathcal{R}}^N$, each voter $i \in N$, and each $\tilde{R}'_i \in \{a, b, 0\}$ such that $\tilde{R}'_i \neq \tilde{R}_i$, i does not prefer $f(\tilde{R}'_i, \tilde{R}_{-i})$ to $f(\tilde{R}_N)$.

Thus, to establish a rule's strategy-proofness it is sufficient to focus on voters having a strict preference for candidate a or b. If such voters cannot profitably misreport their preferences, then the criterion is met.

We need to take similar care in adapting the definition of neutrality. Given a permutation $\pi : \{a, b, 0\} \mapsto \{a, b, 0\}$ such that $\pi(0) = 0$ and a profile $\tilde{R}_N \in \tilde{\mathcal{R}}^N$, define the profile $\pi\tilde{R}_N$ as $\pi\tilde{R}_i = \pi(\tilde{R}_i)$ for all $i \in N$. Thus, voters who are indifferent between candidates remain indifferent after the identities of a and b have been flipped by π.

Definition 13. A rule f is **neutral** if for each permutation π of $\{a, b, 0\}$ such that $\pi(0) = 0$, and all profiles $\tilde{R}_N \in \tilde{\mathcal{R}}^N$,

$$f(\pi\tilde{R}_N) = \pi(f(\tilde{R}_N)).$$

With indifferences, the M-winning coalition rules of Section 3 may not be well-defined. They thus need to be modified. Consider a neutral and strategy-proof rule $f : \tilde{\mathcal{R}}^N \mapsto \{a, b, 0\}$. Examples of such a rule f may include majority rule and dictatorship where, to be clear, majority rule is defined as the rule f^M satisfying:

$$f^M(\tilde{R}_N) = \begin{cases} a, & \text{if } |i \in N : \ s.t.\ \tilde{R}_i = a| > |i \in N : \ s.t.\ \tilde{R}_i = b| \\ b, & \text{if } |i \in N : \ s.t.\ \tilde{R}_i = b| > |i \in N : \ s.t.\ \tilde{R}_i = a| \\ 0, & \text{otherwise.} \end{cases}$$

Definition 14. Consider an M-winning coalition set $\mathcal{C}$ and a neutral and strategy-proof rule $f : \tilde{\mathcal{R}}^N \mapsto \{a, b, 0\}$. The **M-winning coalition rule with f as default** $\mathcal{C}^f : \tilde{\mathcal{R}}^N \mapsto \{a, b, 0\}$ is defined as follows: for all $\tilde{R}_N \in \tilde{\mathcal{R}}^N$ and $x \in \{a, b\}$,

$$\exists C \in \mathcal{C} \text{ s.t. } \tilde{R}_i = x \text{ for all } i \in C \ \Rightarrow \ \mathcal{C}(\tilde{R}_N) = x;$$

otherwise, $\mathcal{C}^f(\tilde{R}_N) = f(\tilde{R}_N)$.

So, if there exists an element of $\mathcal{C}$ that unanimously supports candidate $x \in \{a, b\}$, the rule $\mathcal{C}^f$ picks it. If there is no such consensus, we move on to a second stage in which the outcome is determined via the default rule f that retains the desirable features of strategy-proofness and neutrality. This two-stage way of redefining M-winning coalition rules to accommodate the full domain is consistent with Bartholdi et al. (2021) treatment. Now, we move on to sequential unanimity rules. The full-domain adaptation here is a little more subtle.

Definition 15. Consider a sequence $S = (S_1, \dots, S_K)$ of subsets of N satisfying $|S_K| = 1$ and a neutral and strategy-proof rule $f : \tilde{\mathcal{R}}^N \mapsto \{a, b, 0\}$. The **sequential unanimity rule with f as default** $S^f : \tilde{\mathcal{R}}^N \mapsto \{a, b, 0\}$, is defined as follows:

1. If $\tilde{R}_i = x \in \{a, b\}$ all $i \in S_1$, then $S^f(\tilde{R}_N) = x$. Otherwise

2. For $k = 2, \ldots, K$
   - If $\tilde{R}_i = x \in \{a, b\}$ for all $i \in S_k$, and for all $l = 1, \ldots, k-1$ there exists $i_l \in S_l$ such that $R_{i_l} = x$, then STOP. We have $\mathcal{S}^f(\tilde{R}_N) = x$.
   - If $\tilde{R}_i = 0$ for all $i \in S_k$, then STOP. We have $\mathcal{S}^f(\tilde{R}_N) = f(\tilde{R}_N)$.
3. If the above procedure reaches $k = K$ without stopping, then $\mathcal{S}^f(\tilde{R}_N) = f(\tilde{R}_N)$.

The adapted versions of M-winning coalition and sequential unanimity rules retain their desirable features. The next Proposition states this result.

Proposition 1. *Let $f : \tilde{\mathcal{R}}^N \mapsto \{a, b, 0\}$ be a neutral and strategy-proof rule. Then, all M-winning coalition rules and all sequential unanimity rules with f as default are neutral and strategy-proof.*

Proof. It is easy to show that M-winning coalition rules with f as default are neutral. We turn to strategy-proofness. Consider a rule $\mathcal{C}^f$ with associated M-winning coalition set $\mathcal{C}$ and neutral and default f. Let $\tilde{R}_N$ be a profile. We consider two cases: (i) there exist $C \in \mathcal{C}$ and $x \in \{a, b\}$ such that $\tilde{R}_i = x$ for all $i \in C$ and (ii) there do not exist $C \in \mathcal{C}$ and $x \in \{a, b\}$ such that $\tilde{R}_i = x$ for all $i \in C$. In both cases, and consistent to Definition 12, to establish strategy-proofness we need only consider voters $i \in N$ such that $\tilde{R}_i \in \{a, b\}$.

In case (i), $\mathcal{C}^f(\tilde{R}_N) = x$. Suppose without loss of generality that $x = a$. Let $i \in N$ and suppose that $\tilde{R}_i = b$, meaning that $i \notin C$. In this case, $\mathcal{C}^f(\tilde{R}'_i, \tilde{R}_{-i}) = a$ for any $\tilde{R}'_i \in \{a, b, 0\}$.

In case (ii), $\mathcal{C}^f(\tilde{R}_N) = f(\tilde{R}_N)$. Let $i \in N$ such that $\tilde{R}_i \in \{a, b\}$ and $\tilde{R}_i \neq f(\tilde{R}_N)$. Without loss of generality, suppose that $\tilde{R}_i = b$. Let $\tilde{R}'_i \in \{a, 0\}$.

Suppose first that $\tilde{R}'_i = a$. Then either (1) there exists $C \in \mathcal{C}$ such that $\tilde{R}_j = a$ for all $j \in C$, or (2) there is no such C, so that $\mathcal{C}^f(\tilde{R}'_i, \tilde{R}_{-i}) = f(\tilde{R}'_i, \tilde{R}_{-i})$. In case (1), $\mathcal{C}^f(\tilde{R}'_i, \tilde{R}_{-i}) = a$. Thus, $\mathcal{C}^f(\tilde{R}'_i, \tilde{R}_{-i})$ is not preferred to $f(\tilde{R}_N)$. In case (2), $\mathcal{C}^f(\tilde{R}'_i, \tilde{R}_{-i}) = f(\tilde{R}'_i, \tilde{R}_{-i})$. The strategy-proofness of f ensures that $f(\tilde{R}'_i, \tilde{R}_{-i})$ is not preferred to $f(\tilde{R}_N)$. Thus, in both cases $\mathcal{C}^f(\tilde{R}'_i, \tilde{R}_{-i})$ is not preferred to $\mathcal{C}^f(\tilde{R}_N)$.

Suppose now that $\tilde{R}'_i = 0$. Then $\mathcal{C}^f(\tilde{R}'_i, \tilde{R}_{-i}) = f(\tilde{R}'_i, \tilde{R}_{-i})$. The strategy-proofness of f ensures that $\mathcal{C}^f(\tilde{R}'_i, \tilde{R}_{-i}) = f(\tilde{R}'_i, \tilde{R}_{-i})$ is not preferred to $\mathcal{C}^f(\tilde{R}_N) = f(\tilde{R}_N)$.

We conclude that $\mathcal{C}^f$ is strategy-proof.

We now consider sequential unanimity rules. It is easy to show that they are neutral. To establish their strategy-proofness, consider a rule $\mathcal{S}^f$ with associated sequence $\mathrm{S} = (S_1, \ldots, S_K)$ and default f. We distinguish between two cases.

Case 1: There exist $k \in \{1, \ldots, K\}$ and $x \in \{a, b\}$ such that (i) $\tilde{R}_i = x$ for all $i \in S_k$ and (ii) for all $l = 1, ., k-1$ there exists $i_l \in S_l$ such that $\tilde{R}_{i_l} = x$.

Suppose k is the minimum such index and that, without loss of generality, $x = a$. Hence $\mathcal{S}^f(\tilde{R}_N) = a$. Consider $i \in N$ and suppose that $\tilde{R}_i = b$. This implies that $i \notin S_k$. Let $\tilde{R}'_i \in \{a, 0\}$. Then, repeating the reasoning in the proof of Proposition 2, we obtain $\mathcal{S}^f(\tilde{R}'_i, \tilde{R}_{-i}) = a$. Thus, i dot not prefer $\mathcal{S}^f(\tilde{R}'_i, \tilde{R}_{-i})$ to $\mathcal{S}^f(\tilde{R}_N)$.

Case 2: There exist no $k \in \{1, \ldots, K\}$ and $x \in \{a, b\}$ such that (i) $\tilde{R}_i = x$ for all $i \in S_k$ and (ii) for all $l = 1, ., k-1$ there exists $i_l \in S_l$ such that $\tilde{R}_{i_l} = x$.

Thus, $\mathcal{S}(\tilde{R}_N) = f(\tilde{R}_N)$. Let $i \in N$ and suppose without loss of generality that $\tilde{R}_i = b$. Let $\tilde{R}'_i \in \{a, 0\}$. Suppose first that $\tilde{R}'_i = a$. There are two cases: either (I) there exists $k \in \{1, \ldots, K\}$ such that (i) $\tilde{R}_i = a$ for all $i \in S_k$ and (ii) for all $l = 1, ., k-1$ there exists $i_l \in S_l$ such that $\tilde{R}_{i_l} = a$, or (II) there exists no such $k \in \{1, \ldots, K\}$. In case (I), $\mathcal{S}^f(\tilde{R}'_i, \tilde{R}_{-i}) = a$. In case (II), $\mathcal{S}^f(\tilde{R}'_i, \tilde{R}_{-i}) = f(\tilde{R}'_i, \tilde{R}_{-i})$ and the strategy-proofness of f ensures that i does not prefer $f(\tilde{R}'_i, \tilde{R}_{-i})$ to $f(\tilde{R}_N)$. In either case, i does not prefer $\mathcal{S}^f(\tilde{R}'_i, \tilde{R}_{-i})$ to $\mathcal{S}^f(\tilde{R}_N)$.

Suppose now that $\tilde{R}'_i = 0$. Then, $\mathcal{S}^f(\tilde{R}'_i, \tilde{R}_{-i}) = f(\tilde{R}'_i, \tilde{R}_{-i})$ and the strategy-proofness of f ensures that i does not prefer $f(\tilde{R}'_i, \tilde{R}_{-i})$ to $f(\tilde{R}_N)$. Thus, once again, i does not prefer $\mathcal{S}^f(\tilde{R}'_i, \tilde{R}_{-i})$ to $\mathcal{S}^f(\tilde{R}_N)$.

We conclude that $\mathcal{S}^f$ is strategy-proof. □

We continue by establishing analogues of Theorems 2 and 3 for the full preference domain. In both cases, the proofs are straightforward adaptations of the arguments used previously. We include them for completeness.

Theorem 4. *Given an M-winning coalition set $\mathcal{C}$ and a neutral and strategy-proof rule f, consider the M-winning coalition rule $\mathcal{C}^f$. Furthermore, consider a sequential unanimity rule $\mathcal{S}^f$, whose associated sequence S is an output of Algorithm 1 with $\mathcal{C}$ as input. For all $\tilde{R}_N \in \tilde{\mathcal{R}}^N$, we have $\mathcal{C}^f(\tilde{R}_N) = \mathcal{S}^f(\tilde{R}_N)$.*

Proof. Consider the sequential unanimity rule $\mathcal{S}^f$ with associated sequence S, and let $\tilde{R}_N \in \tilde{\mathcal{R}}^N$. We distinguish between two cases:

Case 1: For some $x \in \{a, b\}$, there exists $C \in \mathcal{C}$ such that $\tilde{R}_i = x$ for all $i \in C$. Then, by definition, $\mathcal{C}^f(\tilde{R}_N) = x$.

Consider the sequence $\mathrm{S} = (S_1, \ldots, S_K)$ produced by Algorithm 1 with $\mathcal{C}$ as input. By construction, there exists $k \in \{1, \ldots, K\}$ such that $S_k \subset C$. We distinguish between two cases: (i) $C = S_k$ and (ii) $S_k \subsetneq C$. In case (i), $S_l \in \mathcal{C}$ for all $l = 1, \ldots, k-1$. Hence, by property P2, $S_l \cap C \neq \emptyset$ for all $l = 1, \ldots, k-1$. In case (ii), for all $l = 1, \ldots, k-1$, either $S_l \in \mathcal{C}$, or $S_l \notin \mathcal{C}$. If $S_l \in \mathcal{C}$, then, by property P2, $S_l \cap C \neq \emptyset$. If $S_l \notin \mathcal{C}$, then condition (iii) in Step 2 of Algorithm 1 implies that $S_l \cap C \neq \emptyset$.

Thus, in either case, for all $l < k$, there exists $i_l \in S_l \cap C$. Moreover, recall that $\tilde{R}_i = x$ for all $i \in C \supset S_k$. Hence, $\mathcal{S}^f(\tilde{R}_N) = x = \mathcal{C}^f(\tilde{R}_N)$.

Case 2: There does not exist any $C \in \mathcal{C}$ and $x \in \{a, b\}$, such that $\tilde{R}_i = x$ for all $i \in C$. In this case, $\mathcal{C}^f(\tilde{R}_N) = f(\tilde{R}_N)$.

Suppose that $\mathcal{S}^f(\tilde{R}_N) \neq f(\tilde{R}_N)$. Then, there exists $x \in \{a, b\}$ and $k \in \{1, ., K\}$ such that (i) $\tilde{R}_i = x \neq f(\tilde{R}_N)$ for all $i \in S_k$ and (ii) for all $l = 1, 2, \ldots, k-1$, there exists $i_l \in S_l$ such that $\tilde{R}_{i_l} = x$. By Theorem 3, the set $\{i_1, i_2, ., i_{k-1}\} \cup S_k$, or a proper subset of it, will belong to the output M-winning coalition set which, by Corollary 3, equals $\mathcal{C}$. Thus, there exists $C \in \mathcal{C}$ such that $\tilde{R}_i = x$ for all $i \in C$, which contradicts the assumption of Case 2. Hence, $\mathcal{S}^f(\tilde{R}_N) = f(\tilde{R}_N) = \mathcal{C}^f(\tilde{R}_N)$.

□

Finally, we prove the reverse result.

Theorem 5. *Given a sequence $S = (S_1, \ldots, S_K)$ of subsets of N satisfying $|S_K| = 1$ and a neutral and strategy-proof rule f, consider the rule $\mathcal{S}^f$. Furthermore, consider the rule $\mathcal{C}^f$, whose associated M-winning coalition set $\mathcal{C}^S$ is the output of Algorithm 2 with S as input. For all $\tilde{R}_N \in \tilde{\mathcal{R}}^N$, we have $\mathcal{C}^f(\tilde{R}_N) = \mathcal{S}^f(\tilde{R}_N)$.*

Proof. Consider a profile $\tilde{R}_N$. By Lemma 1, $\mathcal{C}^S$ is an M-winning coalition set. We distinguish between two cases:

Case 1: There exists $k \in \{1, \ldots, K\}$ and $x \in \{a, b\}$ such that (i) for all $i \in S_k$, $\tilde{R}_i = x$ and (ii) for all $l = 1, \ldots, k-1$, there exists $i_l \in S_l$ such that $\tilde{R}_{i_l} = x$.

For simplicity, suppose that k is the smallest index for which this statement is true and, without loss of generality that $x = a$. As a result, $\mathcal{S}^f(\tilde{R}_N) = a$. Following the same logic as the proof of Theorem 3, there exists $C \in \mathcal{C}^S$ such that $C \subset \{i_1, \ldots, i_{k-1}\} \cup S_k$. This implies $\mathcal{C}^f(\tilde{R}_N) = \mathcal{S}^f(\tilde{R}_N) = a$.

Case 2: There exists no $k \in \{1, \ldots, K\}$ and $x \in \{a, b\}$ such that (i) for all $i \in S_k$, $\tilde{R}_i = x$ and (ii) for all $l = 1, \ldots, k-1$, there exists $i_l \in S_l$ such that $\tilde{R}_{i_l} = x$.

Thus, $\mathcal{S}^f(\tilde{R}_N) = f(\tilde{R}_N)$. We make the following Claim.

Claim 1.. Assume the condition of case 2 holds.[11] *Then, there exists no $C \in \mathcal{C}^S$ and $x \in \{a, b\}$ such that $\tilde{R}_i = x$ for all $i \in C$.*

Proof: We prove the contrapositive statement. Suppose that there exists $C \in \mathcal{C}^S$ and $x \in \{a, b\}$ such that $\tilde{R}_i = x$ for all $i \in C$. Without loss of generality, suppose that $x = a$. By construction, there exists a subset S_k and a path p_k from S_1 to S_k such that $C = \{p_k\} \cup S_k$. Thus, $\tilde{R}_i = a$ for all $i \in S_k$. Moreover, Definition 11 implies that, for all $h = 1, \ldots, k-1$ such that $S_h \cap S_k = \emptyset$, there exists $i_h \in S_h \cap \{p_k\}$. Thus, for all $l = 1, \ldots, k-1$, there exists $i_l \in S_l$ such that either $i_l \in S_k$ or $i_l \in \{p_k\}$. Hence, for all $l = 1, \ldots, k-1$, there exists $i_l \in S_l$ such that $\tilde{R}_{i_l} = a$. Thus, there exists $k \in \{1, \ldots K\}$ and $x \in \{a, b\}$ such that (i) for all $i \in S_k$, $\tilde{R}_i = a$ and (ii) for all $l = 1, \ldots, k-1$, there exists $i_l \in S_l$ such that $\tilde{R}_i = a$. This conclusion contradicts our assumption in Case 2 and thus ends the proof of Claim 1.

Claim 1 implies that $\mathcal{C}^f(\tilde{R}_N) = \mathcal{S}^f(\tilde{R}_N) = f(\tilde{R}_N)$. Thus, the desired conclusion holds also in Case 2. □

References

Bartholdi, L., Hann-Caruthers, W., Josyula, M., Tamuz, O., Yariv, L., 2021. Equitable voting rules Econometrica 89, 563–589.
Basile, A., Rao, S., Rao, K.B., 2021. The structure of two-valued coalitional strategy-proof social choice functions. J. Math. Econ. 95, 102474.
Basile, A., Rao, S., Rao, K.B., 2022a. Anonymous, non-manipulable binary social choice. Games Econ. Behav. 133, 138–149.
Basile, A., Rao, S., Rao, K.B., 2022b. Geometry of anonymous binary social choices that are strategy-proof. Math. Soc. Sci. 116, 85–91.
Basile, A., Rao, S., Rao, K. P. S.B., 2022c. Binary strategy-proof social choice functions with indifference. Econ. Theory 73, 807–826.
Fishburn, P.C., Gehrlein, W.V., 1977. Collective rationality versus distribution of power for binary social choice functions. J. Econ. Theory 15 (1), 72–91.
Freixas, J., Molinero, X., 2009. Simple games and weighted games: a theoretical and computational viewpoint Discrete Appl. Math. 157, 1496–1508.
Gershkov, A., Moldovanu, B., Shi, X., 2017. Optimal voting rules. Rev. Econ. Studies 84 (2), 688–717.
Gershkov, A., Szentes, B., 2009. Optimal voting schemes with costly information acquisition. J. Econ. Theory 144 (1), 36–68.
Houy, N., Zwicker, W.S., 2014. The geometry of voting power: weighted voting and hyper-ellipsoids. Games Econ. Behav. 84, 7–16.
Kivinen, S., 2023. On the manipulability of equitable voting rules.
Lahiri, A., Pramanik, A., 2020. On strategy-proof social choice between two alternatives. Soc. Choice Welfare 54, 581–607.
Larsson, B., Svensson, L.G., 2006. Strategy-proof voting on the full preference domain. Math. Soc. Sci. 52 (3), 272–287.
May, K.O., 1952. A set of independent necessary and sufficient conditions for simple majority decision. Econometrica 20, 680–684.
Moulin, H., 1983. The Strategy of Social Choice. North-Holland.
Riboni, A., Ruge-Murcia, F.J., 2010. Monetary policy by committee: consensus, chairman dominance, or simple majority? Quarterly J. Econ. 125 (1), 363–416.
Taylor, A., Zwicker, W., 1992. A characterization of weighted voting. Proc. Am. Math. Soc. 115, 1089–1094.
Taylor, A., Zwicker, W., 1993. Weighted voting, multicameral representation, and power. Games Econ. Behav. 5 (1), 170–181.
Taylor, A., Zwicker, W., 1995. Simple games and magic squares. J. Combinatorial Theory Series A 71 (1), 67–88.
Taylor, A.D., Zwicker, W.S., 1996. Quasi-weightings, trading, and desirability relations in simple games. Games Econ. Behav. 16 (2), 331–346.
Taylor, A.D., Zwicker, W.S., 1999. Simple Games: Desirability Relations, Trading, Pseudoweightings. Princeton University Press.

[11] That is, there exists no $k \in \{1, \ldots, K\}$ and $x \in \{a, b\}$ such that: (i) for all $i \in S_k$, $\tilde{R}_i = x$ and (ii) for all $l = 1, \ldots, k-1$, there exists $i_l \in S_l$ such that $\tilde{R}_{i_l} = x$.