Computability of simple games: A characterization and application to the core

Kumabe, Masahiro and Mihara, H. Reiju

July 2006

Computability of simple games: A characterization and application to the core

Masahiro Kumabe
kumabe@u-air.ac.jp
Kanagawa Study Center, The University of the Air
2-31-1 Ooka, Minami-ku, Yokohama 232-0061, Japan

H. Reiju Mihara*
reiju@ec.kagawa-u.ac.jp
Graduate School of Management, Kagawa University
Takamatsu 760-8523, Japan

July, 2006

Abstract

It was shown earlier that the class of algorithmically computable simple games (i) includes the class of games that have finite carriers and (ii) is included in the class of games that have finite winning coalitions. This paper characterizes computable games, strengthens the earlier result that computable games violate anonymity, and gives examples showing that the above inclusions are strict. It also extends Nakamura's theorem about the nonemptyness of the core and shows that computable simple games have a finite Nakamura number, implying that the number of alternatives that the players can deal with rationally is restricted.

Journal of Economic Literature Classifications: C71, D71, C69.

Keywords: Voting games, infinitely many players, recursion theory, Turing computability, computable manuals and contracts.

*The corresponding author; the family name is "Mihara." Phone: +81-87-832-1831. Fax: +81-87-832-1820.

1 Introduction

We investigate algorithmic computability of a particular class of coalitional games (cooperative games), called *simple games* (voting games). One can think of simple games as representing voting methods; alternatively, as representing "manuals" or "contracts." We give a characterization of computable simple games and apply it to the theory of the core. For the latter application, we extend Nakamura's theorem [34] regarding the core of simple games to the framework where not all subsets of players are deemed to be a coalition.

1.1 Computability analysis of social choice

Most of the paper (except the part on the theory of the core) can be viewed as a contribution to the foundations of *computability analysis of social choice*, which studies algorithmic properties of social decision-making. This literature includes Kelly [22], Lewis [27], Bartholdi et al. [11, 12], and Mihara [30, 31, 33], who study issues in social choice using *recursion theory* (the theory of computability and complexity, or study of algorithms). These works, which are mainly concerned with the complexity of rules or cooperative games in themselves, can be distinguished from the closely related studies of the complexity of *solutions* for cooperative games, such as Deng and Papadimitriou [15] and Fang et al. [18]. (More generally, applications of recursion theory to economic theory and game theory include Spear [42], Canning [13], Anderlini and Felli [1], Anderlini and Sabourian [2], Prasad [37], Richter and Wong [38, 39], and Evans and Thomas [17]. See also Lipman [28] and Rubinstein [40] for surveys of the literature on bounded rationality in these areas.)

The importance of computability in social choice theory would be unarguable. First, the use of the language by social choice theorists suggests the importance. For example, Arrow defined a social welfare function to be a "*process or rule*" which, for each profile of individual preferences, "states" a corresponding social preference [6, p. 23], and called the function a "*procedure*" [6, p. 2]. Indeed, he later wrote [7, p. S398] in a slightly different context, "The next step in analysis, I would conjecture, is a more consistent assumption of *computability* in the formulation of economic hypotheses" (emphasis added). Second, there is a normative reason. Algorithmic social choice rules specify the procedures in such a way that the same results are obtained irrespective of who carries out a computation, leaving no room for personal judgments. In this sense, computability of social choice rules formalizes the notion of "due process." (Richter and Wong [38] give reasons for studying computability-based economic theories from the viewpoints of bounded rationality, computational economics, and complexity analysis. These reasons partially apply to studying computable rules in social choice.)

1.2 Simple games with countably many players

Simple games have been central to the study of social choice (e.g., Banks [9], Austen-Smith and Banks [8], and Peleg [36]). *Simple games* on an algebra of coalitions of players assign either 0 or 1 to each coalition (member of the algebra). In the setting of players who face a yes/no question, a coalition intuitively describes those players who vote yes. A simple game is characterized by its *winning* coalitions—those assigned the value 1. (The other coalitions are *losing*.) Winning coalitions are understood to be those coalitions whose unanimous votes are decisive.

When there are only *finitely* many players, we can construct a finite table listing all winning coalitions. Computability is automatically satisfied, since such a table gives an algorithm for computing the game. The same argument does not hold when there are *infinitely* many players. Indeed, some simple games are noncomputable, since there are uncountably many simple games but only countably many computable ones (because each computable game is associated with an algorithm).

There are two typical approaches to introducing infinite population to a social choice model. In the "variable population" approach, players are potentially infinite, but each problem (or society) involves only finitely many players. Indeed, well-known schemes, such as simple majority rule, unanimity rule, and the Condorcet and the Borda rules, are all algorithms that apply to problems of any finite size. Kelly [22] adopts this approach, giving examples of noncomputable social choice rules. In the "fixed population" approach, which we adopt, each problem involves the whole set of infinitely many players. This approach dates back to Downs [16], who consider continuous voter distributions. The paper by Banks et al. [10] is a recent example of this approach to political theory.

Taking the "fixed population" approach, we consider a fixed infinite set of players in this study of simple games. Roughly speaking, a simple game is *computable* if there is a Turing program (finite algorithm) that can decide from a description (by integer) of each coalition whether it is winning or losing. To be more precise, we have to be more specific about what we mean by a "description" of a coalition. This suggests the following: First, since each member of a coalition should be describable in words (in English), it is natural to assume that the set N of (the names of) players is countable, say, $N = \mathbf{N} = \{0, 1, 2, \ldots\}$. Second, since one can describe only countably many coalitions, we have to restrict coalitions. Finite or cofinite coalitions can be described by listing their members or nonmembers completely. But restricting coalitions to these excludes too many coalitions of interest—such as the set of even numbers. A natural solution is to describe coalitions by a Turing program that can decide for the name of each player whether she is in the coalition. Since each Turing program has its code number (Gödel number), the coalitions describable in this manner are describable by an integer, as desired. Our notion of computability (δ-*computability*) focuses on this class of coalitions—*recursive* coalitions—as well as the method (characteristic index) of describing them.

A fixed population of countably many *players* arises not only in voting but in other contexts, such as a special class of multi-criterion decision making—depending on how we interpret a "player":

Simulating future generations One may consider countably many *players* (people) extending into the indefinite future.

Uncertainty One may consider finitely many *persons* facing countably many states of the world [30]: each *player* can be interpreted as a particular *person* in a particular *state*. The decision has to be made before a state is realized and identified. (This idea is formalized by Gomberg et al. [21], who introduce "n-period coalition space," where n is the number of persons.)

Team management Putting the right people (and equipment) in the right places is basic to team management.[1] To ensure "due process" (which is sometimes called for), can a manager of a company write a "manual" (computable simple game[2]) elaborating the conditions that a team must meet?

Fix a particular task such as operating an exclusive agency of the company.[3] A *team* consists of members (people) and equipment. The manager's job is to organize or give a licence to a team that satisfactorily performs the task. Each member (or equipment) is described by attributes such as skills, position, availability at a particular time and place (in case of equipment such as a computer, the attributes may be the kind of operating system, the combination of software that may run at the same time, as well as hardware and network specifications). Each such *attribute* can be thought of as a particular yes/no question, and there are countably many such questions.[4]

[1] In line with much of cooperative game theory, we put aside the important problems of economics of organization, such as coordinating the activities of the team members by giving the right incentives.

[2] Like Anderlini and Felli [1], who view contracts as algorithms, we view "manuals" as algorithms. They derive contract incompleteness through computability analysis.

[3] Extension to finitely or countably many tasks is straightforward. Redefine a *team* as consisting of members, equipment, *and* tasks. Then introduce a player for each task. Since a task can be regarded as a negative input, it will be more natural to assign 0 to those tasks undertaken and 1 to those not undertaken (think of the monotonicity condition).

[4] According to a certain approach (e.g., Gilboa [20]) to modeling scientific inquiry, a "state" is an infinite sequence of 0's and 1's (answers to countably many questions) and a "theory" is a Turing program describing a state. This team management example is inspired by this approach in philosophy of science. If we go beyond the realm of social choice, we can indeed find many other interpretations having a structure similar to this

Here, each *player* can be interpreted as a particular attribute of a particular member (or equipment).[5] In other words, each *coalition* is identified with a 0-1 "matrix" of finitely many rows (each row specifying a member) and countably many columns (each column specifying a particular attribute).[6]

1.3 Overview of the results

Adopting the above notion of computability for simple games, Mihara [33] gives a sufficient condition and necessary conditions for computability. The sufficient condition [33, Proposition 5] is intuitively plausible: simple games with a *finite carrier* (such games are in effect finite, ignoring all except finitely many, fixed players' votes) are computable. A necessary condition [33, Corollary 10] in the paper seems to exclude "nice" (in the voting context) infinite games: computable simple games have both finite winning coalitions and cofinite losing coalitions. He leaves open the questions (i) whether there exists a computable simple game that has no finite carrier and (ii) whether there exists a noncomputable simple game that has both finite winning coalitions and cofinite losing coalitions. The first of these questions is particularly important since if the answer were no, then only the games that are in effect finite would be computable, a rather uninteresting result. The answers to these questions (i) and (ii) are affirmative. We construct examples in Section 6 to show their existence. The construction of these examples depends in essential ways on Proposition 4 (which gives a necessary condition for a simple game to be computable) or on the easier direction of Theorem 5 (which gives a sufficient condition). In contrast, the results in Mihara [33] are not useful enough for us to construct such examples.

Theorem 5 gives a necessary and sufficient condition for simple games to be computable. The condition roughly states that "finitely many, unnecessarily fixed players matter."

To explain the condition, let us introduce the notion of a "determining string." Given a coalition S, its *k-initial segment* is the string of 0's and 1's of length k whose jth element (counting from zero) is 1 if $j \in S$ and is 0 if $j \notin S$. For example, if $S = \{0, 2, 4\}$, its 0-initial segment, 1-initial

example, such as elaborating the conditions for a certain medicine to take the desired effects and deciding whether a certain act is legal or not.

[5] From the viewpoint of "due process," it would be reasonable to define a simple game not for the set of (the names of) members but for the set of attributes. (This is particularly important where games cannot meet anonymity.) Considering characteristic games for the set of attributes ("skills") can make it easy to express certain allocation problems and give solutions to them (see Yokoo et al. [45]).

[6] Since different questions may be interrelated, some "matrix" may not make much sense. One might thus want to restrict admissible "matrices." This point is not crucial to our discussion, provided that there are infinitely many admissible "matrices" consisting of infinitely many 0's and infinitely many 1's (in such cases characteristic indices are the only reasonable way of naming coalitions).

segment, ..., 8-initial segment, ... are, respectively, the empty string, the string 1, the string 10, the string 101, the string 1010, the string 10101, the string 101010, the string 1010100, the string 10101000, ... We say that a (finite) string τ is *winning determining* if any coalition G extending τ (i.e., τ is an initial segment of G) is winning. We define *losing determining* strings similarly.

The necessary and sufficient condition for computability according to Theorem 5 is the following: there are computably listable sets T_0 of losing determining strings and T_1 of winning determining strings such that any coalition has an initial segment in one of these sets. In the above example, the condition implies that at least one string from among the empty string, 1, 10, ..., 10101000, ... is in T_0 or in T_1—say, 1010 is in T_1. Then any coalition of which 0 and 2 are members but 1 or 3 is not, is winning. In this sense, *one can determine whether a coalition is winning or losing by examining only finitely many players' membership.* In general, however, *one cannot do so by picking finitely many players before a coalition is given.*

Theorem 5 has an interesting implication for the nature of "manuals" or "contracts," if we regard them as being composed of computable simple games (e.g., the team management example in Section 1.2). Consider how many "criteria" (players; e.g., member-attribute pairs) are needed for a "manual" to determine whether a given "situation" (coalition; e.g., team) is "acceptable" (winning; e.g., satisfactorily performs a given task). *While increasingly complex situations may require increasingly many criteria, no situation (however complex) requires infinitely many criteria.* The conditions (such as "infinitely many of the prime-numbered criteria must be met") based on infinitely many criteria are ruled out.

The proof of Theorem 5 uses the *recursion theorem.* It involves much more intricate arguments of recursion theory than those in Mihara [33] giving only a partial characterization of the computable games.[7]

A natural characterization result might relate computability to well-known properties of simple games, such as monotonicity, properness, strongness, and nonweakness. Unfortunately, we are not likely to obtain such a result: as we clarify in a companion paper [26], computability is "unrelated to" the four properties just mentioned.

The earlier results [33] are easily obtained from Theorem 5. For example, if a computable game has a winning coalition, then, an initial segment of that coalition is winning determining, implying that (Proposition 9) the game has a finite winning coalition and a cofinite winning coalition. We give simple proofs to some of these results in Section 4. In particular, Proposition 12 strengthens the earlier result [33, Corollary 12] that computable games violate anonymity. (Detailed studies of anonymous rules based on in-

[7]Theorem 5 can also be derived from results in Kreisel et al. [25] and Ceĭtin [14]. See Remark 2 for details.

finite simple games include Mihara [29], Fey [19], and Gomberg et al. [21].)

1.4 Application to the theory of the core

Most cooperative game theorists are more interested in the properties of a *solution* (or value) for games than in the properties of a game itself. In this sense, Section 5 deals with more interesting applications of Theorem 5. (Most of the section is of independent interest, and can be read without a knowledge of recursion theory.)

Theorem 16 is our main contribution to the study of acyclic preference aggregation rules in the spirit of Nakamura's theorem [34] on the core of simple games. Banks [9], Truchon [43], and Andjiga and Mbih [3] are recent contributions to this literature. (Earlier papers on acyclic rules can be found in Truchon [43] and Austen-Smith and Banks [8].) Most works in this literature (including those just mentioned) consider finite sets of players. Nakamura [34] considers arbitrary (possibly infinite) sets of players and the algebra of all subsets of players. In contrast, we consider arbitrary sets of players and *arbitrary algebras* of coalitions.

Combining a simple game with a set of alternatives and a profile of individual preferences, we define a *simple game with (ordinal) preferences*. Nakamura's theorem [34] gives a necessary and sufficient condition for a simple game with preferences to have a nonempty core for all profiles: the number of alternatives is below a certain number, called the *Nakamura number* of the simple game. We extend (Theorem 16) Nakamura's theorem to the framework where simple games are defined on an arbitrary algebra of coalitions (so that not all subsets of players are coalitions). It turns out that our proof for the generalized result is more elementary than Nakamura's original proof; the latter is more complex than need be.

Since computable (nonweak) simple games have a finite winning coalition, we can easily prove that they have a finite Nakamura number (Corollary 15). Theorem 16 in turn implies (Corollary 17) that if a game is computable, the number of alternatives that the set of players can deal with rationally is restricted by this number. We conclude Section 5 with Proposition 18, which suggests the fundamental difficulty of obtaining computable aggregation rules in Arrow's setting [6], even after relaxing the transitivity requirement for (weak) social preferences. (Mihara [30, 31] studies computable aggregation rules *without* relaxing the transitivity requirement; these papers build on Armstrong [4, 5], who generalizes Kirman and Sondermann [23].)

2 Framework

2.1 Simple games

Let $N = \mathbf{N} = \{0, 1, 2, \ldots\}$ be a countable set of (the names of) players. Any **recursive** (algorithmically decidable) subset of N is called a **(recursive) coalition**.

Intuitively, a simple game describes in a crude manner the power distribution among *observable* (or describable) subsets of players. Since the cognitive ability of a human (or machine) is limited, it is not natural to assume that all subsets of players are observable when there are infinitely many players. We therefore assume that only **recursive** subsets are observable. This is a natural assumption in the present context, where algorithmic properties of simple games are investigated. According to *Church's thesis* [41, 35], the recursive coalitions are the sets of players for which there is an algorithm that can decide for the name of each player whether she is in the set.[8] Note that **the class** REC **of recursive coalitions** forms a **Boolean algebra**; that is, it includes N and is closed under union, intersection, and complementation. (We assume that observable coalitions are recursive, not just r.e. (*recursively enumerable*). Mihara [33, Remarks 1 and 16] gives three reasons: nonrecursive r.e. sets are observable in a very limited sense; the r.e. sets do not form a Boolean algebra; no satisfactory notion of computability can be defined if a simple game is defined on the domain of all r.e. sets.)

Formally, a **(simple) game** is a collection $\omega \subseteq$ REC of (recursive) coalitions. We will be explicit when we require that $N \in \omega$. The coalitions in ω are said to be **winning**. The coalitions not in ω are said to be **losing**. One can regard a simple game as a function from REC to $\{0, 1\}$, assigning the value 1 or 0 to each coalition, depending on whether it is winning or losing.

We introduce from the theory of cooperative games a few basic notions of simple games [36, 44].[9] A simple game ω is said to be **monotonic** if for all coalitions S and T, the conditions $S \in \omega$ and $T \supseteq S$ imply $T \in \omega$. ω is **proper** if for all recursive coalitions S, $S \in \omega$ implies $S^c := N \setminus S \notin \omega$. ω is **strong** if for all coalitions S, $S \notin \omega$ implies $S^c \in \omega$. ω is **weak** if $\omega = \emptyset$ or the intersection $\bigcap \omega = \bigcap_{S \in \omega} S$ of the winning coalitions is nonempty. The members of $\bigcap \omega$ are called **veto players**; they are the players that belong

[8] Soare [41] and Odifreddi [35] give a more precise definition of *recursive sets* as well as detailed discussion of recursion theory. Mihara's papers [30, 31] contain short reviews of recursion theory.

[9] The desirability of these properties depends, of course, on the context. Consider the team management example in Section 1.2, for example. Monotonicity makes sense, but may be too optimistic (adding a member may turn an acceptable team into an unacceptable one). Properness may be irrelevant or even undesirable (ensuring that a given task can be performed by two non-overlapping teams may be important from the viewpoint of reliability). This observation does not diminish the contribution of the main theorem (Theorem 5), which does not refer to these properties. In fact, one can show [26] that computability is "unrelated to" monotonicity, properness, strongness, and weakness.

to all winning coalitions. (The set $\bigcap \omega$ of veto players may or may not be observable.) ω is **dictatorial** if there exists some i_0 (called a **dictator**) in N such that $\omega = \{ S \in \mathrm{REC} : i_0 \in S \}$. Note that a dictator is a veto player, but a veto player is not necessarily a dictator.

We say that a simple game ω is **finitely anonymous** if for any finite permutation $\pi : N \to N$ (which permutes only finitely many players) and for any coalition S, we have $S \in \omega \iff \pi(S) \in \omega$. In particular, finitely anonymous games treat any two coalitions with the same finite number of players equally. Finite anonymity is a notion much weaker than the version of anonymity that allows any (measurable) permutation $\pi : N \to N$. For example, free ultrafilters (nondictatorial ultrafilters) defined below are finitely anonymous.

A **carrier** of a simple game ω is a coalition $S \subset N$ such that

$$T \in \omega \iff S \cap T \in \omega$$

for all coalitions T. We observe that if S is a carrier, then so is any coalition $S' \supseteq S$.

Finally, we introduce a few notions from the theory of Boolean algebras [24]; they can be regarded as properties of simple games. A monotonic simple game ω satisfying $N \in \omega$ and $\emptyset \notin \omega$ is called a **prefilter** if it has the finite intersection property: if $\omega' \subseteq \omega$ is finite, then $\bigcap \omega' \neq \emptyset$. Intuitively, a prefilter consists of "large" coalitions. A prefilter is **free** if and only if it is nonweak (i.e., it has no veto players). A free prefilter does not contain any finite coalitions (Lemma 13). A prefilter ω is a **filter** if it is closed with respect to finite intersection: if S, $S' \in \omega$, then $S \cap S' \in \omega$. The **principal filter generated by** S is $\omega = \{T \in \mathrm{REC} : S \subseteq T\}$. It is a typical example of a filter that is not free; it has a carrier, namely, S. A filter is a **principal filter** if it is the principal filter generated by some S. A filter ω is called an **ultrafilter** if it is a strong simple game. If ω is an ultrafilter, then $S \cup S' \in \omega$ implies that $S \in \omega$ or $S' \in \omega$. An ultrafilter is free if and only if it is not dictatorial.

2.2 Indicators for simple games

To define the notions of computability for simple games, we introduce below two indicators for them. In order to do that, we first represent each recursive coalition by a natural number: either by a characteristic index (Δ_0-index) or by an r.e. index (Σ_1-index). Here, a number e is a **characteristic index** for a coalition S if φ_e (the partial function computed by the Turing program with code number e) is the characteristic function for S. Intuitively, a characteristic index for a coalition describes the coalition by a Turing program that can decide its membership. A number e is an **r.e. index** for a coalition S if $S = W_e := \{ x : \varphi_e(x) \downarrow \}$, the domain of the eth partial recursive function φ_e. Intuitively, an r.e. index describes a coalition by a Turing program that will halt precisely when a member of the coalition is given as an input. A characteristic index gives more information about the coalition that it represents than an r.e. index does. The indicators then assign the value 0 or 1 to each number representing a coalition, depending on whether the coalition is winning or losing. When a number does not represent a recursive coalition, the value is undefined.

Given a simple game ω, its **δ-indicator** is the partial function δ_ω on $\mathbf{N}$ defined by

$$\delta_\omega(e) = \begin{cases} 1 & \text{if } e \text{ is a characteristic index for a recursive set in } \omega, \\ 0 & \text{if } e \text{ is a characteristic index for a recursive set not in } \omega, \\ \uparrow & \text{if } e \text{ is not a characteristic index for any recursive set.} \end{cases} \tag{1}$$

Note that δ_ω is well-defined since each $e \in \mathbf{N}$ can be a characteristic index (Δ_0-index) for at most one set. If we replace "a characteristic index" with "an r.e. index" in (1), we obtain the **σ-indicator** σ_ω instead of δ_ω.

2.3 Computability notions

We now introduce the notions of δ-computable simple games and σ-computable simple games.

We start by giving a scenario or intuition underlying the notion of δ-computability. (We can give a similar intuition for σ-computability.) A number (characteristic index) representing a coalition (equivalently, a Turing program that can decide the membership of a coalition) is presented by an inquirer to the aggregator (planner), who will compute whether the coalition is winning or not. Though there is no effective (algorithmic) procedure to decide whether a number given by the inquirer is legitimate (i.e., represents some recursive coalition), a human can often check manually (non-algorithmically) if such a number is a legitimate representation. We assume that the inquirer gives the aggregator only those indices that he has checked and proved its legitimacy. This assumption is justified if we assume that the aggregator always demands such proofs. The aggregator, however, cannot know a priori which indices will possibly be presented to her. (There are, of course, indices unlikely to be used by humans. But the aggregator cannot a priori rule out some of the indices.) So, *the aggregator should be ready to compute whenever a legitimate representation is presented to her.* This intuition justifies the following conditions of computability.

δ-computability δ_ω has an extension to a partial recursive function.

σ-computability σ_ω has an extension to a partial recursive function.

Instead of, say, δ-computability, one might want to require the indicator δ_ω *itself* (or its extension that gives a number different from 0 or 1 whenever $\delta_\omega(e)$ is undefined) to be partial recursive [33, Appendix A]. Such a condition cannot be satisfied, however, since the domain of δ_ω is not r.e. [30, Lemma 2]. Since δ-indicators use more descriptive indices than σ-indicators, σ-computability implies δ-computability [33, Lemma 2].

Remark 1. *Multiple-choice or essay.* One might argue that the scenario preceding the definitions of computability makes the aggregator's task more difficult than need be. The difficulty comes from the fact that, like an essay exam, there is too much freedom on the side of the inquirer, the argument would go, in the sense that each recursive coalition has infinitely many indices and that the index presented may be an illegitimate one. An alternative notion of computability that deals with these problems might use a "multiple-choice format," in which the aggregator gives possible indices that the inquirer can choose from. Unfortunately, such a "multiple-choice format" would not work as one might wish.

Indeed, we claim that *there is no effective listing* e_0, e_1, e_2, ... *of characteristic indices such that for each recursive coalition* S *there is at least one* e_i *that represents the coalition (i.e.,* e_i *is a characteristic index for* S*).* To prove this claim, suppose there is such a listing and let S be the set defined by $i \in S$ if and only if $\varphi_{e_i}(i) = 0$. Then since $\varphi_{e_i}(i)\downarrow$ for any i, we have S recursive. On the other hand, the characteristic function for S is not equal to any φ_{e_i}. To see this, suppose that it is equal to φ_{e_i}. Then, if $i \in S$, we have $\varphi_{e_i}(i) = 1$, the definition of S then implies $i \notin S$, a contradiction; if $i \notin S$, we have a similar contradiction. The claim is thus proved.

Given this impossibility result, one might wish to relax the condition and allow some e_i in the listing to fail to be a characteristic index. Adopting a notion of computability based on such a listing is a halfway solution, fitting into neither the essay-exam scenario nor the multiple-choice alternative to it. ‖

The following results suggest that σ-computability is too strong a notion of computability—due to lack of the descriptive power of r.e. indices.

Proposition 1 (Mihara [33, Proposition 3]) *Suppose that* $N \in \omega$ *and* $\emptyset \notin \omega$*. Then the simple game* ω *is not* σ*-computable. In particular, proper simple games (for which* N *is winning) violate the condition.*

Corollary 2 *Suppose that* $N \in \omega$*. If the simple game* ω *is* σ*-computable, then it is not proper. Furthermore, if it is monotonic, then* $\omega = \mathrm{REC}$*; that is, all coalitions are winning.*

For example, even *dictatorial* simple games are not σ-computable. This is because, if the dictator is not in the coalition represented by a given r.e. index e, one cannot generally be assured that he is not, because the

Turing program with the code number e will never halt when the name of the dictator is given as an input.

We therefore focus on δ-computability in the remaining sections of the paper, discarding σ-computability.

3 A Characterization Result

Proposition 1 and its corollary indicate that σ-computable simple games have a rather uninteresting structure. We therefore investigate δ-computability in the rest of the paper.

3.1 Determining strings

The next lemma states that for any coalition S of a δ-computable simple game, there is a cutting number k such that any *finite* coalition G having the same k-players as S (that is, G and S are equal if players $i \geq k$ are ignored) is winning (losing) if S is winning (losing). Note that if k is such a cutting number, then so is any k' greater than k.

Notation. We identify a natural number k with the finite set $\{0, 1, 2, \ldots, k-1\}$, which is an initial segment of $\mathbf{N}$. Given a coalition $S \subseteq N$, we write $S \cap k$ to represent the coalition $\{i \in S : i < k\}$ consisting of the members of S whose name is less than k. We call $S \cap k$ the **k-initial segment of** S, and view it either as a subset of $\mathbf{N}$ or as the string $S[k]$ of length k of 0's and 1's (representing the restriction of its characteristic function to $\{0, 1, 2, \ldots, k-1\}$). Note that if G is a coalition and $G \cap k = S \cap k$ (that is, G and S are equal if players $i \geq k$ are ignored), the characteristic function of G extends the k-initial segment (viewed as a string of 0's and 1's) of S.

Lemma 3 *Let ω be a δ-computable simple game. If $S \in \omega$, then there is an initial segment $k \geq 0$ of $\mathbf{N}$ such that for any* finite $G \in \mathrm{REC}$, *if $G \cap k = S \cap k$, then $G \in \omega$. Similarly, if $S \notin \omega$, then there is an initial segment $k \geq 0$ of $\mathbf{N}$ such that for any* finite $G \in \mathrm{REC}$, *if $G \cap k = S \cap k$, then $G \notin \omega$.*

Proof. Let $S \in \omega$ and assume for a contradiction that there is no such initial segment k. Then, for each initial segment k of $\mathbf{N}$, there is a finite coalition G_k such that $G_k \cap k = S \cap k$ and $G_k \notin \omega$. Note that we can find such G_k recursively (algorithmically) in k since it is finite.

Let K be a nonrecursive r.e. set such as $\{\, e : e \in W_e \,\}$. Since K is r.e., there is a recursive set $R \subseteq \mathbf{N} \times \mathbf{N}$ such that $e \in K \Leftrightarrow \exists z R(e, z)$. Define $g(e, u) = \mu y \leq u \; R(e, y)$ (i.e., the least $y \leq u$ such that $R(e, y)$) if such y exists, and $g(e, u) = 0$ otherwise. Then g is recursive.

Using the Parameter Theorem, define a recursive function f by

$$\varphi_{f(e)}(u) = 1 \qquad \text{if } \neg\exists z \leq u \; R(e, z) \text{ and } u \in S,$$

$$\varphi_{f(e)}(u) = 0 \qquad \text{if } \neg\exists z \leq u\ R(e,z) \text{ and } u \notin S,$$
$$\varphi_{f(e)}(u) = 1 \qquad \text{if } \exists z \leq u\ R(e,z) \text{ and } u \in G_{g(e,u)}, \text{ and}$$
$$\varphi_{f(e)}(u) = 0 \qquad \text{otherwise.}$$

Now, on the one hand, $e \in K$ implies that $f(e)$ is a characteristic index for $G_{u'} \notin \omega$ for some u'. (*Details*: Given $e \in K$, let $u' = \mu y\ R(e,y)$, which is well-defined since $\exists z R(e,z)$. Then $\varphi_{f(e)}(u) = 1$ iff (i) $u < u'$ and $u \in S$ [that is, $u < u'$ and $u \in G_{u'}$] or (ii) $u \geq u'$ and [since $g(e,u) = u'$ in this case] $u \in G_{g(e,u)} = G_{u'}$. Thus $\varphi_{f(e)}(u) = 1$ iff $u \in G_{u'}$.) Hence $\delta_\omega(f(e)) = 0$. On the other hand, $e \notin K$ implies that $f(e)$ is a characteristic index for $S \in \omega$. Hence $\delta_\omega(f(e)) = 1$.

Since δ_ω has an extension to a p.r. function (because ω is δ-computable), the last paragraph implies that K is recursive. This is a contradiction.

To prove the last half of the lemma, note that the set-theoretic difference $\hat{\omega} = \text{REC} - \omega$ is also a δ-computable simple game. Let $S \notin \omega$. Then the first half applies to $\hat{\omega}$ and $S \in \hat{\omega}$. Since $G \in \hat{\omega}$ iff $G \notin \omega$, the desired result follows. ∎

In fact, the coalition G in Lemma 3 need not be finite. Before stating an extension (Proposition 4) of Lemma 3, we introduce the notion of *determining strings*:

Definition 1. Consider a simple game. A string τ (of 0's and 1's) of length $k \geq 0$ is said to be **determining** if either any coalition $G \in \text{REC}$ extending τ (in the sense that τ is an initial segment of G, i.e., $G \cap k = \tau$) is winning or any coalition $G \in \text{REC}$ extending τ is losing. A string τ is said to be **determining for finite coalitions** if either any finite coalition G extending τ is winning or any finite coalition G extending τ is losing. A string is called **nondetermining** if it is not determining.

Proposition 4 below states that for δ-computable simple games, (the characteristic function for) every coalition S has an initial segment $S \cap k$ that is determining. (The number $k - 1$ may be greater than the greatest element, if any, of S):

Proposition 4 *Suppose that a δ-computable simple game is given.* (i) *If a coalition S is winning, then there is an initial segment $k \geq 0$ of* $\mathbf{N}$ *such that for* any *(finite or infinite) coalition G, if $G \cap k = S \cap k$, then G is winning.* (ii) *If S is losing, then there is an initial segment $k \geq 0$ of* $\mathbf{N}$ *such that for* any *coalition G, if $G \cap k = S \cap k$, then G is losing.* (iii) *If $S \cap k$ is an initial segment that is determining* for finite coalitions, *then $S \cap k$ is an initial segment that is determining.*

Proof. It suffices to prove (i). As a byproduct, we obtain (iii).

Suppose $S \in \omega$, where ω is a δ-computable simple game. Then by the first half of Lemma 3, there is $k \geq 0$ such that (a) for any finite G', if $G' \cap k = S \cap k$, then $G' \in \omega$.

To obtain a contradiction, suppose that there is $G \notin \omega$ such that (b) $G \cap k = S \cap k$. By the last half of Lemma 3, there is $k' \geq 0$ such that (c) for any finite G', if $G' \cap k' = G \cap k'$, then $G' \notin \omega$. Without loss of generality, assume $k' \geq k$.

Consider $G' = G \cap k'$, which is finite. Then, on the one hand, since $G' \cap k' = G \cap k'$, we get $G' \notin \omega$ by (c). On the other hand, since $k' \geq k$, we get $G' \cap k = G \cap k = S \cap k$ (the last equality by (b)). Then (a) implies that $G' \in \omega$. This is a contradiction. ∎

3.2 Characterization of computable games

The next theorem characterizes δ-computable simple games in terms of sets of determining strings. Roughly speaking, finitely many players determine whether a coalition is winning or losing. Though we cannot tell in advance which finite set of players determines that, we can list such sets in an effective manner.

Note that $T_0 \cup T_1$ in the theorem does not necessarily contain all determining strings. (The $\Longrightarrow$ direction can actually be strengthened: we can find *recursive*, not just r.e., sets T_0 and T_1 satisfying the conditions. We do not prove the strengthened result, since we will not use it.)

Theorem 5 *A simple game ω is δ-computable if and only if there are an r.e. set T_0 of losing determining strings and an r.e. set T_1 of winning determining strings such that (the characteristic function for) any coalition has an initial segment in T_0 or in T_1.*

Remark 2. We can derive Theorem 5 from a result in Kreisel et al. [25] and Ceĭtin [14]. In this remark we largely follow the terminology of Odifreddi [35, pages 186–192 and 205–210], who gives a topological argument. In this remark only, a *string* refers to a finite sequence $\sigma = \sigma(0)\sigma(1)\cdots\sigma(k)$ of natural numbers (not necessarily 0 or 1).

Let $\mathcal{PR}$ be the class of partial recursive (unary) functions and $\mathcal{R}$ the class of recursive functions. An *effective operation on* $\mathcal{R}$ is a functional (function) $F\colon \mathcal{R} \to \mathcal{R}$ such that for some partial recursive function ψ,

$$\varphi_e \in \mathcal{R} \Longrightarrow [\psi(e)\downarrow \text{ and } F(\varphi_e) = \varphi_{\psi(e)}].$$

We introduce a topology into the set of partial (unary) functions by viewing it as a product space $S^{\mathbf{N}}$, with $S = \mathbf{N} \cup \{\uparrow\}$, $\uparrow$ being a distinguished element for the undefined value. For a string σ, let $A_\sigma = \{f \in \mathcal{R} :$

if $\sigma(x)\downarrow$, then $f(x)=\sigma(x)\}$ be the set of recursive functions that extend σ. These sets A_σ are the basic open sets. Let t be a recursive bijection between the set $\mathbf{N}$ of natural numbers and the set of strings. We say a continuous functional $F\colon \mathcal{PR}\to\mathcal{PR}$ is *effectively continuous on* $\mathcal{R}$ if F maps $\mathcal{R}$ to $\mathcal{R}$ and for some recursive function ψ,

$$F^{-1}(A_\sigma)=\{f : f\in A_\nu \text{ for some } \nu\in\{t(a) : a\in W_{\psi(t^{-1}(\sigma))}\}\} \tag{2}$$

(requiring that the open sets $F^{-1}(A_\sigma)$ be obtained in a certain effective way).

Kreisel et al. [25] and Ceĭtin [14] prove the theorem [35, Theorem II.4.6] stating that *the effective operations on* $\mathcal{R}$ *are exactly the restrictions of the effectively continuous functionals on* $\mathcal{R}$.

In our context, suppose that ω is a δ-computable simple game. Let δ' be a p.r. extension of δ_ω. Further let δ'' be such that $\delta'(x)\uparrow\Leftrightarrow\delta''(x)\uparrow$ and $\delta'(x)=i\Leftrightarrow\delta''(x)=e_i$, where for each $i\in\mathbf{N}$, e_i is an index of the constant (recursive) function whose value is always i. By the *s-m-n* theorem, define a recursive function g such that $\varphi_{g(e)}(x)$ is 1, 0, or undefined, depending on whether $\varphi_e(x)$ is positive, zero, or undefined. In particular, if e is a characteristic index, then $g(e)$ is a characteristic index and $\varphi_{g(e)}=\varphi_e$. Define F on $\mathcal{R}$ by $F(\varphi_e)=\varphi_{\delta''(g(e))}$. Then F is an effective operation on $\mathcal{R}$ (depending on whether $g(e)$ is a characteristic index for a winning coalition or a losing coalition, $F(\varphi_e)=\varphi_{e_1}$ or $F(\varphi_e)=\varphi_{e_0}$). By the theorem above, F is effectively continuous on $\mathcal{R}$ so that for some recursive ψ, (2) holds. Denote by A_i the set A_σ where $\sigma=\sigma(0)=i$. Since F maps any $\varphi_e\in\mathcal{R}$ into constant functions $\varphi_{e_1}\in A_1$ and $\varphi_{e_0}\in A_0$, we have $F^{-1}(\{\varphi_{e_i}\})=F^{-1}(A_i)$ for $i\in\{0,1\}$. We therefore have φ_e in $F^{-1}(A_1)$ or in $F^{-1}(A_0)$, depending on whether e is a characteristic index for a winning coalition or a losing coalition. The $\Longrightarrow$ direction of Theorem 5 is obtained by letting T_i be the r.e. set $\{t(a) : a\in W_{\psi(t^{-1}(i))}\}$ restricted to the 0-1 strings.

In this paper, we choose to give a different proof, which is perhaps more self-contained. The fact that the proof uses the recursion theorem should also be of some interest. $\|$

Proof. ($\Longleftarrow$). We give an algorithm that can decide for each coalition whether it is winning or not: Given is a characteristic index e of a coalition S. Generate the elements of T_0 and T_1; we can do that effectively since these sets are r.e. Wait until an initial segment of S is generated. (Since a characteristic index is given, we can decide whether a string generated is an initial segment of S.) If the initial segment is in T_0, then S is losing; if it is in T_1, then S is winning.

($\Longrightarrow$). Suppose ω is δ-computable. Let δ' be a p.r. extension of δ_ω; such a δ' exists since ω is δ-computable.

Overview. From Proposition 4, our goal is to *effectively enumerate* a determining initial segment $S \cap k$ of each losing coalition S in T_0 and that of each winning coalition in T_1.

We will define a certain recursive function $y(e)$ in Step 1. In Step 2, we will first define the sets T_i, where $i \in \{0, 1\}$, as the collection of certain strings (of 0's and 1's) of length $k(e)$ (to be defined) for those $e \in \mathbf{N}$ satisfying $\delta'(y(e)) = i$. In particular, $T_0 \cup T_1$ includes, for each characteristic index e, the $k(e)$-initial segment of the recursive coalition indexed by e. We will then show that T_0 and T_1 satisfy the conditions stated.

We use the following **notation**. We write $\varphi_{e,s}(x) = y$ if x, y, $e < s$ and y is the output of $\varphi_e(x)$ in less than s steps of the eth Turing program [41, p. 16]. We fix a Turing program for δ' and denote by $\delta'_s(y)$ the computation of $\delta'(y)$ up to step s of the program.

Step 1. Defining a recursive function $y(e)$.

We define a recursive function $f(e, y)$ in Step 1.1. In Step 1.2, we apply a variant of the Recursion Theorem to $f(e, y)$ and obtain $y(e)$.

Step 1.1. Defining a recursive function $f(e, y)$. Define an r.e. set $Q_0 \subseteq \mathbf{N}$ by $y \in Q_0$ iff there exists s such that $\delta'_s(y) = 0$ or $\delta'_s(y) = 1$. Define a p.r. function

$$s_0(y) = \mu s[\delta'_s(y) \in \{0, 1\}],$$

which converges for $y \in Q_0$.

Fix a recursive set $\mathbf{F}$ of characteristic indices for finite sets such that each finite set has at least one characteristic index in $\mathbf{F}$. (An example of $\mathbf{F}$ is the set consisting of the code numbers (Gödel numbers) of the Turing programs of a particular form.) For $s \in \mathbf{N}$, let $\mathbf{F}_s = \mathbf{F} \cap s$ be the finite set of numbers $e < s$ in $\mathbf{F}$.

Define a set $Q_1 \subseteq \mathbf{N} \times \mathbf{N}$ by $(e, y) \in Q_1$ iff (i) $y \in Q_0$ and (ii) there exist $s' \geq s_0 := s_0(y)$ and $e' \in \mathbf{F}_{s'}$ such that (ii.a) $\delta'_{s'}(e') = 1 - \delta'_{s_0}(y)$ and that (ii.b) $\varphi_{e',s'}$ is an extension of φ_{e,s_0-1}. (Condition (ii.b), written $\varphi_{e',s'} \supseteq \varphi_{e,s_0-1}$, means that if $\varphi_{e,s_0-1}(z) = u$, then $\varphi_{e',s'}(z) = u$.) Note that if $(e, y) \in Q_1$, then $s_0 = s_0(y)$ is defined and $\delta'_{s_0}(y) \in \{0, 1\}$. We can easily check that Q_1 is r.e. Given $(e, y) \in Q_1$, let s_1 be the least $s' \geq s_0$ such that conditions (ii.a) and (ii.b) hold for some $e' \in \mathbf{F}_{s'}$. Let e_0 be the least $e' \in \mathbf{F}_{s'}$ such that conditions (ii.a) and (ii.b) hold for $s' = s_1$. We can view e_0 as a p.r. function $e_0(e, y)$, which converges for $(e, y) \in Q_1$.

Define the partial function ψ by

$$\psi(e, y, z) = \begin{cases} \varphi_{e_0}(z) & \text{if } y \in Q_0 \text{ and } (e, y) \in Q_1, \\ \varphi_{e,s_0-1}(z) & \text{if } y \in Q_0 \text{ and } (e, y) \notin Q_1, \\ \varphi_e(z) & \text{if } y \notin Q_0. \end{cases}$$

Lemma 6 *ψ is p.r.*

Proof. We show there is a sequence of p.r. functions ψ^s such that $\psi = \bigcup_s \psi^s$. We then apply the Graph Theorem to conclude ψ is p.r.

For each $s \in \mathbf{N}$, define a recursive set $Q_0^s \subseteq \mathbf{N}$ by $y \in Q_0^s$ iff there exists $s' \leq s$ such that $\delta'_{s'}(y) = 0$ or $\delta'_{s'}(y) = 1$. We have $y \in Q_0$ iff $y \in Q_0^s$ for some s. Note that if $y \in Q_0^s$ for some y, then $s \geq s_0$.

For each $s \in \mathbf{N}$, define a recursive set $Q_1^s \subseteq \mathbf{N} \times \mathbf{N}$ by $(e, y) \in Q_1^s$ iff (i) $y \in Q_0^s$ and (ii) there exist s' such that $s_0 := s_0(y) \leq s' \leq s$ and $e' \in \mathbf{F}_{s'}$ such that (ii.a) $\delta'_{s'}(e') = 1 - \delta'_{s_0}(y)$ and that (ii.b) $\varphi_{e',s'}$ is an extension of φ_{e,s_0-1}. (Conditions (ii.a) and (ii.b) are the same as those in the definition of Q_1.) We have $(e, y) \in Q_1$ iff $(e, y) \in Q_1^s$ for some s. Note that if $(e, y) \in Q_1^s$ for some (e, y), then $s_0 \leq s_1 \leq s$.

For each $s \in \mathbf{N}$, define the p.r. function ψ^s by

$$\psi^s(e, y, z) = \begin{cases} \varphi_{e_0,s}(z) & \text{if } y \in Q_0^s \text{ and } (e, y) \in Q_1^s, \\ \varphi_{e,s_0-1}(z) & \text{if } y \in Q_0^s \text{ and } (e, y) \notin Q_1^s, \\ \varphi_{e,s}(z) & \text{if } y \notin Q_0^s. \end{cases}$$

We claim that $\bigcup_s \psi^s$ is a partial function (i.e., $\bigcup_s \psi^s(e, y, z)$ does not take more than one value) and that $\psi = \bigcup_s \psi^s$:

- Suppose $y \notin Q_0$. Then $y \notin Q_0^s$ for any s. So, for all s, $\psi^s(e, y, z) = \varphi_{e,s}(z)$. Hence $\bigcup_s \psi^s(e, y, z) = \varphi_e(z) = \psi(e, y, z)$ as desired.

- Suppose ($y \in Q_0$ and) $(e, y) \in Q_1$. Then s_0, s_1, and e_0 are defined and $s_1 \geq s_0$. If $s < s_0$, then since $y \notin Q_0^s = \emptyset$, we have $\psi^s(e, y, z) = \varphi_{e,s}(z)$. If $s_0 \leq s < s_1$, then since $y \in Q_0^s$ and $(e, y) \notin Q_1^s = \emptyset$, we have $\psi^s(e, y, z) = \varphi_{e,s_0-1}(z)$. If $s_1 \leq s$, then since $y \in Q_0^s$ and $(e, y) \in Q_1^s$, we have $\psi^s(e, y, z) = \varphi_{e_0,s}(z)$. Hence $\bigcup_s \psi^s(e, y, z) = \varphi_{e,s_0-1}(z) \cup (\bigcup_{s \geq s_1} \varphi_{e_0,s}(z))$. The definition of s_1 implies that when $s_1 \leq s$, $\varphi_{e,s_0-1} \subseteq \varphi_{e_0,s_1} \subseteq \varphi_{e_0,s}$. Thus $\bigcup_s \psi^s(e, y, z) = \bigcup_{s \geq s_1} \varphi_{e_0,s}(z) = \varphi_{e_0}(z) = \psi(e, y, z)$ as desired.

- Suppose $y \in Q_0$ and $(e, y) \notin Q_1$. Then s_0 is defined. If $s < s_0$, then since $y \notin Q_0^s = \emptyset$, we have $\psi^s(e, y, z) = \varphi_{e,s}(z)$. If $s_0 \leq s$, then since $y \in Q_0^s$ and $(e, y) \notin Q_1^s$, we have $\psi^s(e, y, z) = \varphi_{e,s_0-1}(z)$. Hence $\bigcup_s \psi^s(e, y, z) = \varphi_{e,s_0-1}(z) = \psi(e, y, z)$ as desired.

Define the partial function $\hat{\psi}$ by $\hat{\psi}(s, e, y, z) = \psi^s(e, y, z)$. Then from the construction of ψ^s, $\hat{\psi}$ is p.r. by Church's Thesis. By the Graph Theorem, the graph of $\hat{\psi}$ is r.e.

We claim that $\psi = \bigcup_s \psi^s$ is p.r. By the Graph Theorem it suffices to show that its graph is r.e. We have $(e, y, z, u) \in \psi \iff \exists s \ (e, y, z, u) \in \psi^s \iff \exists s \ (s, e, y, z, u) \in \hat{\psi}$. Since the graph of $\hat{\psi}$ is r.e., it follows that the graph of ψ is r.e. $\|$

Since ψ is p.r., there is a recursive function $f(e, y)$ such that $\varphi_{f(e,y)}(z) = \psi(e, y, z)$ by the Parameter Theorem.

Step 1.2. Applying the Recursion Theorem to obtain $y(e)$. Since $f(e, y)$ is recursive, by the Recursion Theorem with Parameters [41, p. 37] there is a recursive function $y(e)$ such that $\varphi_{y(e)} = \varphi_{f(e,y(e))}$. So, we have $\varphi_{y(e)}(z) = \psi(e, y(e), z)$.

We claim that the $y = y(e)$ cannot meet the first case ($y \in Q_0$ and $(e, y) \in Q_1$) in the definition of ψ. Suppose $y(e) \in Q_0$ and $(e, y(e)) \in Q_1$. Since $\varphi_{y(e)}(z) = \psi(e, y(e), z)$, by the definition of ψ we have on the one hand $\varphi_{y(e)} = \varphi_{e_0}$. By (ii.a) of the definition of Q_1 and by the definition of e_0, we have on the other hand $\delta'(e_0) = 1 - \delta'(y(e)) \neq \delta'(y(e))$. This contradicts the fact that δ' extends the δ-indicator δ_ω.

Therefore, we can express $\varphi_{y(e)}(z) = \psi(e, y(e), z)$ as follows:

$$\varphi_{y(e)}(z) = \begin{cases} \varphi_{e,s_0-1}(z) & \text{if } y(e) \in Q_0 \ ((e, y(e)) \notin Q_1 \text{ implied}), \\ \varphi_e(z) & \text{if } y(e) \notin Q_0. \end{cases} \tag{3}$$

Step 2 Defining T_0 and T_1 and verifying the conditions.

For $i \in \{0, 1\}$, let T_i be the collection of all the strings τ of length $k(e) := s_0 - 1$ (where $s_0 = s_0(y(e))$) that extends φ_{e,s_0-1} for all those e such that $\delta'(y(e)) = i$. (Note that $\delta'(y(e)) \in \{0, 1\}$ iff $y(e) \in Q_0$.) We show T_0 and T_1 satisfy the conditions.

Step 2.1. T_0 and T_1 are r.e. This is obvious since s_0, δ', and y are p.r. (In other words, for each e, first find whether $\delta'(y(e)) \in \{0, 1\}$. If not, we do not enumerate any segment in T_0 or T_1. If $\delta'(y(e)) = i \in \{0, 1\}$, then we have corresponding strings whose length is effectively obtained. So we enumerate them in T_i. This procedure ensures that T_0 and T_1 are r.e.)

Step 2.2. T_0 and T_1 consist of determining strings. We show that T_0 consists of losing determining strings. We can show that T_1 consists of winning determining strings in a similar way.

Suppose $\delta'(y(e)) = 0$. Then $y(e) \in Q_0$. Let $s_0 = s_0(y(e))$ and $k(e) = s_0 - 1$. Since $(e, y(e)) \notin Q_1$ by (3), there is no $e' \in \mathbf{F}$ such that $\delta'(e') = 1 - \delta'(y(e)) = 1$ and that $\varphi_{e'}$ is an extension of φ_{e,s_0-1}. (Note that $\delta'(e') \in \{0, 1\}$ if $e' \in \mathbf{F}$.) Hence any finite coalition that extends φ_{e,s_0-1} is losing.

Therefore, all strings τ in T_0 (i.e., all the finite strings of length $k(e)$ that extend φ_{e,s_0-1} for some e such that $\delta'(y(e)) = 0$) are losing determining for *finite* coalitions. By Proposition 4 (iii), all strings τ in T_0 are losing determining strings.

Step 2.3. Any coalition has an initial segment in $T_0 \cup T_1$. Let S be a coalition, which is recursive. Pick a characteristic index e for S. We first show that $\delta'(y(e)) \in \{0, 1\}$ (i.e., $y(e) \in Q_0$). Suppose $y(e) \notin Q_0$. By

(3), we have $\varphi_{y(e)} = \varphi_e$. So $y(e)$ is a characteristic index for S. Hence $\delta'(y(e)) \in \{0,1\}$. That is, $y(e) \in Q_0$, which is a contradiction.

By the definitions of T_0 and T_1, since $\delta'(y(e)) \in \{0,1\}$, all the strings of length $k(e) = s_0 - 1$ extending φ_{e,s_0-1} are in $T_0 \cup T_1$. In particular, since the $k(e)$-initial segment $S \cap k(e)$ of the characteristic function φ_e for S extends φ_{e,s_0-1}, the initial segment $S \cap k(e)$ is in $T_0 \cup T_1$. ∎

4 Applications: Finite Carriers, Finite Winning Coalitions, Prefilters, and Nonanonymity

Theorem 5 is a powerful theorem. We can obtain as its corollaries some of the results in Mihara [33].

4.1 Finite carriers

The following proposition asserts that games that are essentially finite satisfy δ-computability, as might be expected. We give here a proof that uses the characterization theorem.

Proposition 7 (Mihara [33, Proposition 5]) *Suppose that a simple game ω has a finite carrier. Then ω is δ-computable.*

Proof. Suppose that ω has a finite carrier S. Let $k = \max S + 1$ (we let $k = 0$ if $S = \emptyset$). Let

$$T_1 = \{\tau : \tau \text{ is a string of length } k \text{ and } \tau \in \omega\}$$

(where $\tau \in \omega$ means that the set $\{i < k : \tau(i) = 1\}$ represented by τ, viewed as a characteristic function, is in ω) and $T_0 = \{\tau : \tau \text{ is a string of length } k \text{ and } \tau \notin \omega\}$. We verify the conditions of Theorem 5.

Since T_0 and T_1 are finite, they are r.e. Since $T_0 \cup T_1$ consists of all strings of length k, any coalition has an initial segment in it.

We show that T_1 consists of winning determining strings. (We can show that T_0 consists of losing determining strings in a similar way.) Suppose $G \cap k = \tau \in T_1$. It suffices to show that $G \in \omega$. By the definition of T_1, we have $\tau \in \omega$. This implies $\tau \cap S \in \omega$ since S is a carrier. Since $\tau \cap S = (G \cap k) \cap S = G \cap (k \cap S) = G \cap S$, it follows that $G \cap S \in \omega$. Since S is a carrier, we get $G \in \omega$. ∎

In particular, *if a simple game ω is dictatorial, then ω is δ-computable.* Indeed, the coalition consisting of the dictator is a finite carrier for the dictatorial game ω.

4.2 Finite winning coalitions

Note in Proposition 7 that if a game has a finite carrier S and N is winning, then there exists a finite winning coalition, namely $S = N \cap S$. When there does not exist a finite winning coalition, it is a corollary of the following negative result—itself a corollary of Theorem 5—-that the computability condition is violated.

Proposition 8 (Mihara [33, Proposition 6]) *Suppose that a simple game ω has an infinite winning coalition $S \in \omega$ such that for each $k \in \mathbf{N}$, its k-initial segment $S \cap k$ is losing. Then ω is not δ-computable.*

Proof. Suppose that ω is δ-computable. Since S is winning, by Lemma 3 (or by Proposition 4 or by Theorem 5) there is some $k \in \mathbf{N}$ such that $G = S \cap k$ is winning. This contradicts the assumption of the proposition. ∎

Theorem 5 immediately gives the following extension of Corollary 7 of Mihara [33]. It gives a useful criterion for checking computability of simple games. Here, a *cofinite* set is the complement of a finite set.

Proposition 9 *Suppose that a δ-computable simple game has a winning coalition. Then, it has both finite winning coalitions and cofinite winning coalitions.*

We also prove a result that is close to Proposition 8.

Proposition 10 (Mihara [33, Proposition 8]) *Suppose that $\emptyset \notin \omega$. Suppose that the simple game ω has an infinite coalition $S \in \omega$ such that for each $k \in \mathbf{N}$, its difference $S \setminus k = \{ s \in S : s \geq k \}$ from the initial segment is winning. Then ω is not δ-computable.*

Proof. Suppose ω is δ-computable. Since $\emptyset \notin \omega$, there is a losing determining string $\tau = 00 \cdots 0$ of length k by Theorem 5. By assumption, $S \setminus k$ is winning. But $(S \setminus k) \cap k = \tau$ and that τ is a losing determining string imply that $S \setminus k$ is losing, which is a contradiction. ∎

Again, the following proposition gives a useful criterion for checking computability of simple games.

Proposition 11 *Suppose that a δ-computable simple game has a losing coalition. Then, it has both finite losing coalitions and cofinite losing coalitions.*

4.3 Prefilters, filters, and ultrafilters

From the propositions in Section 4.2, examples of a noncomputable simple game are easy to come by.

Example 1. For any q satisfying $0 < q \leq \infty$, let ω be the ***q*-complement rule** defined as follows: $S \in \omega$ if and only if $\#(N \setminus S) < q$. For example, if $q = 1$, then the q-complement rule is the unanimous game, consisting of N alone. If $q = \infty$, then the game consists of cofinite coalitions (the complements of finite coalitions). Proposition 9 implies that *q-complement rules are not δ-computable*, since they have no finite winning coalitions. Any q-complement rule is a prefilter and it is a monotonic, proper, nonstrong, and anonymous simple game. If $q > 1$, it is nonweak, but any finite intersection of winning coalitions is nonempty (i.e., has an infinite Nakamura number, to be defined in Section 5). Note that if $1 < q < \infty$, then the q-complement rule is not a filter since it is not closed with respect to finite intersection. ‖

Example 1 gives examples of a prefilter that is not a filter. It also gives two examples of a filter that is not an ultrafilter: the unanimous game is a principal filter and the game consisting of all cofinite coalitions is a nonprincipal filter. Mihara [32] gives a constructive example of an ultrafilter.

Some prefilters are computable, but that is true only if they have a veto player: according to Proposition 18 below, if a prefilter is δ-computable, then it is weak.

If ω is a filter, then it is δ-computable if and only if it is has a finite carrier [33, Corollary 11]. In particular, the principal filter $\omega = \{T \in \text{REC} : S \subseteq T\}$ generated by S has a carrier, namely, S. So, if S is finite, the principal filter is computable. If S is infinite, it is noncomputable, since it does not have a finite winning coalition. For example, the principal filter generated by $S = 2\mathbf{N} := \{0, 2, 4, \ldots\}$ is a monotonic, proper, nonstrong, weak, and noncomputable simple game.

If ω is a nonprincipal ultrafilter, it is not δ-computable by Proposition 9, since it has no finite winning coalitions (or no cofinite losing coalitions). It is a monotonic, proper, strong, and nonweak noncomputable simple game. In fact, an ultrafilter is δ-computable if and only if it is dictatorial [30, Lemma 4].

4.4 Nonanonymity

As an application of the characterization result in Section 3, we show that δ-computable simple games violate finite anonymity, a weak notion of anonymity. Proposition 12 below strengthens an earlier result [33, Corollary 12] about computable games. The latter result assumes proper, monotonic, δ-computable simple games, instead of just δ-computable simple games.

Proposition 12 *Suppose that $N \in \omega$ and $\emptyset \notin \omega$. If the simple game ω is δ-computable, then it is not finitely anonymous.*

Proof. Let ω be a finitely anonymous δ-computable simple game such that $N \in \omega$ and $\emptyset \notin \omega$. Since $N \in \omega$, there is an initial segment $k := \{0, 1, \ldots, k-1\} = 11\cdots 1$ (string of 1's of length k), by Lemma 3 (or by Proposition 4 or by Theorem 5). Since $\emptyset \notin \omega$, there is a losing determining string $\tau = 00\cdots 0$ of length k' by Theorem 5. Then the concatenation $\tau * k = 00\cdots 011\cdots 1$ of τ and k, viewed as a set, has the same number of elements as k. Since coalitions $\tau * k$ and k are finite and have the same number of elements, they should be treated equally by the finitely anonymous ω. But $\tau * k$ is losing and k is winning. ∎

5 The Number of Alternatives and the Core

In this section, we apply Theorem 5 to a social choice problem. We show (Corollary 17) that computability of a simple game entails a restriction on the number of alternatives that the set of players (with the coalition structure described by the simple game) can deal with rationally.

For that purpose, we define the notion of a *simple game with (ordinal) preferences*, a combination of a simple game and a set of alternatives and individual preferences. After defining the *core* for simple games with preferences, we extend (Theorem 16) Nakamura's theorem [34] about the nonemptyness of the core: the core of a simple game with preferences is always (i.e., for all profiles of preferences) nonempty if and only if the number of alternatives is finite and below a certain critical number, called the *Nakamura number* of the simple game. We need to do this extension since what we call a "simple game" is not generally what is called a "simple game" in Nakamura [34].

We show (Corollary 15) that the Nakamura number of a nonweak simple game is finite if it is computable, though (Proposition 14) there is no upper bound for the set of the Nakamura numbers of such games. It follows from Theorem 16 that (Corollary 17) in order for a set of alternatives to always have a maximal element given a nonweak, computable game, the number of alternatives must be restricted. In contrast, *some* noncomputable (and nonweak) simple games do not have such a restriction (Proposition 18), and in fact have some nice properties. These results have implications for social choice theory; we suggest its connection with the study of Arrow's Theorem [6].

5.1 Framework

Let N' be an arbitrary nonempty set of players and $\mathcal{B} \subseteq 2^{N'}$ an arbitrary Boolean algebra of subsets (called "coalitions" in this section) of N'. A $\mathcal{B}$-

simple game ω is a subcollection of $\mathcal{B}$ such that $\emptyset \notin \omega$. The elements of ω are said to be winning, and the other elements in $\mathcal{B}$ are losing, as before. Our "simple game" is a $\mathcal{B}$-simple game with $N = \mathbf{N}$ and $\mathcal{B} = \mathrm{REC}$, if it does not contain $\emptyset$. Nakamura's "simple game" [34] is one with $\mathcal{B} = 2^{N'}$. The properties (such as monotonicity and weakness, defined in Section 2.1) for simple games are redefined for $\mathcal{B}$-simple games in an obvious way.

Let X be a (finite or infinite) set of **alternatives**, with cardinal number $\#X \geq 2$. Let $\mathcal{A}$ be the set of **(strict) preferences**, i.e., *acyclic* (for any finite set $\{x_1, x_2, \ldots, x_m\} \subseteq X$, if $x_1 \succ x_2, \ldots, x_{m-1} \succ x_m$, then $x_m \not\succ x_1$; in particular, $\succ$ is asymmetric and irreflexive) binary relations $\succ$ on X. (If $\succ$ is acyclic, we can show that the relation $\succeq$, defined by $x \succeq y \Leftrightarrow y \not\succ x$, is complete, i.e., reflexive and total.) A **($\mathcal{B}$-measurable) profile** is a list $\mathbf{p} = (\succ_i^{\mathbf{p}})_{i \in N'} \in \mathcal{A}^{N'}$ of individual preferences $\succ_i^{\mathbf{p}}$ such that $\{\, i \in N' : x \succ_i^{\mathbf{p}} y \,\} \in \mathcal{B}$ for all $x, y \in X$. Denote by $\mathcal{A}_{\mathcal{B}}^{N'}$ the set of all profiles.

A **$\mathcal{B}$-simple game with (ordinal) preferences** is a list $(\omega, X, \mathbf{p})$ of a $\mathcal{B}$-simple game $\omega \subseteq \mathcal{B}$, a set X of alternatives, and a profile $\mathbf{p} = (\succ_i^{\mathbf{p}})_{i \in N'} \in \mathcal{A}_{\mathcal{B}}^{N'}$. Given the $\mathcal{B}$-simple game with preferences, we define the dominance relation $\succ_\omega^{\mathbf{p}}$ by $x \succ_\omega^{\mathbf{p}} y$ if and only if there is a winning coalition $S \in \omega$ such that $x \succ_i^{\mathbf{p}} y$ for all $i \in S$. (In this definition, $\{\, i \in N' : x \succ_i^{\mathbf{p}} y \,\}$ need not be winning since we do not assume ω is monotonic. Andjiga and Mbih [3] study Nakamura's theorem, adopting the notion of dominance that requires the above coalition to be winning.) The **core** $C(\omega, X, \mathbf{p})$ of the $\mathcal{B}$-simple game with preferences is the set of undominated alternatives:

$$C(\omega, X, \mathbf{p}) = \{x \in X : \nexists y \in X \text{ such that } y \succ_\omega^{\mathbf{p}} x\}.$$

A **(preference) aggregation rule** is a map $\succ : \mathbf{p} \mapsto \succ^{\mathbf{p}}$ from profiles $\mathbf{p}$ of preferences to binary relations (social preferences) $\succ^{\mathbf{p}}$ on the set of alternatives. For example, the mapping $\succ_\omega$ from profiles $\mathbf{p} \in \mathcal{A}_{\mathcal{B}}^{N'}$ of acyclic preferences to dominance relations $\succ_\omega^{\mathbf{p}}$ is an aggregation rule. We typically restrict individual and social preferences to those binary relations $\succ$ on X that are asymmetric (i.e., complete $\succeq$) and either (i) acyclic or (ii) transitive (i.e., quasi-transitive $\succeq$) or (iii) negatively transitive (i.e., transitive $\succeq$). An aggregation rule is often referred to as a *social welfare function* when individual preferences and social preferences are restricted to the asymmetric, negatively transitive relations.

5.2 Nakamura's theorem and its consequences

Nakamura [34] gives a necessary condition for a $2^{N'}$-simple game with preferences to have a nonempty core for any profile $\mathbf{p}$, which is also sufficient if the set X of alternatives is finite. To state Nakamura's theorem, we define the **Nakamura number** $\nu(\omega)$ of a $\mathcal{B}$-simple game ω to be the size of the smallest collection of winning coalitions having empty intersection

$$\nu(\omega) = \min\{\#\omega' : \omega' \subseteq \omega \text{ and } \bigcap \omega' = \emptyset\}$$

if $\bigcap \omega = \emptyset$ (i.e., ω is nonweak); otherwise, set $\nu(\omega) = \#(2^X) > \#X$. Note that the Nakamura number is independent of X and $\mathbf{p}$.

The following useful lemma [34, Lemma 2.1] states that the Nakamura number of a $\mathcal{B}$-simple game cannot exceed the size of a winning coalition by more than one.

Lemma 13 *Let ω be a nonweak $\mathcal{B}$-simple game. Then $\nu(\omega) \leq \min\{\#S : S \in \omega\} + 1$.*

Proof. Choose a coalition $S \in \omega$ such that $\#S = \min\{\#S : S \in \omega\}$. Since $\bigcap \omega = \emptyset$, for each $i \in S$, there is some $S^i \in \omega$ with $i \notin S^i$. So, $S \cap (\bigcap_{i \in S} S^i) = \emptyset$. Therefore, $\nu(\omega) \leq \#S + 1$. ∎

It is easy to prove [34, Corollary 2.2] that the Nakamura number of a nonweak $\mathcal{B}$-simple game is at most equal to the cardinal number $\#N$ of the set of players and that this maximum is attainable if $\mathcal{B}$ contains all finite coalitions. In fact, one can easily construct a computable, nonweak simple game with any given Nakamura number:

Proposition 14 *For any integer $k \geq 2$, there exists a δ-computable, nonweak simple game ω with Nakamura number $\nu(\omega) = k$.*

Proof. Given an integer $k \geq 2$, let $S = \{0, 1, \ldots, k-1\}$ be a carrier and define $T \in \omega$ iff $\#(S \cap T) \geq k - 1$. Then $\nu(\omega) = k$. ∎

Since computable, nonweak simple games have winning coalitions, it has *finite* winning coalitions by Proposition 9. An immediate corollary of Lemma 13 is the following:

Corollary 15 *Let ω be a δ-computable, nonweak simple game. Then its Nakamura number $\nu(\omega)$ is finite.*

Nakamura [34] proves the following theorem for $\mathcal{B} = 2^{N'}$:

Theorem 16 *Let $\mathcal{B}$ be a Boolean algebra of sets of N'. Suppose that $\emptyset \notin \omega$ and $\omega \neq \emptyset$. Then the core $C(\omega, X, \mathbf{p})$ of a $\mathcal{B}$-simple game $(\omega, X, \mathbf{p})$ with preferences is nonempty for all (measurable) profiles $\mathbf{p} \in \mathcal{A}_{\mathcal{B}}^{N'}$ if and only if X is finite and $\#X < \nu(\omega)$.*

Remark 3. At first glance, Nakamura's proof [34, Theorem 2.3] of the necessary condition $\#X < \nu(\omega)$, does not appear to generalize to an arbitrary Boolean algebra $\mathcal{B}$: he constructs certain coalitions from winning coalitions by taking possibly *infinite unions and intersections*, as well as complements; a difficulty is that the resulting set of players may not belong to the Boolean algebra $\mathcal{B}$. However, it turns out that once we make use of the other necessary condition (disregarded by Nakamura) that X is finite, we only need to consider *finite* unions and intersections, and his proof actually works. Though accessible proofs are readily available in the literature (e.g., [8, Theorem 3.2]) for $\mathcal{B} = 2^{N'}$ and finite sets N' of players, we choose to give a proof here since most available proofs pay little attention to the *measurability* condition ($\mathbf{p} \in \mathcal{A}_{\mathcal{B}}^{N'}$) for the profiles $\mathbf{p}$ that they construct. ‖

Proof. ($\Longleftarrow$). Suppose that X is finite, $\#X < \nu(\omega)$, and $C(\omega, X, \mathbf{p}) = \emptyset$ for some measurable profile $\mathbf{p} \in \mathcal{A}_{\mathcal{B}}^{N'}$. Then follow the proof of Theorem 2.5 in Nakamura [34] to find a cycle with respect to $\succ_{\omega}^{\mathbf{p}}$ consisting of at most $\#X$ alternatives.

($\Longrightarrow$). Suppose $C(\omega, X, \mathbf{p}) \neq \emptyset$ for all $\mathbf{p} \in \mathcal{A}_{\mathcal{B}}^{N'}$.

(i) To show that X is finite, suppose it is infinite. Then X contains a countable subset $X' = \{x_1, x_2, x_3, \ldots\} \subseteq X$. Let $\mathbf{p} \in \mathcal{A}^{N'}$ be a profile such that all players $i \in N'$ have an identical preference $\succ_i^{\mathbf{p}}$ (e.g., the transitive closure of itself) satisfying $x_{j+1} \succ_i^{\mathbf{p}} x_j$ for all $j \in \{1, 2, \ldots\}$ and $x_1 \succ_i^{\mathbf{p}} y$ for all $y \in X \setminus X'$. The measurability condition $\mathbf{p} \in \mathcal{A}_{\mathcal{B}}^{N'}$ is satisfied since for all x, $y \in X$, we have $\{ i \in N' : x \succ_i^{\mathbf{p}} y \} = N'$ or $\emptyset$, both in $\mathcal{B}$. Choose any winning coalition $S \in \omega$, which exists by assumption. Then all players in S have the same preference $\succ_i^{\mathbf{p}}$, implying $x_{j+1} \succ_{\omega}^{\mathbf{p}} x_j$ for all j and $x_1 \succ_{\omega}^{\mathbf{p}} y$ for all $y \in X \setminus X'$. It follows that $C(\omega, X, \mathbf{p}) = \emptyset$; a contradiction.

(ii) To show that $\#X < \nu(\omega)$, suppose $r := \#X \geq \nu(\omega)$. This excludes the possibility that ω is weak or $\nu(\omega)$ is infinite. We will construct a profile $\mathbf{p}$ such that the dominance relation $\succ_{\omega}^{\mathbf{p}}$ has a cycle. By the definition of the Nakamura number, there is a collection $\omega' = \{L_1, \ldots, L_r\} \subseteq \omega$ such that $\bigcap \omega' = \bigcap_{k=1}^{r} L_k = \emptyset$. Define $L_0 = N'$ and for all $k \in \{1, \ldots, r\}$,

$$D_k = (L_0 \cap L_1 \cap \cdots \cap L_{k-1}) \setminus L_k.$$

Then $\{D_1, \ldots, D_r\}$ is a family of (possibly empty) pairwise disjoint coalitions in $\mathcal{B}$ such that $L_k \subseteq D_k^c := N' \setminus D_k$ for all k and $\bigcup_{k=1}^{r} D_k = N'$ ($i \in N'$ is in the first D_k such that $i \notin L_k$).

Write $X = \{x_1, \ldots, x_r\}$ and $x_0 = x_r$. Fix the cycle

$$\succ = \{(x_k, x_{k-1}) : k \in \{1, \ldots, r\}\}.$$

Define $\mathbf{p} \in \mathcal{A}^{N'}$ as follows: for each k, all players i in D_k have the same (acyclic) preference $\succ_i^{\mathbf{p}} = \succ \setminus \{(x_k, x_{k-1})\}$. Then for all $(x, y) \notin \succ$, we have $\{ i \in N' : x \succ_i^{\mathbf{p}} y \} = \emptyset \in \mathcal{B}$. On the other hand, for all $(x, y) = (x_k, x_{k-1}) \in \succ$, we have $\{ i \in N' : x \succ_i^{\mathbf{p}} y \} = D_k^c \in \mathcal{B}$ and $L_k \subseteq D_k^c$. Therefore, $\mathbf{p} \in \mathcal{A}_{\mathcal{B}}^{N'}$ and $\succ_{\omega}^{\mathbf{p}} = \succ$, a cycle. It follows that $C(\omega, X, \mathbf{p}) = \emptyset$; a contradiction. ∎

It follows from Theorem 16 that if a $\mathcal{B}$-simple game ω is *weak* (and satisfies $\emptyset \notin \omega$ and $\omega \neq \emptyset$), then the core $C(\omega, X, \mathbf{p})$ is nonempty for all profiles $\mathbf{p} \in \mathcal{A}_{\mathcal{B}}^{N'}$ if and only if X is finite. The more interesting case is where

ω is nonweak. Combined with Corollary 15, Theorem 16 has a consequence for nonweak, computable simple games:

Corollary 17 *Let ω be a δ-computable, nonweak simple game satisfying $\emptyset \notin \omega$ and $\omega \neq \emptyset$. Then there exists a finite number ν (the Nakamura number $\nu(\omega)$) such that the core $C(\omega, X, \mathbf{p})$ is nonempty for all profiles $\mathbf{p} \in \mathcal{A}^N_{\mathrm{REC}}$ if and only if $\#X < \nu$.*

If we drop the computability condition, the above conclusion no longer holds. An example of ω that has no such restriction on the size of the set X of alternatives is a nonweak prefilter (e.g., the q-complement rule of Example 1, for $q > 1$), which has an infinite Nakamura number.

In fact, we can say more, if we shift our attention from the core—the set of undominated alternatives with respect to the dominance relation $\succ^{\mathbf{p}}_{\omega}$—to the dominance relation itself.

Proposition 18 *Let ω be a nonweak simple game satisfying $\emptyset \notin \omega$ and $\omega \neq \emptyset$.* (i) *ω cannot be a δ-computable prefilter.* (ii) *If ω is δ-computable; then $\nu(\omega)$ is finite, and $\succ^{\mathbf{p}}_{\omega}$ is acyclic for all $\mathbf{p} \in \mathcal{A}^N_{\mathrm{REC}}$ if and only if $\#X < \nu(\omega)$.* (iii) *If ω is a prefilter, then $\succ^{\mathbf{p}}_{\omega}$ is acyclic for all $\mathbf{p} \in \mathcal{A}^N_{\mathrm{REC}}$, regardless of the cardinal number $\#X$ of X.*

Proof. (i) If ω is a nonweak prefilter, then it has an infinite Nakamura number. But nonweak computable games have finite Nakamura number by Corollary 15.

(ii) and (iii) are obvious from the following corollary [34, Theorem 3.1] of Theorem 16: $\succ^{\mathbf{p}}_{\omega}$ is acyclic for all $\mathbf{p} \in \mathcal{A}^{N'}_{\mathcal{B}}$ if and only if $\#X' < \nu(\omega)$ for all finite $X' \subseteq X$. (This corollary can also be obtained from the well-known fact that $\succ^{\mathbf{p}}_{\omega}$ is acyclic if and only if the set $C(\omega, X', \mathbf{p})$ of maximal elements with respect to $\succ^{\mathbf{p}}_{\omega}$ is nonempty for all finite subsets X' of X.) ∎

We can strengthen the acyclicity of the dominance relation $\succ^{\mathbf{p}}_{\omega}$ in statement (iii) of Proposition 18 by replacing the statement with one of the following: (iv) if ω is a *filter*, then $\succ^{\mathbf{p}}_{\omega}$ is transitive for all $\mathbf{p}$ such that all individuals have transitive preferences $\succ^{\mathbf{p}}_{i}$; (v) if ω is an *ultrafilter*, then $\succ^{\mathbf{p}}_{\omega}$ is asymmetric and negatively transitive for all $\mathbf{p}$ such that all individuals have asymmetric, negatively transitive preferences $\succ^{\mathbf{p}}_{i}$. In fact, statements (iii), (iv), and (v) each gives an aggregation rule $\succ_{\omega} : \mathbf{p} \mapsto \succ^{\mathbf{p}}$ that satisfies Arrow's conditions of "Unanimity" and "Independence of irrelevant alternatives." These results are immediate from the relevant definitions (Armstrong [4, Proposition 3.2] gives a proof). According to Arrow's Theorem [6], however, if the set N of players were replaced by a *finite* set, then social welfare functions given by statement (v) would be dictatorial (and ω would be weak).

In an attempt to escape from Arrow's impossibility, many authors have investigated the consequences of relaxing the rationality requirement (negative transitivity of $\succ_{\omega}^{\mathbf{p}}$) for social preferences. In view of the close connection between the rationality properties of an aggregation rule and preflters [8, Theorems 2.6 and 2.7] (also [23, 4, 5]), Proposition 18 has a significant implication for this investigation.

6 Examples

Propositions 7, 9, and 11 show that the class of computable games (i) includes the class of games that have finite carriers and (ii) is included in the class of games that have both finite winning coalitions and cofinite losing coalitions. In this section, we construct examples showing that these inclusions are strict.

We can find such examples without sacrificing the desirable properties of simple games. We pursue this task thoroughly in a companion paper [26]. The *noncomputable* simple game example in Section 6.1 that has both finite winning coalitions and cofinite losing coalitions is a sample of that work. It is monotonic, proper, strong, and nonweak. An example of a *computable* simple game that is monotonic, proper, strong, nonweak, and has no finite carrier is given in that work [26].

6.1 A noncomputable game with finite winning coalitions

We exhibit here a noncomputable simple game that is monotonic, proper, strong, nonweak, and have both finite winning coalitions and cofinite losing coalitions. It shows in particular that the class of computable games is strictly smaller than the class of games that have both finite winning coalitions and cofinite losing coalitions. In this respect, the game is unlike nonweak prefilters (such as the q-complement rules in Example 1); those examples do not have any finite winning coalitions. Furthermore, unlike nonprincipal ultrafilters—which are also monotonic, proper, strong, and nonweak noncomputable simple games—the game is nonweak in a stronger sense: it violates the finite intersection property.

Let $A = N \setminus \{0\} = \{1, 2, 3, \ldots\}$. We define the simple game ω as follows: Any coalition except $A^c = \{0\}$ extending the string 1 of length 1 (i.e., any coalition containing 0) is winning; any coalition except A extending the string 0 is losing; A is winning and A^c is losing. In other words, for all $S \in \text{REC}$,

$$S \in \omega \iff [S = A \text{ or } (0 \in S \;\&\; S \neq A^c)].$$

Remark 4. The reader familiar with the notion of repeated games (or binary rooted trees) may find the following visualization helpful. Think of the extensive form of an infinitely repeated game played by you, with the stage game consisting of two moves 0 and 1. If you choose 1 in the first stage, you will win unless you keep choosing 0 indefinitely thereafter; if you choose 0 in the first stage, you will lose unless you keep choosing 1 indefinitely thereafter. Now, you "represent" a certain coalition and play 1 in stage i if i is in the coalition; you play 0 in that stage otherwise. Then the coalition that you represent is winning if you win; it is losing if you lose. $\|$

Lemma 19 *ω is not δ-computable.*

The following proof demonstrates the power of Theorem 5, although its full force is not used (Proposition 4 suffices). Proposition 8, which appeared earlier in Mihara [33], does not have this power.

Proof. If ω is δ-computable, then by Theorem 5 (or by Proposition 4), A has an initial segment $A \cap k$ that is a winning determining string. But $A \cap k$ itself is not winning (though it extends the string trivially). ∎

Lemma 20 *ω has both finite winning coalitions and cofinite losing coalitions.*

Proof. For instance, $\{0,1\}$ is finite and winning. $N \setminus \{0,1\} = \{2,3,4,\dots\}$ is cofinite and losing. ∎

Lemma 21 *ω is monotonic.*

Proof. Suppose $S \in \omega$ and $S \subsetneq T$. There are two possibilities. If $S = A$, then $T = N$, and we have $N \in \omega$ by the definition of ω. Otherwise, S contains 0 and some other number i. The same is true of T, implying that $T \in \omega$. ∎

Lemma 22 *ω is proper and strong.*

Proof. It suffices to show that $S^c \in \omega \iff S \notin \omega$. From the definition of ω, we have

$$\begin{aligned} S \notin \omega &\iff S \neq A \;\&\; (0 \notin S \text{ or } S = A^c) \\ &\iff S^c \neq A^c \;\&\; (0 \in S^c \text{ or } S^c = A) \\ &\iff (0 \in S^c \;\&\; S^c \neq A^c) \text{ or } S^c = A \\ &\iff S^c \in \omega. \end{aligned}$$

∎

Lemma 23 *ω is not a prefilter. In particular, it is not weak.*

Proof. We show that the intersection of some finite family of winning coalitions is empty. The coalitions $\{0,1\}$, $\{0,2\}$, and A form such a family. (Incidentally, this shows that the Nakamura number of ω is three, since ω is proper.) ∎

6.2 A computable game without a finite carrier

We exhibit here a computable simple game that does not have a finite carrier. It shows that the class of computable games is strictly larger than the class of games that have finite carriers.

Our approach is to construct r.e. (in fact, recursive) sets T_0 and T_1 of determining strings (of 0's and 1's) satisfying the conditions of Theorem 5 (the full force of the theorem is not needed; the easier direction suffices). We first give a condition that any string in $T_0 \cup T_1$ must satisfy. We then specify each of T_0 and T_1, and construct the simple game by means of these sets. We conclude that the game is computable by checking (Lemmas 24, 25, and 27) that T_0 and T_1 satisfy the conditions of the theorem. Finally, we show (Lemma 28) that the game does not have a finite carrier.

Let $\{k_s\}_{s=0}^{\infty}$ be an effective listing (recursive enumeration) of the members of the r.e. set $\{k : \varphi_k(k) \in \{0,1\}\}$, where $\varphi_k(\cdot)$ is the kth p.r. function of one variable. We can assume that all elements k_s are distinct. (Such a listing $\{k_s\}_{s=0}^{\infty}$ exists by the Listing Theorem [41, Theorem II.1.8 and Exercise II.1.20].) Thus,

$$\mathrm{CRec} \subset \{k : \varphi_k(k) \in \{0,1\}\} = \{k_0, k_1, k_2, \ldots\},$$

where CRec is the set of characteristic indices for recursive sets.

Let $l_0 = k_0 + 1$, and for $s > 0$, let $l_s = \max\{l_{s-1}, k_s + 1\}$. We have $l_s \geq l_{s-1}$ (that is, $\{l_s\}$ is an nondecreasing sequence of numbers) and $l_s > k_s$ for each s. Note also that $l_s \geq l_{s-1} > k_{s-1}$, and $l_s \geq l_{s-2} > k_{s-2}$, etc. imply that $l_s > k_s, k_{s-1}, k_{s-2}, \ldots$.

For each s, let F_s be the set of strings $\alpha = \alpha(0)\alpha(1)\cdots\alpha(l_s - 1)$ (the *'s denoting the concatenation are omitted) of length l_s such that

$$\alpha(k_s) = \varphi_{k_s}(k_s) \text{ and for each } s' < s,\ \alpha(k_{s'}) = 1 - \varphi_{k_{s'}}(k_{s'}). \tag{4}$$

Note that (4) imposes no constraints on $\alpha(k_{s'})$ for $s' > s$ and no constraints on $\alpha(k)$ for $k \notin \{k_0, k_1, k_2, \ldots\}$, while it imposes real constraints for $s' \leq s$, since $|\alpha| = l_s > k_{s'}$ for such s'. We observe that if $\alpha \in F_s \cap F_{s'}$, then $s = s'$.

Let $F = \bigcup_s F_s$. (F will be the union of T_0 and T_1 defined below.) We claim that for any two distinct elements α and β in F we have neither $\alpha \subseteq \beta$

(α is an initial segment of β) nor $\beta \subseteq \alpha$ (i.e., there is $k < \min\{|\alpha|, |\beta|\}$ such that $\alpha(k) \neq \beta(k)$). To see this, let $|\alpha| \leq |\beta|$, without loss of generality. If α and β have the same length, then the conclusion follows since otherwise they become identical strings. If $l_s = |\alpha| < |\beta| = l_{s'}$, then $s < s'$ and by (4), $\alpha(k_s) = \varphi_{k_s}(k_s)$ on the one hand, but $\beta(k_s) = 1 - \varphi_{k_s}(k_s)$ on the other hand. So $\alpha(k_s) \neq \beta(k_s)$.

The game ω will be constructed from the sets T_0 and T_1 of strings defined as follows:

$$\begin{aligned} \alpha \in T_0 &\iff \exists s\, [\alpha \in F_s \text{ and } \alpha(k_s) = \varphi_{k_s}(k_s) = 0] \\ \alpha \in T_1 &\iff \exists s\, [\alpha \in F_s \text{ and } \alpha(k_s) = \varphi_{k_s}(k_s) = 1]. \end{aligned}$$

We observe that $T_0 \cup T_1 = F$ and $T_0 \cap T_1 = \emptyset$.

Define ω by $S \in \omega$ if and only if S has an initial segment in T_1. Lemmas 24, 25 and 27 establish computability of ω by way of Theorem 5.

Lemma 24 *T_0 and T_1 are recursive.*

Proof. We prove that T_0 is recursive; the proof for T_1 is similar. We give an algorithm that can decide for each given string whether it is in T_0 or not.

To decide whether a string σ is in T_0, generate k_0, k_1, k_2, ..., compute l_0, l_1, l_2, ..., and determine F_0, F_1, F_2, ... until we find the least s such that $l_s \geq |\sigma|$. If $l_s > |\sigma|$, then $\sigma \notin F_s$. Since l_s is nondecreasing in s and F_s consists of strings of length l_s, it follows that $\sigma \notin F$, implying $\sigma \notin T_0$.

If $l_s = |\sigma|$, then check whether $\sigma \in F_s$; this can be done since the values of $\varphi_{k_{s'}}(k_{s'})$ for $s' \leq s$ in (4) are available and F_s determined by time s. If $\sigma \notin F_s$ and $l_{s+1} > l_s$, then $\sigma \notin T_0$ as before. Otherwise check whether $\sigma \in F_{s+1}$. If $\sigma \notin F_{s+1}$ and $l_{s+2} > l_{s+1} = l_s$, then $\sigma \notin T_0$ as before. Repeating this process, we either get $\sigma \in F_{s'}$ for some s' or $\sigma \notin F_{s'}$ for all $s' \in \{s' : l_{s'} = l_s\}$. In the latter case, we have $\sigma \notin T_0$. In the former case, if $\sigma(k_{s'}) = \varphi_{k_{s'}}(k_{s'}) = 1$, then $\sigma \in T_1$ by the definition of T_1; hence it is not in T_0. Otherwise $\sigma(k_{s'}) = \varphi_{k_{s'}}(k_{s'}) = 0$, and we have $\sigma \in T_0$. ∎

Lemma 25 *T_1 consists only of winning determining strings for ω; T_0 consists only of losing determining strings for ω.*

Proof. Let $\alpha \in T_1$. If a coalition S extends α, then by the definition of ω, S is winning. This proves that α is a winning determining string.

Let $\alpha \in T_0$. Suppose a coalition S extends $\alpha \in T_0 \subset T_0 \cup T_1 = F$. If $\beta \in F$ and $\beta \neq \alpha$, we have, as shown before, $\alpha \not\subseteq \beta$ and $\beta \not\subseteq \alpha$, which implies that S does not extend β. So, in particular, S does not extend any string in T_1. It follows from the definition of ω that S is losing. This proves that α is a losing determining string. ∎

Lemma 26 *For each s, any string α of length l_s such that $\alpha(k_s) = \varphi_{k_s}(k_s)$ extends a string in $\bigcup_{t \leq s} F_t$.*

Proof. We proceed by induction on s. Let α be a string of length l_s such that $\alpha(k_s) = \varphi_{k_s}(k_s)$. If $s = 0$, we have $\alpha \in F_0$; hence the lemma holds for $s = 0$. Suppose the lemma holds for $s' < s$. If for some $s' < s$, $\alpha(k_{s'}) = \varphi_{k_{s'}}(k_{s'})$, then by the induction hypothesis, the $l_{s'}$-initial segment $\alpha \cap l_{s'}$ of α extends a string in $\bigcup_{t \leq s'} F_t$. So α extends a string in $\bigcup_{t \leq s} F_t$. Otherwise, we have for each $s' < s$, $\alpha(k_{s'}) = 1 - \varphi_{k_{s'}}(k_{s'})$. Then by (4), $\alpha \in F_s \subset \bigcup_{t \leq s} F_t$. ∎

Lemma 27 *Any coalition $S \in$ REC has an initial segment in T_0 or T_1.*

Proof. Suppose φ_k is the characteristic function for a recursive coalition S. Then $k \in \{k_0, k_1, k_2, \ldots\}$ since this set contains the set CRec of characteristic indices. So $k = k_s$ for some s. Consider the initial segment $S \cap l_s$. It extends a string in $\bigcup_{t \leq s} F_t$ by Lemma 26. The conclusion follows since $\bigcup_{t \leq s} F_t \subset F = T_0 \cup T_1$. ∎

Lemma 28 *ω does not have a finite carrier.*

Proof. We will construct a set A such that for infinitely many l, the l-initial segment $A \cap l$ has an extension (as a string) that is winning and for infinitely many l', $A \cap l'$ has an extension that is losing. This implies that $A \cap l$ is not a carrier of ω for any such l. So no subset of $A \cap l$ is a carrier. Since there are arbitrarily large such l, this proves that ω has no finite carrier.

Let A be a set such that for each k_t, $A(k_t) = 1 - \varphi_{k_t}(k_t)$. For any $s' > 0$ and $i \in \{0, 1\}$, there is an $s > s'$ such that $k_s > l_{s'}$ and $\varphi_{k_s}(k_s) = i$.

For a temporarily chosen s', fix i and fix such s. Then choose the greatest s' satisfying these conditions. Since $l_s > k_s > l_{s'}$, there is a string α of length l_s extending (as a string) $A \cap l_{s'}$ such that $\alpha \in F_s$. Since $\alpha(k_s) = \varphi_{k_s}(k_s) = i$, we have $\alpha \in T_i$.

There are infinitely many such s, so there are infinitely many such s'. It follows that for infinitely many $l_{s'}$, the initial segment $A \cap l_{s'}$ is a substring of some string α in T_1 (by Lemma 25, α is winning in this case), and for infinitely many $l_{s'}$, $A \cap l_{s'}$ is a substring of some (losing) string α in T_0. ∎

References

[1] L. Anderlini and L. Felli. Incomplete written contracts: Undescribable states of nature. *Quarterly Journal of Economics*, 109:1085–1124, 1994.

[2] L. Anderlini and H. Sabourian. Cooperation and effective computability. *Econometrica*, 63:1337–1369, 1995.

[3] N. G. Andjiga and B. Mbih. A note on the core of voting games. *Journal of Mathematical Economics*, 33:367–372, 2000.

[4] T. E. Armstrong. Arrow's Theorem with restricted coalition algebras. *Journal of Mathematical Economics*, 7:55–75, 1980.

[5] T. E. Armstrong. Precisely dictatorial social welfare functions: Erratum and addendum to 'Arrow's Theorem with restricted coalition algebras'. *Journal of Mathematical Economics*, 14:57–59, 1985.

[6] K. J. Arrow. *Social Choice and Individual Values*. Yale University Press, New Haven, 2nd edition, 1963.

[7] K. J. Arrow. Rationality of self and others in an economic system. *Journal of Business*, 59:S385–S399, 1986.

[8] D. Austen-Smith and J. S. Banks. *Positive Political Theory I: Collective Preference*. University of Michigan Press, Ann Arbor, 1999.

[9] J. S. Banks. Acyclic social choice from finite sets. *Social Choice and Welfare*, 12:293–310, 1995.

[10] J. S. Banks, J. Duggan, and M. L. Breton. Social choice and electoral competition in the general spatial model. *Journal of Economic Theory*, 126:194–234, 2006.

[11] J. Bartholdi III, C. A. Tovey, and M. A. Trick. Voting schemes for which it can be difficult to tell who won the election. *Social Choice and Welfare*, 6:157–165, 1989.

[12] J. J. Bartholdi III, C. A. Tovey, and M. A. Trick. The computational difficulty of manipulating an election. *Social Choice and Welfare*, 6:227–241, 1989.

[13] D. Canning. Rationality, computability, and Nash equilibrium. *Econometrica*, 60(4):877–888, July 1992.

[14] G. S. Ceĭtin. Algorithmic operators in constructive complete separable metric spaces. *Doklady Akad. Nauk*, 128:49–52, 1959. (in Russian).

[15] X. Deng and C. H. Papadimitriou. On the complexity of cooperative solution concepts,. *Mathematics of Operations Research*, 19:257–266, 1994.

[16] A. Downs. *An Economic Theory of Democracy*. Addison-Wesley, Boston, 1957.

[17] R. Evans and J. P. Thomas. Cooperation and punishment. *Econometrica*, 69:1061–1075, 2001.

[18] Q. Fang, S. Zhu, M. Cai, and X. Deng. On computational complexity of membership test in flow games and linear production games. *International Journal of Game Theory*, 31:39–45, 2002.

[19] M. Fey. May's theorem with an infinite population. *Social Choice and Welfare*, 23:275–293, 2004.

[20] I. Gilboa. Philosophical applications of Kolmogorov's complexity measure. Discussion Paper 923, Northwestern University, Center for Mathematical Studies in Economics and Management Science, Oct. 1990.

[21] A. Gomberg, C. Martinelli, and R. Torres. Anonymity in large societies. *Social Choice and Welfare*, 25:187–205, 2005.

[22] J. S. Kelly. Social choice and computational complexity. *Journal of Mathematical Economics*, 17:1–8, 1988.

[23] A. P. Kirman and D. Sondermann. Arrow's Theorem, many agents, and invisible dictators. *Journal of Economic Theory*, 5:267–277, 1972.

[24] S. Koppelberg. *Handbook of Boolean Algebras*, volume 1. North-Holland, Amsterdam, 1989. Edited by J. D. Monk, with the cooperation of R. Bonnet.

[25] G. Kreisel, D. Lacombe, and J. R. Shoenfield. Partial recursive functionals and effective operations. In A. Heyting, editor, *Constructivity in Mathematics*, Studies in Logic and the Foundations of Mathematics, pages 290–297, Amsterdam, 1959. North-Holland.

[26] M. Kumabe and H. R. Mihara. Computability of simple games: A complete investigation of the sixty-four possibilities. In preparation, 2006.

[27] A. A. Lewis. An infinite version of Arrow's Theorem in the effective setting. *Mathematical Social Sciences*, 16:41–48, 1988.

[28] B. L. Lipman. Information processing and bounded rationality: A survey. *Canadian Journal of Economics*, 28:42–67, 1995.

[29] H. R. Mihara. Anonymity and neutrality in Arrow's Theorem with restricted coalition algebras. *Social Choice and Welfare*, 14(4):503–12, Oct. 1997.

[30] H. R. Mihara. Arrow's Theorem and Turing computability. *Economic Theory*, 10(2):257–76, Aug. 1997.

[31] H. R. Mihara. Arrow's theorem, countably many agents, and more visible invisible dictators. *Journal of Mathematical Economics*, 32(3):267–287, 1999.

[32] H. R. Mihara. Existence of a coalitionally strategyproof social choice function: A constructive proof. *Social Choice and Welfare*, 18(3):543–553, 2001.

[33] H. R. Mihara. Nonanonymity and sensitivity of computable simple games. *Mathematical Social Sciences*, 48(3):329–341, 2004.

[34] K. Nakamura. The vetoers in a simple game with ordinal preferences. *International Journal of Game Theory*, 8:55–61, 1979.

[35] P. Odifreddi. *Classical Recursion Theory: The Theory of Functions and Sets of Natural Numbers*. Elsevier, Amsterdam, 1992.

[36] B. Peleg. Game-theoretic analysis of voting in committees. In K. J. Arrow, A. K. Sen, and K. Suzumura, editors, *Handbook of Social Choice and Welfare*, volume 1, chapter 8, pages 395–423. Elsevier, Amsterdam, 2002.

[37] K. Prasad. On the computability of Nash equilibria. *Journal of Economic Dynamics and Control*, 21:943–953, 1997.

[38] M. K. Richter and K.-C. Wong. Computable preference and utility. *Journal of Mathematical Economics*, 32:339–354, 1999.

[39] M. K. Richter and K.-C. Wong. Non-computability of competitive equilibrium. *Economic Theory*, 14:1–27, 1999.

[40] A. Rubinstein. *Modeling Bounded Rationality*. MIT Press, Cambridge, Massachusetts, 1998.

[41] R. I. Soare. *Recursively Enumerable Sets and Degrees: A Study of Computable Functions and Computably Generated Sets*. Springer-Verlag, Berlin, 1987.

[42] S. E. Spear. Learning rational expectations under computability constraints. *Econometrica*, 57:889–910, July 1989.

[43] M. Truchon. Voting games and acyclic collective choice rules. *Mathematical Social Sciences*, 29, 1995.

[44] R. J. Weber. Games in coalitional form. In R. J. Aumann and S. Hart, editors, *Handbook of Game Theory*, volume 2, chapter 36, pages 1285–1303. Elsevier, Amsterdam, 1994.

[45] M. Yokoo, V. Conitzer, T. Sandholm, N. Ohta., and A. Iwasaki. Coalitional games in open anonymous environments. In *Proceedings of the National Conference on Artificial Intelligence (AAAI)*, Pittsburgh, PA, 2005. American Association for Artificial Intelligence.