5-2008

# A Survey of Social Choice Failures: Majority and Borda Rules

Lauren Nicole Merrill
*College of William and Mary*

# A Survey of Social Choice Failures: Majority and Borda Rules

A thesis submitted in partial fulfillment of the requirement
for the degree of Bachelor of Science with Honors in
Mathematics from the College of William and Mary in Virginia,

by

Lauren Nicole Merrill.

Accepted for Highest Honors
(Honors, High Honors or Highest Honors)

Advisor: Professor Charles R. Johnson

Professor Donald E. Campbell

Professor Lawrence Leemis

Williamsburg, Virginia
April 25, 2008

**Abstract**

A social welfare rule $g$ selects a complete asymmetric binary relation on a set of alternatives $A$ as a function of voter preferences over $A$. Arrow's Impossibility Theorem and the Gibbard–Satterthwaite Theorem show that all social welfare rules fail to satisfy a small number of seeminly innocuous properties when voter preferences are unrestricted. In this paper, we propose several techniques for quantifying the degree of these failures for simple majority rule and Borda's rule. In addition, we develop a matricial framework for analyzing social welfare rules. We believe that the tools and methods proposed have significant potential in future analysis.

**Acknowledgments**

This work was performed in part using computational facilities at The College of William & Mary which were enabled by grants from Sun Microsystems, the National Science Foundation, and Virginia's Commonwealth Technology Research Fund. Special thanks is due to R. Chris Bording for technical and programming support.

Portions of this project were presented at a poster session entitled "Calculating the Probability of a Transitive Majority Rule Outcome" at the SIAM Conference on Parallel Processing for Scientific Computing held in Atlanta, Georgia March 12-14, 2008.

Funding for this research was provided in part by The College of William & Mary, through the Monroe Scholars Program administered by the Roy R. Charles Center.

# Contents

# Chapter 1

# Introduction

Social choice theory is often characterized as a field of impossibility theorems. Indeed, its inception was largely tied to Arrow's Theorem [2], which demonstrated the impossibility of constructing a voting mechanism that satisfies a number of seemingly innocuous properties. Since then numerous other impossibility theorems have been proved, and conditions have been identified under which possibility theorems may be obtained.

The axiomatic approach to possibility and impossibility theorems is largely a question of existence: does there exist a voting rule that is well-behaved for some large family of voting situations; or, do there exist voting situations in which all voting rules are poorly behaved? Although such inquiries have significantly advanced the theoretical underpinnings of the field, they often fail to provide insight for practical voting situations.

There are many real-world situations in which group decisions need to be made, and the results of social choice theory offer little practical advice for how such decisions should be made. Theory might tell us, for example, that a voting rule will be subject to manipulation; what we desire to know is the degree to which it can be manipulated. Theory might also tell us that a particular voting rule, though attractive in many ways, is undesirable in certain settings. What we attempt to quantify how often such settings occur, and how poorly the rules can perform.

This paper attempts to address this gap between theoretical (often existence) results and more practical measures of performance. In particular, we focus on two common voting rules: simple majority rule and Borda's rule. The first is known to fail to produce useful outcomes, while the second is known to be subject to strategic manipulation. We attempt to quantify in certain way both of these failures.

This paper contains few concrete results. As a relatively unexplored field, there was much groundwork to set in place and little foundation of existing results on which to build. To this end, we have developed a number of tools that we believe will be of extreme importance to the further development of this field. We have uncovered a number of difficult and important questions, and hope that the investment we have made in developing new tools will prove valuable to future researchers.

Chapter 1 begins with a qualitative introduction to the field of social choice theory. Section 1.2 introduces the standard social choice notation and definitions in addition to new notation and a number of alternative definitions developed specifically for this project. Section 1.3 introduces an algorithm for creating vector representations of standard social choice objects, and contains a cursory discussion of the benefits of such a representation.

In Chapter 2 we begin with an introduction to generalized majority rules and show that these rules display a number of desirable properties. Section 2.2 focuses on a particularly famous generalized majority rule: simple majority rule. In this section, we illustrate the failures of simple majority rule by means of an example. In Section 2.3 we present a linear algebraic proof that all logically possible voting outcomes can be obtained by applying majority rule to the appropriate group of voters. We also present some known results, and show that no finite number of voters is sufficient to obtain all logically possible voting outcomes above some threshold number of alternatives.

Chapter 3 introduces generalized Borda rules, and shows that these rules also display a number of desirable properties; we also show their most pronounced failures. Section 3.2 introduces the standard Borda rule (also known simply as Borda's rule). In Section 3.3 we discuss strategic manipulation of Borda's rule and introduce a formal measure of manipulability. In Section 3.3, exact manipulation measures are presented for small cases and a computer simulation is developed for estimating manipulability when there are more voters or alternatives.

In Chapter 4 we introduce a matricial approach to social choice theory. Sections 4.1 and 4.2 recast generalized majority rules and Borda rules in the matricial framework. Section 4.3 introduces Perron's rule, a social choice rule that is defined solely in terms of matrices.

Finally, the appendices contain simulation and computation output, as well as source code. A large amount of data was produced through this research, and we believe that future research may benefit from access to these data.

## 1.1 Social Choice Theory

There are many everyday situations in which a group of individuals must make a collective decision: an academic department electing a chair, a community deciding which public project to fund, a family choosing a restaurant at which to eat dinner. Similarly, there are situations in which a group must collectively order a set of alternatives, such as candidates for a job or college applicants. Social choice theory provides an axiomatic framework in which to analyze such decision processes.

The field of social choice theory is primarily concerned with the formal analysis of collective decision-making processes. The roots of the field date to 1785, when the French philosopher and mathematician Marie Jean Antoine Nicolas de Caritat, marquis de Condorcet published his *Essays on the Application of Analysis to the Probability of Majority Decisions* [3]. Among other topics the essay introduced the Condorcet method, a formulation of majority rule, and illustrated Condorcet's Voting Paradox, which states that group preferences may be intransitive even when individual preferences are not.

The field was founded in its modern form by the work of Kenneth Arrow in 1951, and was propelled forward by early landmark results, including Arrow's Impossibility Theorem [2] and its refinements. Arrow's theorem sets forth a number of natural properties we might desire a decision-making rule to possess, then shows that the properties are incompatible. We give a qualitative version of Arrow's theorem here; a formal version follows in Section 1.2.

**Theorem 1.1.1** (Arrow's Impossibility Theorem)**.** *Suppose there are at least two voters and at least three alternatives. Then no deterministic voting rule that produces a linear ordering of the alternatives as a function of voter preferences can simultaneously exhibit all of the following properties:*

1. *The rule is defined for all specifications of voter preferences.*
2. *There is no individual for whom the societal ranking always corresponds with his or her individual ranking.*
3. *Regardless of voter preferences, the introduction (or removal) of a single alternative does not change the relative ranking of the existing (or remaining) alternatives.*
4. *If a voter promotes an alterative in his or her individual ranking, the alternative can never be demoted in the societal ranking.*
5. *Every possible ranking of the alternatives is attainable as the societal ranking of some set of voter preferences.*

Another famous result was developed in independently by Gibbard [6] and Satterthwaite [17]. The result is closely related to Arrow's theorem; Reny [15] provides a single proof that yields both results. Again, we present the theorem informally.

**Theorem 1.1.2** (Gibbard–Satterthwaite Theorem)**.** *Suppose there are at least two voters and at least three alternatives, and that voters* *are allowed to have any preferences over the set of alternatives. Then one of the following must be true for any deterministic voting rule that selects one alternative as a function of the voter preferences:*

1. *One of the voters is a dictator, so the rule always chooses from that voter's most-preferred set of alternatives.*

2. *There is some alternative that is never selected by the rule.*

3. *There exist situations in which an individual voter can benefit from misrepresenting his or her preferences over the set of alternatives.*

Social choice theory therefore provides a framework in which collective decision-making processes can be analyzed axiomatically. A social choice (similarly, welfare) rule is a mechanism for choosing an alternative (ranking of alternatives) as a function of individual preferences. Within the field, there are multitudes of identified properties a given rule might display, much analysis has been devoted to identifying situations in which there exist rules that possess "good" properties.

The focus of this paper is to investigate and partially quantify the occurrence of "bad" properties in rules that are otherwise appealing. In the next section, we establish the notation and definitions necessary for this analysis.

## 1.2 Notation and Definitions

Let $V$ be the set of voters indexed by the natural numbers, $V = \{1, \ldots, m\}$. Let $A = \{a_1, \ldots, a_n\}$ be the set of alternatives. Unless explicitly stated, we impose no assumptions on $V$ and $A$ except that each is nonempty, and $m, n < \infty$.

Let $L_W(A)$ denote the set of weak linear orders on the set $A$ and let $L_S(A)$ denote the set of strict linear orders on $A$. Each voter has a *preference order* over the alternatives in $A$ that is a member of $L_W(A)$; we will occasionally further restrict voter preferences to

$L_S(A)$. Clearly, $L_S(A) \subset L_W(A)$. These axiomatic assumptions coincide with out intuitive understanding of individual preferences:

- Complete: For all alternative pairs $(a_i, a_j)$, individuals have some preference relation (potentially indifference) between the pair.
- Transitive: For all alternative triples $(a_i, a_j, a_k)$, no individual would prefer simultaneously $a_i$ to $a_j$, $a_j$ to $a_k$, and $a_k$ to $a_i$.
- Asymmetric Part: There exist alternative pairs $(a_i, a_j)$ for which individuals strictly prefer alternative $a_i$ to $a_j$.
- Symmetric Part: There exist alternative pairs $(a_i, a_j)$ for which individuals are indifferent between the alternatives $a_i$ and $a_j$.

A *profile* $p$ is a mapping that assigns a preference ordering to each voter. We let $p(i)$ indicate voter $i$'s preference ordering at profile $p$. When voter preference orderings are restricted to $L_S(A)$, let $p_k(i)$ indicate voter $i$'s $k$th most-preferred alternative at profile $p$. When voter preferences are elements of $L_W(A)$, let $p_k(i)$ indicate the $k$th ranked set of alternatives (where voter $i$ is indifferent between all alternatives within the set). Since there are a finite number of alternatives, $p_1(i)$ is nonempty for all profiles $p$ and voters $i$; for $k = 2, \ldots, n$ it may be that $p_k(i) = \emptyset$.

**Example 1.2.1.** *Let $A = \{a_1, a_2, a_3\}$ be the set of alternatives, $p$ be a profile, and $i$ be a voter such that*

$$p(i) = (a_1 \sim a_2 \succ a_3).$$

*Then*

$$p_1(i) = \{a_1, a_2\}, \qquad p_2(i) = \{a_3\}, \qquad \textit{and} \qquad p_3(i) = \emptyset.$$

The notation $a_i \succ_{p(k)} a_j$ is used to indicate that individual $k$ prefers alternative $a_i$ to alternative $a_j$ at profile $p$. Similarly, $a_i \sim_{p(k)} a_j$ indicates that voter $k$ is indifferent over the pair $(a_i, a_j)$ at profile $p$.

**Example 1.2.2.** *Suppose that $A = \{a_1, a_2, a_3\}$, and that at profiles $p$ and $q$ (respectively) voter $k$ has the preference orderings*

$$p(k) = (a_1 \succ a_2 \succ a_3), \qquad q(k) = (a_1 \sim a_2 \succ a_3).$$

*These preference orderings indicate that at $p$ voter $k$ prefers alternative $a_1$ to $a_2$ and $a_3$, and alternative $a_2$ to $a_3$. At profile $q$ voter $k$ prefers alternative $a_1$ and $a_2$ to $a_3$, but is indifferent between $a_1$ and $a_2$.*

*Therefore, we have that $p \in L_S(A)^m \subset L_W(A)^m$ and $q \in L_W(A)^m \setminus L_S(A)^m$, where $m$ is the total number of voters.*

We can imagine a profile in the form of a 2–dimensional array, in which each column represents a particular voter's preference ordering. When voter preferences are in $L_S(A)$, the array has dimension $n \times m$; for convention we will retain these dimensions even when voter preferences contain indifference, and could be visualized in a smaller array. For clarity, we will generally append a row to the top of the profile matrix to indicate the voter to whom each column is assigned. The following example illustrates the use of this notation.

**Example 1.2.3.** *Let $V = \{1, 2, 3\}$ and $A = \{a_1, a_2, a_3\}$. Then the profile*

$$p = \begin{bmatrix} \underline{1} & \underline{2} & \underline{3} \\ a_1 & a_2 & a_1 \\ a_2 & a_1 & a_3 \\ a_3 & a_3 & a_2 \end{bmatrix}$$

*indicates that $p(1) = (a_1 \succ a_2 \succ a_3)$ and $p(3) = (a_1 \succ a_3 \succ a_2)$. Furthermore, we have $p_1(2) = a_2$ and $p_3(3) = a_2$. Since all preferences are strict, we have $p \in L_S(A)^3$.*

*Next consider the profile*

$$q = \begin{bmatrix} \underline{1} & \underline{2} & \underline{3} \\ a_1 \sim a_2 & a_2 & a_1 \\ a_3 & a_1 \sim a_3 & a_3 \\ & & a_2 \end{bmatrix},$$

*where* $q(1) = (a_1 \sim a_2 \succ a_3)$*,* $q(2) = (a_2 \succ a_2 \sim a_3)$*, and* $q(3) = (a_1 \succ a_3 \succ a_2)$*. Since voters 1 and 2 are indifferent over some pairs of alternatives, we have* $q \in L_W(A)^3 \setminus L_S(A)^3$*.*

We use the notation $V_p(a_i \succ a_j)$ to indicate the subset of voters preferring alternative $a_i$ to $a_j$ at profile $p$. Similarly, we use $V_p(a_i \sim a_j)$ to indicate the subset of voters preferring alternative $a_i$ to $a_j$ at profile $p$. In the above example, we had $V_p(a_1 \succ a_3) = \{1, 3\}$ and $V_p(a_1 \sim a_2) = \{1\}$. For all $a_i, a_j \in A$ and all profiles $p \in L_W(A)^m$, the sets $V_p(a_i \succ a_j)$, $V_p(a_i \sim a_j)$, $V_p(a_j \succ a_i)$ form a partition of the set $V$.

A *social welfare rule* is a function $g : \wp \longrightarrow R(A)$, where $\wp$ denotes some profile space and $R(A)$ is the set of all complete anti-symmetric relations on $A$, with asymmetric part $\succ$ and symmetric part $\sim$. For our purposes, we will assume that either $\wp = L_W(A)^m$ or $\wp = L_S(A)^m$. We use the notation $g_k(p)$ to indicate the set of alternatives ranking $k$th in the *societal preference outcome* $g(p)$. As before, $g_1(p)$ is always nonempty although $g_k(p)$ may be empty for $k = 2, \ldots, n$.

**Example 1.2.4.** *Let* $A = \{a_1, a_2, a_3\}$*,* $p, q \in L_W(A)^m$*, and* $g$ *be a social welfare rule with*

$$g(p) = (a_1 \sim a_2 \sim a_3), \qquad g(q) = (a_1 \succ a_2 \sim a_3).$$

*Then* $g_1(p) = A$ *and* $g_2(p) = \emptyset$*. Also* $g_1(q) = \{a_1\}$*,* $g_2(q) = \{a_2, a_3\}$*, and* $g_3(q) = \emptyset$*.*

There are several properties of interest that an arbitrary social welfare rule may possess. We next define a number of these properties.

**Definition 1.2.5** (Dictatorship)**.** *A social welfare rule* $g$ *is dictatorial if and only if* $\exists i \in V$ *such that* $\forall p \in \wp$*,* $g(p) = p(i)$*. We call such an individual* $i$ *the dictator. A rule is non-dictatorial if and only if it is not dictatorial.*

**Definition 1.2.6** (Manipulability)**.** *A social welfare rule* $g$ *is manipulable if and only if* $\exists\ p, q \in \wp$ *and* $i \in V$ *such that* $p(j) = q(j)$

*for all* $j \neq i$ *and* $g_1(p) \neq g_1(q)$*. A social choice rule is strategy-proof if and only if it is not manipulable.*

If a social welfare rule $g$ is manipulable as above, we say that individual $i$ can manipulate at profile $q$ via the preference ordering $p(i)$ (or, symmetrically, can manipulate at profile $p$ via the preference ordering $q(i)$). We note that this definition of manipulation differs from the definition traditionally used in the literature.

**Definition 1.2.7** (Unanimity)**.** *A social welfare rule* $g$ *is unanimous if and only if whenever* $\exists p \in \wp$ *and* $a_i \in A$ *such that* $a_i \in p_1(j)$ *for all* $j \in V$*, then* $a_i \in g_1(p)$*.*

In particular, when all voters share a single most-preferred alternative a unanimous social welfare rule must rank that alternative as the single most-preferred in the societal preference outcome.

**Definition 1.2.8** (Anonymity)**.** *For a profile* $p \in \wp$ *and any permutation* $\sigma_V$ *of the voters in* $V$*, define* $\sigma_V(p)$ *to be the induced permutation on* $p$*. Then a social welfare rule* $g$ *is anonymous if and only if* $g(p) = g(\sigma_V(p))$*.*

We will next show that an anonymity and dictatorship are incompatible.

**Lemma 1.2.9.** *If* $g$ *be an anonymous social welfare rule, then* $g$ *is non-dictatorial.*

*Proof.* Let $p$ be a profile at which each voter's preference ordering is distinct; that is $p(i) \neq p(j)$ for all $i, j \in V$ $(i \neq j)$. Suppose that $g$ is an anonymous and dictatorial social welfare rule, with voter $k$ the dictator. Let $\sigma_V$ be any permutation on the set of voters such that $\sigma_V(k) \neq k$.

Since $g$ is anonymous, $g(p) = g(\sigma_V(p))$. Moreover, since $g$ has dictator $k$, $g(p) = p(k)$ and $g(\sigma_V(p)) = p(\sigma_V(k))$. However, $p(k) \neq p(\sigma_V(k))$, by construction.

Since this is a contradiction, there does not exist a social welfare rule $g$ that is both anonymous and dictatorial. □

We next introduce a dual property of anonymity, for which social welfare rules are invariant in the natural way to permutations of the alternatives.

**Definition 1.2.10** (Neutrality)**.** *Let* $\sigma_A$ *be a permutation of the alternatives in* $A$ *and* $\sigma_A(p)$ *be the induced permutation on* $p$*. Then a social welfare* $g$ *is neutral if and only if* $\sigma_A \circ g(p) = g(\sigma_A(p))$*.*

Admittedly the $\sigma$ notation in the preceding two definitions is somewhat abusive. Nevertheless the concepts are sufficiently clear that the abuse is preferable to the introduction of additional notation. For clarity, the following example illustrates permutations of the voters and of the alternatives.

**Example 1.2.11.** *Let* $V = \{1, 2, 3\}$ *and* $A = \{a_1, a_2, a_3\}$*. Let* $\sigma_V = (1, 3, 2)$ *be a permutation on the set of voters that relabels voter 1 as 3, 3 as 2, and 2 as 1. Then for*

$$p = \begin{bmatrix} \underline{1} & \underline{2} & \underline{3} \\ a_1 & a_2 & a_1 \\ a_2 & a_1 & a_3 \\ a_3 & a_3 & a_2 \end{bmatrix}, \sigma_V(p) = \begin{bmatrix} \underline{1} & \underline{2} & \underline{3} \\ a_1 & a_1 & a_2 \\ a_3 & a_2 & a_1 \\ a_2 & a_3 & a_3 \end{bmatrix}.$$

*Next, let* $\sigma_A = (a_3, a_1, a_2)$ *be a permutation on the set of alternatives that relabels alternative* $a_3$ *as* $a_1$*,* $a_1$ *as* $a_2$*, and* $a_2$ *as* $a_3$*. Then for*

$$p = \begin{bmatrix} \underline{1} & \underline{2} & \underline{3} \\ a_1 & a_2 & a_1 \\ a_2 & a_1 & a_3 \\ a_3 & a_3 & a_2 \end{bmatrix}, \sigma_A(p) = \begin{bmatrix} \underline{1} & \underline{2} & \underline{3} \\ a_2 & a_3 & a_2 \\ a_3 & a_2 & a_1 \\ a_1 & a_1 & a_3 \end{bmatrix}.$$

**Definition 1.2.12** (Monotonicity)**.** *Let* $g$ *be a social welfare rule,* $V = \{1, \ldots, m\}$ *be the set of voters and* $A = \{a_1, \ldots, a_n\}$ *be the set of alternatives. Let* $p, q \in L_W(A)^m$ *be profiles such that there exists* $i \in V$ *and* $a_k \in A$ *such that* $p(j) = q(j)$ *for all* $j \neq i$ *and*

$$\{a_\ell \in A \,:\, a_k \succ_{p(i)} a_\ell\} \subset \{a_\ell \in A \,:\, a_k \succ_{q(i)} a_\ell\}.$$

*Then g is monotonic if and only if*

$$\{a_\ell \in A \ : \ a_k \succ_{g(p)} a_\ell\} \subset \{a_\ell \in A \ : \ a_k \succ_{g(q)} a_\ell\}.$$

Qualitatively, monotonicity requires that a voter cannot cause the demotion of an alternative in the societal preference outcome by promoting the alternative in his or her preference ordering.

The previous definitions have referred to the behavior of a social welfare rule over a fixed domain with a fixed number of voters and alternatives. The following property concerns the behavior of a social welfare rule with respect to varying numbers of individuals.

**Definition 1.2.13** (Consistency)**.** *Let $g$ be a social welfare rule, $A$ be the set of alternatives, $V_1$ and $V_2$ be disjoint sets of voters, $p_1$ a profile on $V_1$ and $A$ and $p_2$ a profile on $V_2$ and $A$, with $g_1(p_1) = g_1(p_2)$. Let $p$ be the profile on $V = V_1 \cup V_2$ such that $p(i) = p_1(i)$ for all $i \in V_1$ and $p(i) = p_2(i)$ for all $i \in V_2$. Then $g$ is consistent if and only if $g_1(p) = g_1(p_1) = g_1(p_2)$.*

Again, we define a dual property which describes the behavior of social welfare rules with respect to varying numbers of alternatives.

**Definition 1.2.14** (Independence of Irrelevant Alternatives)**.** *Let $g$ be a social choice welfare rule, and let $V$ be the set of voters and $A$ be the set of alternatives. Furthermore, define $A^+ = A \cup \{a_{n+1}\}$ and for any profile $p$ on the alternatives in $A$, define $p^+$ such that $\forall\ k \in N$ and $\forall\ a_i, a_j \in A$, $a_i \succ_{p(k)} a_j$ implies that $a_i \succ_{p^+(k)} a_j$. Then $g$ satisfies independence of irrelevant alternatives (IIA) if and only if $\forall\ a_i, a_j \in A$, $a_i \succ_{g(p)} a_j$ implies that $a_i \succ_{g(p^+)} a_j$.*

Intuitively, IIA requires that the introduction of new alternatives cannot change the relative ranking of other alternatives in a societal preference outcome. A classic anecdote illustrating a violation of IIA was proposed by Columbia University philosophy professor Sidney Morgenbesser:

> After finishing dinner, I decided to order dessert. The waitress told me there were two choices: apple pie and

> blueberry pie. I ordered the apple pie. After a few minutes the waitress returned and said that they also have cherry pie at which point I replied "In that case I'll have the blueberry pie."

In later chapters we will see examples of social welfare rules that satisfy IIA and social welfare rules that do not.

Of the above properties, a number are considered desirable for normative reasons. Unanimity, anonymity, neutrality, and consistency appeal to universal ideas of fairness and equal treatment. Independence of irrelevant alternatives if considered by many an equally natural requirement for a social welfare rule, although it is considerably more controversial than the other properties under consideration. With the notation and definitions we have developed, we next restate Arrow's Theorem more precisely.

**Theorem 1.2.15** (Arrow's Impossibility Theorem)**.** *Let $V$ be the set of voters and $A$ be the set of alternatives, with $|V| = m \geq 2$ and $|A| = n \geq 3$. Let $g : \wp \longrightarrow L_W(A)$ be a social welfare rule. Then $g$ must fail to satisfy one of the following properties:*

1. *The domain $\wp = L_W(A)^m$.*
2. *$g$ is non-dictatorial.*
3. *$g$ is consistent.*
4. *$g$ is monotonic.*
5. *$g$ is onto.*

The above definitions and notation are largely standard to the social choice literature. In some cases (as mentioned) we have chosen to use alternative definitions for convenience, and have introduced our own notation for some concepts. In the next section, we introduce a new procedure that we believe has beneficial implications for research in social choice theory.

## 1.3 Alternative Representations of Voter and Societal Preferences

We next introduce a procedure that allows for voter preference orderings and societal preference outcome to be encoded into a simple mathematical object: a vector of integers. By using a vector representation, we allow for the use of standard geometric and linear algebraic tools. In Chapter 4 we will further investigate the many advantages of a matricial approach.

Given a voter preference ordering $p(\ell)$ on the set alternatives $A = \{a_1, \ldots, a_n\}$, we can encode the voter's preferences in a $\pm 1/0$-vector in $\mathbb{R}^{\binom{n}{2}}$ via the following algorithm.

**Algorithm 1.3.1** (Ternary Representation)**.** *Given a voter preference ordering $p(\ell)$ we generate $v(\ell) \in \mathbb{R}^{\binom{n}{2}}$ by*

1. *Fix an ordering of the alternatives, e.g. $(a_1, a_2, \ldots, a_n)$.*
2. *Fix an ordering of all $\binom{n}{2}$ unique pairings of alternatives, e.g.*

$$x_1 = (a_1, a_2), x_2 = (a_1, a_3), \ldots, x_{n-1} = (a_1, a_n), x_n = (a_2, a_3), \ldots.$$

3. *For $x_k = (a_i, a_j)$,*
   (a) *If $a_i \succ_{p(\ell)} a_j$, set $v(/l)_k = 1$.*
   (b) *If $a_i \sim_{p(\ell)} a_j$, set $v(/l)_k = 0$.*
   (c) *If $a_i \prec_{p(\ell)} a_j$, set $v(/l)_k = -1$.*

This algorithm assigns to each voter preference ordering a unique $\pm 1/0$-vector in $\mathbb{R}^{\binom{n}{2}}$. However, when voter preference orderings are elements of $L_W(A)$, this algorithm does not produce a 1–1 correspondence, as the following example illustrates.

**Example 1.3.2.** *Let $v = (1, -1, 1) \in \mathbb{R}^{\binom{3}{2}}$. If we attempt to reverse our algorithm to generate a voter preference ordering for some voter $\ell$, we arrive at*

$$p(\ell) = (a_1 \succ a_2 \succ a_3 \succ a_1).$$

*Clearly $p(\ell) \notin L_W(A)$ since it is not a weak linear order (it contains a cycle).*

Algorithm 1.3.1 can also be used to create a vector representation of a societal preference outcome $g(p)$ by replacing $p(\ell)$ with $g(p)$. Whether or not the algorithm produces a 1–1 correspondence between societal preference outcomes and $\pm 1/0$-vectors in $\mathbb{R}^{\binom{n}{2}}$ depends on the social welfare rule being used.

In addition to the $\pm 1/0$-vector representation, societal preference outcomes can be naturally represented by directed graphs (digraphs). We next present an algorithm for creating a digraph representation of a societal preference outcome, $g(p)$.

**Algorithm 1.3.3.** *Given a societal preference outcome $g(p)$ we generate $G(A, g(p))$ by*

1. *Fix an ordering of the alternatives, e.g. $(a_1, a_2, \ldots, a_n)$.*
2. *Assign each alternative to a node of the graph.*
3. *For all alternative pairs $(a_i, a_j)$, add a directed arc from $a_i$ to $a_j$ if and only if $a_i \succ_{g(p)} a_j$.*

We next present a concrete example.

**Example 1.3.4.** *Let $A = \{a_1, a_2, a_3\}$ be the set of alternatives, $g$ be a social welfare rule, and $p$ be a profile such that $g(p) = (a_1 \succ a_2 \succ a_3)$. Then by Algorithm 1.3.3 be can represent $g(p)$, as shown in Figure 1.1.*

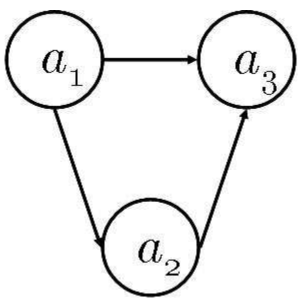


Figure 1.1: Digraph Representation of Societal Preference Outcome

Graphical representations will be of extreme importance for visualizing social preference outcomes. By assumption we have restricted voter preferences to be linear orders, which permit a number of natural visual representations including lists (potentially with ties). As we shall see in the next chapter, societal preference outcomes need not be linear orders, even when voter preferences are. In this case, lists are no longer appropriate vehicles for visual representations.

When voter preference orderings are restricted to $L_S(A)$, we will sometimes wish to employ a binary vector representation rather than a ternary representation. The algorithm is presented below.

**Algorithm 1.3.5** (Binary Representation)**.** *Given a voter preference ordering* $p(\ell) \in L_S(A)$ *we generate* $v(\ell) \in \mathbb{R}^{\binom{n}{2}}$ *by*

1. *Fix an ordering of the alternatives, e.g.* $(a_1, a_2, \ldots, a_n)$*.*
2. *Fix an ordering of all* $\binom{n}{2}$ *unique pairings of alternatives, e.g.*

$$x_1 = (a_1, a_2), x_2 = (a_1, a_3), \ldots, x_{n-1} = (a_1, a_n), x_n = (a_2, a_3), \ldots.$$

3. *For* $x_k = (a_i, a_j)$*,*
   (a) *If* $a_i \succ_{p(\ell)} a_j$*, set* $v(/l)_k = 1$*.*
   (b) *Else, set* $v(/l)_k = 0$*.*

# Chapter 2

# Majority Rule

We next formally introduce a social welfare rule that is well-known outside the literature: simple majority rule. We begin with a discussion of generalized majority rules and their properties. We next introduce simple majority rule, and compare it with other generalized majorities rules. We follow with a known result and original linear algebraic proof that is a 1–1 correspondence between simple majority rule societal preference outcomes and ±1/0-vectors; refinements of this result are also presented. Finally, we investigate the distribution of societal outcomes and present some cursory findings.

## 2.1 Generalized Majority Rules

Define a generalized majority rule to be the social welfare rule $g_{M_\gamma}$ : $\wp \longrightarrow R(A)$ such that $\forall a_i, a_j \in A$,

$$a_i \succ_{g_{M_\gamma}(p)} a_j \quad \Leftrightarrow \quad |V_p(a_i \succ a_j)| > \gamma m,$$

and

$$a_i \not\succ_{g_{M_\gamma}(p)} a_j \text{ and } a_j \not\succ_{g_{M_\gamma}(p)} a_i \Rightarrow a_i \sim_{g_{M_\gamma}(p)} a_j,$$

for a fixed $\gamma \in [\frac{m}{2}, 1]$. That is, alternative $a_i$ is preferred to alternative $a_j$ in the societal preference outcome $g(p)$ if and only if more than $\gamma m$ of the voters prefer $a_i$ to $a_j$ at profile $p$. If $a_i$ is not preferred to $a_j$ in $g(p)$ and $a_j$ is not preferred to $a_i$ in $g(p)$, then $g(p)$ is indifferent between $a_i$ and $a_j$. By construction, it is clear that for all profiles $p$, $g_{M_\gamma}(p)$ is a complete antisymmetric relation on $A$.

As we shall next prove, generalized majority rules have a number of desirable properties.

**Lemma 2.1.1.** *Generalized majority rules are unanimous, anonymous (and thus non-dictatorial), neutral, consistent, and satisfy independence of irrelevant alternatives.*

*Proof.* Let $V = \{1, \ldots, m\}$ be the set of voters and $A = \{a_1, \ldots, a_n\}$ be the number of alternatives.

Suppose that at profile $p$ there exists $a_i \in A$ such that $p_1(k) = a_i$ for all $k \in V$. Then $V_p(a_i \succ a_j) = V$ so that $|V_p(a_i \succ a_j)| = m > \gamma m$ for any $\gamma \in (0, 1)$. Therefore alternative $a_i$ is the unique most-preferred alternative in $g_{M_\gamma}(p)$ so generalized majority rules have the unanimity property.

Next let $\sigma_V$ be some permutation of the voters. At any profile $p$ and for all $i, j$, $|V_p(a_i \succ a_j)| = |V_{\sigma_V(p)}(a_i \succ a_j)|$, although the composition of the coalitions will differ in the natural way induced by $\sigma_V$. Since it is only the size of the coalitions that determines the societal ordering, we therefore have that $g_{M_\gamma}(p) = g_{M_\gamma}(\sigma_V(p))$, so generalized majority rules are anonymous. Since generalized majority rules are anonymous, they are also non-dictatorial.

Next let $\sigma_A$ be some permutation of the names of the alternatives, and let $\sigma_A(p)$ be the permutation of a profile $p$ induced by $\sigma_A$. Then for all $i, j$,

$$|V_p(a_i \succ a_j)| = |V_{\sigma_A(p)}(\sigma_A(a_i) \succ \sigma_A(a_j))|,$$

so that $\sigma_A(g_C(p)) = g_{M_\gamma}(\sigma_A(p))$, so generalized majority rules are neutral.

Let $V_1$ ($|V_1| = m_1$) and $V_2$ ($|V_2| = m_2$) be two disjoint sets of voters. Let profile $p_1$ contain preference orderings for voters in $V_1$, and profile $p_2$ contain preference orderings for voters in $V_2$. Let $p$ be the profile on $V = V_1 \cup C_2$ such that $p$ coincides with $p_1$ on $V_1$ and with $p_2$ on $V_2$. If for some pair $i, j$, $a_i \succ_{g_{M_\gamma}(p_1)} a_j$ and $a_i \succ_{g_{M_\gamma}(p_2)} a_j$

then by definition

$$|V_{p_1}(a_i \succ a_j)| > \gamma m_1 \qquad \text{and} \qquad |V_{p_2}(a_i \succ a_j)| > \gamma m_2$$

and therefore

$$|V_p(a_i \succ a_j)| = |V_{p_1}(a_i \succ a_j)| + |V_{p_2}(a_i \succ a_j)| > \gamma m_1 + \gamma m_2 = \gamma(m_1 + m_2).$$

Therefore $a_i \succ_{g_{M_\gamma}(p)} a_j$, so generalized majority rules are consistent.

Let $V = \{1, \ldots, m\}$ be a set of voters, $A = \{a_1, \ldots, a_n\}$ be a set of alternatives, $p$ be a profile on the voters in $V$ and the alternatives in $A$. Define $A^+ = A \cup \{a_{n+1}\}$ and a profile $p^+$ such that $\forall\ k \in V$ and $\forall\ a_i, a_j \in A$, $a_i \succ_{p(k)} a_j$ implies that $a_i \succ_{p^+(k)} a_j$.

Suppose that $a_i \succ_{g_{M_\gamma}(p)} a_j$, so that $V_p(a_i \succ a_j) > \frac{m}{2}$. By construction, $V_{p^+}(a_i \succ a_j) = V_p(a_i \succ a_j) > \frac{m}{2}$, so $a_i \succ_{g_{M_\gamma}(p^+)} a_j$. Therefore generalized majority rules satisfy independence of irrelevant alternatives. □

Although all generalized majority rules have the properties of the previous lemma, there are three particular values of $\gamma$ that are most frequently studied: $\gamma = \frac{1}{2}$, $\gamma = \frac{2}{3}$, and $\gamma = 1$. These values of $\gamma$ correspond to simple majority rule, 2/3-majority rule and unanimous rule, respectively. We next consider simple majority rule is greater detail.

## 2.2 Simple Majority Rule

We next focus on a particular generalized majority rule known as simple majority rule, for which $\gamma = \frac{1}{2}$. As a generalized majority rule, simple majority rule possesses the properties of the previous lemma; in particular, it is non-dictatorial, consistent, monotonic, and onto. From Arrow's Theorem, there must therefore exist profiles at which simple majority rule fails to produce a societal preference outcome that is a weak linear order. The following example illustrates such a profile, and is attributed to Condorcet [3].

**Example 2.2.1** (Condorcet's Voting Paradox)**.** *Let* $V = \{1, 2, 3\}$ *and* $A = \{a_1, a_2, a_3\}$ *and consider the profile*

$$p = \begin{bmatrix} \underline{1} & \underline{2} & \underline{3} \\ a_1 & a_3 & a_2 \\ a_2 & a_1 & a_3 \\ a_3 & a_2 & a_1 \end{bmatrix}.$$

*At this profile, we have*

$$\begin{aligned} V_p(a_1 \succ a_2) &= \{1,2\}, \text{ so } |V_p(a_1 \succ a_2)| &= 2 > \tfrac{3}{2}, \\ V_p(a_2 \succ a_3) &= \{1,3\}, \text{ so } |V_p(a_2 \succ a_3)| &= 2 > \tfrac{3}{2}, \\ V_p(a_3 \succ a_1) &= \{2,3\}, \text{ so } |V_p(a_3 \succ a_1)| &= 2 > \tfrac{3}{2}. \end{aligned}$$

*Therefore, the simple majority rule societal preference outcome is the cycle*

$$g_M(P) = (a_1 \succ a_2 \succ a_3 \succ a_1).$$

In this example the societal outcome was a simple three-cycle. With $n$ alternatives, we can construct profiles at which the majority rule societal outcome contains a $k$ cycle for any $3 \leq k \leq n$. The next example illustrates a profile at which the simple majority rule societal preference outcome contains a number of cycles of various lengths.

**Example 2.2.2.** *Let* $V = \{1, 2, 3\}$ *and* $A = \{a_1, a_2, a_3 a_4, a_5\}$*. Consider the profile*

$$p = \begin{bmatrix} \underline{1} & \underline{2} & \underline{3} \\ a_1 & a_3 & a_2 \\ a_4 & a_5 & a_4 \\ a_2 & a_1 & a_3 \\ a_3 & a_4 & a_5 \\ a_5 & a_2 & a_1 \end{bmatrix}.$$

*It can be verified that the simple majority rule societal outcome contains the three-cycles*

$$a_1 \succ a_2 \succ a_3 \succ a_1,$$

$$a_1 \succ a_4 \succ a_3 \succ a_1,$$

$$a_2 \succ a_3 \succ a_5 \succ a_2,$$

$$a_1 \succ a_4 \succ a_3 \succ a_1,$$

*the four-cycles*

$$a_1 \succ a_2 \succ a_3 \succ a_5 \succ a_1,$$

$$a_2 \succ a_3 \succ a_1 \succ a_5 \succ a_2,$$

*and the five-cycle*

$$a_2 \succ a_3 \succ a_5 \succ a_1 \succ a_4.$$

Not all generalized majority rules suffer from cyclic societal outcomes. Abello and Johnson [1] have shown that for $\gamma \geq \frac{2}{3}$, $g_{M_\gamma}$ produces transitive societal outcomes. However, this result should not be interpreted as unambiguous support for the generalized majority rule with $\frac{2}{3} \leq \gamma \leq 1$ over other generalized majority rules, as the following example illustrates.

**Example 2.2.3.** *Let $V = \{1, \ldots, 8\}$ be the set of voters and $A = \{a_1, a_2\}$ be the set of alternatives. Consider the profile*

$$p = \begin{bmatrix} \underline{1} & \underline{2} & \underline{3} & \underline{4} & \underline{5} & \underline{6} & \underline{7} & \underline{8} \\ a_1 & a_1 & a_1 & a_1 & a_1 & a_2 & a_2 & a_2 \\ a_2 & a_2 & a_2 & a_2 & a_2 & a_1 & a_1 & a_1 \end{bmatrix}.$$

*At this profile, we have*

$$g_{M_{(2/3)}}(p) = (a_1 \sim a_2), \qquad \textit{and} \qquad g_{M_{(1/2)}}(p) = (a_1 \succ a_2).$$

*Therefore, the generalized majority rule $g_{M_{(2/3)}}$ produces a societal outcome that fails to distinguish between the two alternatives although nearly twice as many voters prefer $a_1$ to $a_2$. Here however simple majority rule $g_{M_{(1/2)}}$ produces a societal preference outcome in which alternative $a_1$ is most-preferred, and moreover, which is a strict linear order.*

For a fixed number of alternatives cycles can arise in a simple majority societal preference outcome in a number of ways. We will say that a societal preference outcome $g_M(p)$ is *intransitive* if it contains any cycle of any length. As a matter of practical implementation, we may wish to distinguish between different types of intransitive societal preference outcomes. We propose an example.

**Example 2.2.4.** *Let* $V = \{1, 2, 3\}$ *be the set of voters and* $A = \{a_1, a_2, a_3, a_4\}$ *be the set of alternatives. Consider the profiles*

$$p = \begin{bmatrix} \underline{1} & \underline{2} & \underline{3} \\ a_1 & a_1 & a_1 \\ a_2 & a_4 & a_3 \\ a_3 & a_2 & a_4 \\ a_4 & a_3 & a_2 \end{bmatrix}, \qquad q = \begin{bmatrix} \underline{1} & \underline{2} & \underline{3} \\ a_1 & a_3 & a_2 \\ a_2 & a_1 & a_3 \\ a_3 & a_2 & a_1 \\ a_4 & a_4 & a_4 \end{bmatrix}.$$

*The simple majority outcome at profiles p and q are depicted in Figure 2.1 and Figure 2.2, respectively. From these depictions, we see that the outcome at profile p is in some sense "nicer," especially if we wish to use the societal preference outcome to choose a single "winning" alternative.*

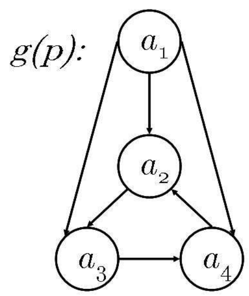


Figure 2.1: Digraph Representation of $g(p)$

We therefore propose the following measure of transitivity. For the sake of simplicity, we will assume that there are an odd number of voters and that voter preferences are restricted to $L_S(A)$; this will ensure that there are no ties in the simple majority rule societal preference ordering. Then given a simple majority rule societal pref-

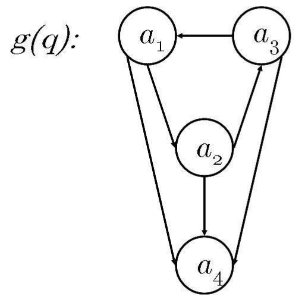


Figure 2.2: Digraph Representation of $g_M(q)$

erence outcome, $g_M(p)$, on a set of alternatives $A = \{a_1, \ldots, a_n\}$, let $v_p$ the binary vector representations of $g_M(p)$ given by Algorithm 1.3.5. Moreover, define the set $\mathcal{T}$ to be the set of binary vector representations of all transitive simple majority rule societal preference outcomes on the set of alternatives $A$. Then define $\tau(p)$ to be the *transitivity score* of $g_M(p)$, where

$$\tau(p) = \max_{t \in \mathcal{T}} \frac{v_p \cdot t}{\binom{n}{2}}.$$

By the normalization, $0 \leq \tau(p) \leq 1$ for all profiles $p$.

Given a profile $p$ and the transitivity score $\tau(p)$, we can interpret $\tau(p)$ as the distance from the simple majority rule societal preference outcome $g_M(p)$ to the nearest transitive societal preference outcome.

**Example 2.2.5.** *Let $A = \{a_1, \ldots, a_4\}$ be the set of alternatives, and let $p$ be a profile such that $g_M(p)$ has binary vector representation*

$$v_p = (0 \quad 1 \quad 1 \quad 0 \quad 1 \quad 1).$$

*Then*

$$\tau(p) = \max_{t \in \mathcal{T}} \frac{v_p \cdot t}{6} = 0.667.$$

As a matter of theory it may be sufficient to know that profiles such as those in the previous examples exist. However, as a matter of practical implementation it is important to know how many such profiles occur or, equivalently, the probability of producing an intransitive simple majority rule societal preference outcome.

## 2.3 Attainability of Societal Outcomes

We have shown that every societal preference ordering on $n$ alternatives has a unique $\pm 1/0$-vector representation in $\mathbb{R}^{\binom{n}{2}}$. Conversely, we may ask if an arbitrary $\pm 1/0$-vector in $\mathbb{R}^{\binom{n}{2}}$ can be a societal preference outcome.

The problem was first solved by McGarvey[13], who proved the following equivalent theorem:

**Theorem 2.3.1** (McGarvey, 1953)**.** *Given an arbitrary preference pattern over a set of n elements, a group of individuals exists with strong individual preference orderings such that the group preference pattern as determined by the method of simple majority decision is the given preference pattern.*

McGarvey employed a constructive proof, which we summarize here. For continuity, we have adapted our notation to the proof.

*Proof.* Let $p$ be an arbitrary preference pattern over a set of $n$ elements, $\{a_1, \ldots, a_n\}$. For each element pair $(a_i, a_j)$ with $a_i \succ_p a_j$, relabel the remaining alternatives as $\{a_1, \ldots, a_{n-2}\}$ in any manner. Then introduce two new voters to the society with preference orderings

$$p(1) = (a_i \succ a_j \succ a_1 \succ a_2 \succ \cdots \succ a_{n-2})$$

and

$$p(2) = (a_i \succ a_j \succ a_{n-2} \succ a_{n-3} \succ \cdots \succ a_1).$$

By construction, simple majority rule applied to a society consisting of this pair will produce a societal preference outcome in which $a_i \succ a_j$, $a_i \succ a_k$ for all $k \neq i$, $a_j \succ a_k$ for all $k \neq i, j$, and for all $a_k \sim a_\ell$ for all $k, l \neq i, j$.

Therefore, when simple majority rule is applied to the entire society, the societal preference outcome will coincide with the preference pattern $p$. □

We propose a linear algebraic proof of McGarvey's theorem.

*Proof.* Let $v \in \mathbb{R}^{\binom{n}{2}}_{\pm 1/0}$ be the vector representation of an arbitrary preference pattern with fixed ordering of the alternatives $a_1, a_2, \ldots, a_n$. Define the matrix

$$T = \begin{bmatrix} 1 & 1 & 1 & \cdots & 1 \\ -1 & 1 & 1 & \cdots & 1 \\ -1 & -1 & 1 & \cdots & 1 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ -1 & -1 & -1 & \cdots & 1 \end{bmatrix}_{\binom{n}{2} \times \binom{n}{2}}$$

to be an $\binom{n}{2} \times \binom{n}{2}$ matrix with

$$\begin{array}{llll} t_{ij} = +1 & \text{for} \quad i = 1, \ldots, \binom{n}{2} & \text{and} & j = 1, \ldots, i \\ t_{ij} = -1 & \text{for} \quad i = 1, \ldots, \binom{n}{2} & \text{and} & j = i+1, \ldots, n \end{array}$$

Letting $\xi(T)$ denote the matrix $T$ after performing the following elementary row operations:

1. Add Row 1 to Row $i$, for $i = 2, \ldots, \binom{n}{2}$;
2. Multiply Row $i$ by $\frac{1}{2}$, for $i = 2, \ldots, \binom{n}{2}$.

Then we have

$$\xi(T) = \begin{bmatrix} 1 & 1 & 1 & \cdots & 1 \\ 0 & 1 & 1 & \cdots & 1 \\ 0 & 0 & 1 & \cdots & 1 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 1 \end{bmatrix}_{\binom{n}{2} \times \binom{n}{2}} ,$$

an upper triangular matrix. Clearly, $\det(\xi(T)) = 1$ so that $\det(T) \neq 0$. Therefore, the column vectors $T_i$ $(i = 1, \ldots, \binom{n}{2})$ form a spanning set of $\mathbb{R}^{\binom{n}{2}}$. Moreover, it can be verified that each column vector $T_i$ $(i = 1, \ldots, \binom{n}{2})$ corresponds to strict linear order on the set of alternatives $A$ by considering a partition of the components of a given vector $T_i$.

Given $T_i$, we form a partition $P = \{\rho_1, \ldots, \rho_{n-1}\}$ of the components as follows

$$
\begin{aligned}
\rho_1 &= \{t_{i\binom{n}{2}}\} \\
\rho_2 &= \left\{t_{i(\binom{n}{2}-1)}, t_{i(\binom{n}{2}-2)}\right\} \\
\vdots \quad & \quad \vdots \\
\rho_{n-2} &= \{t_{in}, t_{i(n+1)}, \ldots, t_{i(2n-3)}\} \\
\rho_{n-1} &= \{t_{i1}, t_{i2}, \ldots, t_{i(n-1)}\}
\end{aligned}
$$

so that $|\rho_k| = k$. We show that this does indeed produce the desired partition, as

$$
\binom{n}{2} = \frac{n(n-1)}{2} = \sum_{k=1}^{n-1} k
$$

and clearly $\rho_k \cap \rho_m = \emptyset$ for $k \neq m$. More intuitively, this partition corresponds to

$$
T_i = \left[\begin{array}{cc}
 & t_{i1} \\
(\rho_{n-1}) & \vdots \\
 & t_{i(n-1)} \\
\hline
 & t_{in} \\
(\rho_{n-2}) & \vdots \\
 & t_{i(2n-3)} \\
\hline
 & \vdots \\
\hline
(\rho_2) & t_{i(\binom{n}{2}-1)} \\
 & t_{i(\binom{n}{2}-2)} \\
\hline
(\rho_1) & t_{i\binom{n}{2}}
\end{array}\right]
$$

so that after fixing an ordering of the alternatives, the vector components assigned to $\rho_{n-i}$ contain the outcomes of all comparisons of the alternatives $(a_i, a_j)$, for $i = i+1, \ldots, n$.

Starting with the final column vector of $T$, the following correspondences can also be easily verified:

$$
\begin{aligned}
T_{\binom{n}{2}} &\sim (a_1 \succ a_2 \succ \cdots \succ a_{n-1} \succ a_n), \\
T_{\binom{n}{2}-1} &\sim (a_1 \succ a_2 \succ \cdots \succ a_n \succ a_{n-1}), \\
T_{\binom{n}{2}-2} &\sim (a_1 \succ a_2 \succ \cdots \succ a_n \succ a_{n-2} \succ a_{n-1}).
\end{aligned}
$$

Moreover, for any $T_i$ $(i = 1, \ldots, \binom{n}{2})$ the strict linear order of the alternatives corresponding to $T_i$ can be found by the following algorithm:

1. Set $j = \binom{n}{2}$ and $r = 1$.
2. Begin with the natural ordering, $p(j+1) = (a_1 \succ \cdots \succ a_n)$.
3. While $t_{ij} = -1$,
   - If $t_{ij} \in \rho_r$, create $p(j)$ from $p(j+1)$ by promoting alternative $a_n$ one ranking, e.g. $(a_1, a_4, a_2, a_3) \to (a_4, a_1, a_2, a_3)$. Decrement $j$ by one unit.
   - Else, create $p(j)$ from $p(j+1)$ by permuting the alternatives ranked below $a_n$ to the right, e.g. $(a_1, a_4, a_2, a_3) \to (a_1, a_4, a_3, a_2)$. Increment $r$ and decrement $j$ by one unit.
4. Return $p(1)$.

Since this algorithm produces a permutation of the strict linear order $p\left(\binom{n}{2}+1\right)$, the resulting $p(1)$ is a strict linear order.

Returning to $v \in \mathbb{R}^{\binom{n}{2}}_{\pm 1/0}$, a vector representation of an arbitrary preference pattern with fixed ordering of the alternatives $a_1, a_2, \ldots, a_n$, we can therefore write $v$ as a linear combination of the columns of $T$,

$$v = \sum_{i=1}^{\binom{n}{2}} \alpha_i T_i.$$

As $v$ and the $T_i$ are rational, we can restrict the coefficients to be rational. Since our sum is finite we may further restrict the coefficients to be integer valued by clearing the denominators. Moreover, we note that given a strict linear order $T_i$, the vector $-T_i$ corresponds to the reversal of the order of $T_i$. Therefore, letting

$$T_i' = \begin{cases} -T_i & \text{if } \alpha_i < 0 \\ T_i & \text{otherwise} \end{cases} \qquad \text{and} \qquad \beta_i = |\alpha_i|,$$

we have the representation

$$v = \sum_{i=1}^{\binom{n}{2}} \beta_i T_i', \qquad \beta_i \in \mathbb{N},$$

with the following interpretation: Given $v$, an arbitrary preference pattern on $n$ alternatives, we may construct a society of $\beta = \sum_{i=1}^{\binom{n}{2}} \beta_i$ individuals, with $\beta_i$ individuals holding the preference ordering $T_i'$, for $i = 1, \ldots, \binom{n}{2}$. By construction, the societal majority rule outcome of this society is $v$. □

Since Algorithm 1.3.1 applied to any complete antisymmetric relation, we have confirmed McGarvey's result that an arbitrary preference relation on $n$ alternatives is a simple majority rule societal preference outcome of some society. Like McGarvey, we have shown that this society can be composed of individuals whose preferences are strict linear orders. We next consider a concrete example of our results.

**Example 2.3.2.** *Let $A\{a_1, a_2, a_3, a_4\}$ be the set of alternatives. In this case we have*

$$T = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 \\ -1 & 1 & 1 & 1 & 1 & 1 \\ -1 & -1 & 1 & 1 & 1 & 1 \\ -1 & -1 & -1 & 1 & 1 & 1 \\ -1 & -1 & -1 & -1 & 1 & 1 \\ -1 & -1 & -1 & -1 & -1 & 1 \end{bmatrix}$$

*on which we may perform elementary matrix operations*

$$T \overset{t_1+t_i, i=2,\ldots,6}{\Longrightarrow} \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 2 & 2 & 2 & 2 & 2 \\ 0 & 0 & 2 & 2 & 2 & 2 \\ 0 & 0 & 0 & 2 & 2 & 2 \\ 0 & 0 & 0 & 0 & 2 & 2 \\ 0 & 0 & 0 & 0 & 0 & 2 \end{bmatrix}$$

$$\overset{t_i/2, i=2,\ldots,6}{\Longrightarrow} \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} = \xi(T)$$

*Furthermore, we note that the preference relations corresponding to the columns of $T$ are strict linear orders:*

$$\begin{array}{lcl} T_1 &=& (1, \ -1, \ -1, \ -1, \ -1, \ -1) \ \sim \ (a_4 \succ a_3 \succ a_1 \succ a_2), \\ T_2 &=& (1, \ 1, \ -1, \ -1, \ -1, \ -1) \ \sim \ (a_4 \succ a_1 \succ a_3 \succ a_2), \\ T_3 &=& (1, \ 1, \ 1, \ -1, \ -1, \ -1) \ \sim \ (a_1 \succ a_4 \succ a_3 \succ a_2), \\ T_4 &=& (1, \ 1, \ 1, \ 1, \ -1, \ -1) \ \sim \ (a_1 \succ a_4 \succ a_2 \succ a_3), \\ T_5 &=& (1, \ 1, \ 1, \ 1, \ 1, \ -1) \ \sim \ (a_1 \succ a_2 \succ a_4 \succ a_3), \\ T_6 &=& (1, \ 1, \ 1, \ 1, \ 1, \ 1) \ \sim \ (a_1 \succ a_2 \succ a_3 \succ a_4). \end{array}$$

It is important to note that the linear combination representation of an arbitrary societal preference outcome is not necessarily unique, as the following example illustrates.

**Example 2.3.3.** *Let $V = \{1, 2, 3\}$ be the set of voters and $A = \{a_1, a_2, a_3\}$ be the set of alternatives. Consider the societal preference ordering*

$$g(p) = (a_1 \succ a_2 \succ a_3).$$

*Then each of the following specifications of the profile $p$ give rise to $g(p)$ as above.*

$$(1) \quad p = \begin{bmatrix} \underline{1} & \underline{2} & \underline{3} \\ a_1 & a_1 & a_1 \\ a_2 & a_2 & a_2 \\ a_3 & a_3 & a_3 \end{bmatrix}$$

$$(2) \quad p = \begin{bmatrix} \underline{1} & \underline{2} & \underline{3} \\ a_3 & a_1 & a_2 \\ a_1 & a_2 & a_1 \\ a_2 & a_3 & a_3 \end{bmatrix}$$

$$(3) \quad p = \begin{bmatrix} \underline{1} & \underline{2} & \underline{3} \\ a_1 & a_1, a_2 & a_1 \\ a_2, a_3 & a_3 & a_2 \\ & & a_3 \end{bmatrix}$$

In addition to the profiles in the previous example, we also note that the linear combination

$$v = \sum_{i=1}^{\binom{n}{2}} \beta_i T_i', \qquad \beta_i \in \mathbb{N},$$

is not unique in that once we have fixed a labeling of the voters, there are

$$\prod_{i=1}^{\binom{n}{2}-1} \binom{\beta - \sum_{j=1}^{i-1} \beta_j}{\beta_i}$$

ways to assign the preference orderings to the $\beta$ individuals.

## 2.4 Minimum Voter Attainability Thresholds

We have shown that an arbitrary complete antisymmetric relation on $n$ alternatives is a simple majority rule societal preference outcome of some society. By example, we have seen that a given societal preference outcome may be the result of majority rule applied to distinct profiles on a fixed number of voters. The following example illustrates that the number of voters necessary to produce a given societal preference outcome is also not unique.

**Example 2.4.1.** *Let $A = \{a_1, a_2, a_3\}$ be the set of alternatives and consider the societal preference ordering*

$$g(p) = (a_1 \succ a_2 \succ a_3).$$

*Then each of the following specifications of the profile p give rise to $g(p)$ as above.*

$$(1) \quad p = \begin{bmatrix} \underline{1} \\ a_1 \\ a_2 \\ a_3 \end{bmatrix}$$

$$(2) \quad p = \begin{bmatrix} \underline{1} & \underline{2} & \underline{3} \\ a_1 & a_1 & a_1 \\ a_2 & a_2 & a_2 \\ a_3 & a_3 & a_3 \end{bmatrix}$$

$$(3) \quad p = \begin{bmatrix} \underline{1} & \underline{2} & \underline{3} & \underline{4} & \underline{5} \\ a_3 & a_1 & a_2 & a_1 & a_3 \\ a_1 & a_2 & a_1 & a_2 & a_2 \\ a_2 & a_3 & a_3 & a_3 & a_1 \end{bmatrix}$$

The above example illustrates a trivial manner in which the number of voters can be increased without changing the societal preference outcome: pairs of voters can be added whose preference orderings are opposite. There are indeed more complex ways in which the number of voters can vary, and we wish to determine the minimum number of voters necessary to produce a given societal preference outcome. Let $\mu(n)$ denote the minimum number of voters with preference orderings in $L_S(A)$ needed to obtain all societal preference outcomes on $n$ alternatives.

Although the linear algebraic proof of attainability is more elegant than direct construction, it offers no insight concerning the function $\mu$. However, using the vector representation we have developed we can provide a lower bound on $\mu(n)$.

**Lemma 2.4.2.** *If $A = \{a_1, \dots, a_n\}$ is the set of alternatives, then*

$$\mu(n) \geq \left\lceil \binom{n}{2} \frac{\ln(3)}{\ln(n!)} \right\rceil .$$

*Proof.* Let $V = \{1, \dots, m\}$ be the set of voters and $A = \{a_1, \dots, a_n\}$ be the set of alternatives. Since voter preference orderings are strict linear orderings, there are $n!$ preference orderings voters may hold. Suppose that every assignment of preference orderings to the voters produces a unique societal preference ordering, so that $m$ voters can produce $n!^m$ distinct societal preference outcomes. Note that this is a severe overestimate, as the anonymity property of simple majority rules ensures that relabeling of the voters will have no impact on the societal preference outcome.

For each pair of alternatives $(a_i, a_j)$, there are three possible relations that can exist in the societal preference ordering: $a_i \succ a_j$, $a_i \sim a_j$, or $a_i \prec a_j$. Therefore there are $3^{\binom{n}{2}}$ possible preference patterns on $n$ alternatives.

Therefore, the minimum number of voters $m$ must be such that

$$3^{\binom{n}{2}} \leq (n!)^m .$$

Solving for $m$ gives

$$\mu(n) \geq \left\lceil \binom{n}{2} \frac{\ln(3)}{\ln(n!)} \right\rceil ,$$

where we take the ceiling since $m \in \mathbb{Z}$.

$$\begin{aligned} \mu(n) &> \max \left\{ m \mid (n!)^m \geq 3^{\binom{n}{2}} \right\} \\ &= \max \left\{ m \mid m \ln(n!) \geq \tbinom{n}{2} \ln(3) \right\} \\ &= \max \left\{ m \mid m \geq \tbinom{n}{2} \frac{\ln(3)}{\ln(n!)} \right\} \\ &= \left\lceil \tbinom{n}{2} \frac{\ln(3)}{\ln(n!)} \right\rceil . \end{aligned}$$

□

It is important to note that the function $\mu(n)$ gives the minimum number of voters necessary to produce an arbitrary simple majority rule societal preference outcome. This number may be greater than the minimum number of voters needed to produce a specific outcome. Letting $\mathcal{S}$ represent the set of social preference outcomes on $n$ alternatives,

$$\mu(n) = \max_{s \in \mathcal{S}} \min \{m \ : \ m \text{ voters are sufficient to produce } s\}.$$

The minimum number of voters needed to produce a specific simple majority rule societal outcome on $n$ alternatives may differ substantially from $\mu(n)$; the most extreme example is that of a transitive societal preference outcome, in which one voter is sufficient.

**Example 2.4.3.** *Let $A = \{a_1, \ldots, a_n\}$ be the set of alternatives and $g_M(p)$ be a transitive simple majority rule societal outcome. Then we can construct such a profile $p$ with only one voter, by setting $p(1) = g(p)$.*

The work of McGarvey and others provides improved bounds. Using McGarvey's constructive approach to proving attainability requires $m = 2\binom{n}{2}$ voters, where $n$ is the number of alternatives. This arises from construction, since for each unique pair $(a_i, a_j)$ two voter preference orderings are assigned. Not surprisingly McGarvey's method can be modified so as to decrease the number of voters required by incorporating the preference relations on multiple alternative pairs $(a_i, a_j)$ into each voter's preference ordering. A detailed discussion of the modified algorithm is given in McGarvey[13]. With these modifications, at least $m = n(n+1)$ voters are needed, where $n$ is the number of alternatives.

Stearns [18] shows that the information included in each voter preference ordering can be further compacted, so that $\mu(n) \leq n+1$ when $n$ is odd. Using an approximation for $n!$, Stearns further shows that $\mu(n) > \frac{0.55n}{\ln n}$ when $n$ is large. Figure 2.3 displays Stearns upper and lower bounds on the minimum number $m$ of voters needed to obtain an arbitrary societal preference outcome on $n$ alternatives, as well as the bounds computed here. Table 2.1 summarizes the results.

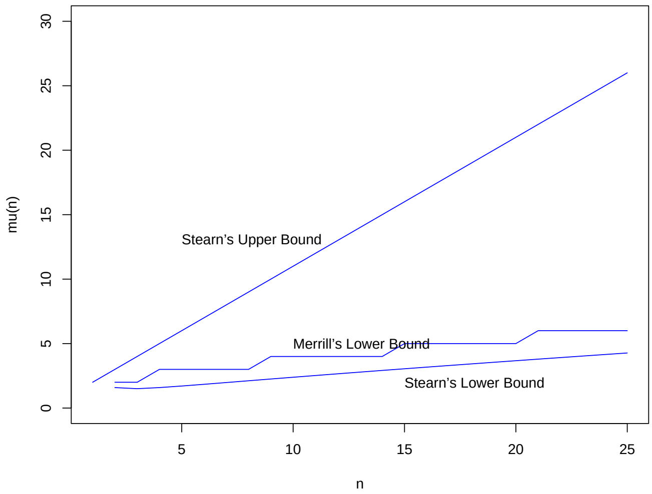


Figure 2.3: Upper and Lower Minimum Voter Attainability Bounds

| | | |
|---|---|---|
| McGarvey: | Naive Construction | $m = 2\binom{n}{2}$ |
| | Compact Construction | $m = n(n+1)$ |
| Stearns: | $n$ Odd | $\mu(n) \leq n+1$ |
| | $n$ Large | $\mu(n) > \dfrac{0.55n}{\ln n}$ |
| Merrill: | Any $n$ | $\mu(n) \geq \left\lceil \binom{n}{2} \dfrac{\ln(3)}{\ln(n!)} \right\rceil$ |

Table 2.1: Minimum Voter Attainability Results

As a final note, Stearns' lower bound and the lower bound derived here demonstrate that no finite number of voters is ever sufficient to produce all societal preference outcomes if the number of alternatives is made sufficiently large.

## 2.5 The Distribution of Societal Outcomes

We have seen that societal preference outcomes from simple majority rule may contain cycles, and that the existence of such cycles complicates practical decision-making processes. We have also seen that majority rules possess a number of desirable properties. Thus as a mater of practical implementation the frequency of intransitive simple majority rule societal preference outcomes is of vital importance: if the probability of obtaining an intransitive societal preference outcome is small, we may wish to implement simple majority rule to enjoy its desirable properties (most of the time). In situations where majority rule fails to produce a transitive outcome, another rule could be used that guarantees transitivity.

We therefore wish to determine the probability distribution of all simple majority rule societal preference outcomes with respect to the profile space, as a function of the number of voters and number of alternatives. This is equivalent to determining the probability distribution of the intransitive societal preference outcomes alone, as symmetry ensures that all transitive outcomes will be equally likely.

Our vector representations provide a constructive method for counting the total number of societal outcomes. For any set of voters $V = \{1, \dots, m\}$ and alternatives $A = \{a_1, \dots, a_n\}$ where $m$ is even, there are exactly $3^{\binom{n}{2}}$ societal preference outcomes, since there are three possible values $(-1, 0, 1)$ for each of the $\binom{n}{2}$ components of a societal preference outcome vector representations, and each unique assignment of the components produces a unique social preference outcome. As previously discussed, when $m$ is odd there are only $2^{\binom{n}{2}}$ societal preference outcomes. Moreover exactly $n!$ of the societal preference outcomes are transitive regardless of the parity of $m$, because there are exactly $n!$ unique permutations of the natural transitive ordering

$$a_1 \succ a_2 \succ \cdots \succ a_{n-1} \succ a_n.$$

A naive approach might suppose that each societal outcome is

| $n$ | $P(T\|n)$, $m$ Even | $P(T\|n)$, $m$ Odd |
|---|---|---|
| 3 | 0.222 | 0.750 |
| 4 | 0.033 | 0.375 |
| 5 | 0.002 | 0.117 |
| 6 | $5.018 \times 10^{-5}$ | 0.022 |
| 7 | $4.818 \times 10^{-7}$ | 0.002 |
| 8 | $1.762 \times 10^{-9}$ | $1.502 \times 10^{-4}$ |
| 9 | $2.418 \times 10^{-12}$ | $5.281 \times 10^{-6}$ |
| 10 | $1.228 \times 10^{-15}$ | $1.031 \times 10^{-7}$ |

Table 2.2: Naive Estimation of Transitivity

equally likely (although it is quickly apparent that this assumption is grossly inaccurate). In that case, we might estimate the probability of transitivity by

$$P(T|n) = \frac{n!}{3^{\binom{n}{2}}} \text{ or } P(T|n) = \frac{n!}{2^{\binom{n}{2}}},$$

when $m$ is even and when $m$ is odd, respectively. Table 2.2 contains exact values for $n = 3, \ldots, 10$ and Figure 2.4 plots the probability functions. Note that the function has no interpretation at non-integer values; evaluations at integer values have been connected so that the two functions can be easily distinguished.

The naive approach to calculating offers significant insight into the question of transitivity. The probability of obtaining a transitive simple majority rule societal preference outcome goes to zero very quickly, which implies that if the true probability of transitivity likely also goes to zero unless transitive outcomes must be substantially more probable than intransitive outcomes.

Attempts to compute the distribution of simple majority rule societal preference outcomes analytically proved extraordinarily complex; indeed, we have no analytic solution to present. In the absence of an analytic solution, an empirical analysis was undertaken. The next section presents and discusses our empirical results.

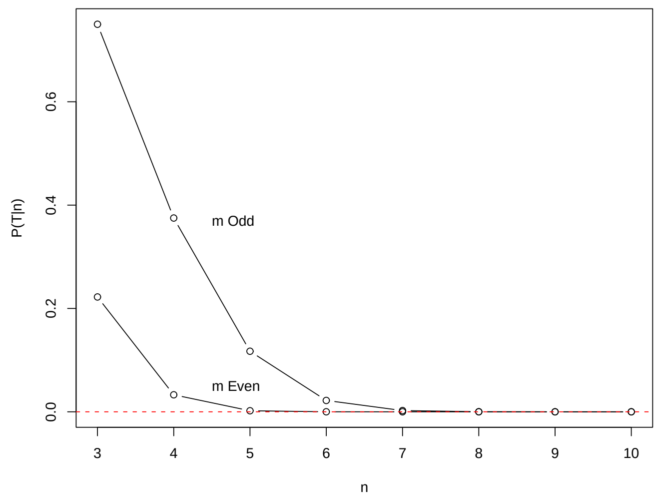


Figure 2.4: Naive Estimation of Transitivity

## 2.6 Empirical Probability of Transitivity

The distribution of societal preference outcomes arising from simple majority rule appears to be an intractable combinatoric problem. Nevertheless, the formulation of the problem is sufficiently simple that it may be calculated explicitly via exhaustive computation. For the sake of computational simplicity, the following assumptions were imposed:

- There is a fixed labeling of voters and alternatives.
- All voter preference orderings are strict linear orders on the set of alternatives.
- There is an odd number of voters, so that the societal preference outcome is a complete, asymmetric relation.

Given these assumptions, the algorithm is formulated as follows.

**Algorithm 2.6.1.** *For each profile $p$ in the profile space,*

1. *Compute the simple majority rule societal preference outcome, $g(p)$.*

| $m$ | $n = 3$ | $n = 4$ | $n = 5$ | $n = 6$ | $n = 7$ |
|---|---|---|---|---|---|
| 3 | 0.9444 | 0.8299 | 0.6757 | 0.5099 | 0.3573 |
| 5 | 0.9306 | 0.7898 | – | – | – |

Table 2.3: Computed Probabilities of Transitivity

*2. Increment the frequency for the observed outcome.*

The algorithm was implemented serially for three and five voters in MATLAB and FORTRAN, and in parallel in FORTRAN. With three voters, the program was run with three through seven alternatives; with five voters, only three and four alternatives were considered due to excessive runtimes.

In addition to calculating the frequency of each simple majority rule societal preference outcome, the transitivity score of each outcome was computed. Appendices A-G contain the computed frequencies and transitivity scores for the computed cases. Appendices H-I contain the serial and parallel FORTRAN code, respectively.

The computations provided several interesting results, both expected and unexpected. First, we found that for the cases considered, the probability of transitivity decreased in both the number of voters and the number of alternatives. This result was generally expected, as increased voters allow for more complex cycle structure in the outcome, and increased alternatives provide increased opportunities for cycles to form. The rate at which the probability went to zero was somewhat unexpected; it suggests that the true probability may in fact go to zero.

Other surprising features arose from the data. With three voters and three, four, or five alternatives, we found that the frequencies were monotonic in transitivity scores, although the frequencies provided a finer partition of the societal preference outcomes. These observations seemed to support the practical implementation of majority rule, as the most frequent outcomes were also the most transitive. However when three voters and six alternatives were considered, this monotonicity broke down.

Our analysis has been unable to discover the reason for this change, although we suspect it might be related to an open question

| $n$ | $F(T)$ | $F(IT_{min})$ | $F(IT_{avg})$ |
|---|---|---|---|
| 3 | 34 | 6 | 6.00 |
| 4 | 478 | 30 | 58.80 |
| 5 | 9730 | 30 | 619.91 |
| 6 | 264334 | 54 | 5707.92 |
| 7 | 9076864 | 42 | 38819.86 |

Table 2.4: Outcome Frequencies for Three Voters

in graph theory: the number of distinct unlabeled complete asymmetric digraphs with six or more nodes. An unlabeled digraph with five or fewer nodes can be completely characterized by the in-degree and out-degree of each node; this is not the case for graphs with six or more nodes. Unfortunately, transitivity scores could not be computed for the case of three voters and seven alternatives, again due to excessive runtimes. We therefore do not have subsequent data to confirm this break down in monotonicity for greater numbers of alternatives.

Another characteristic of the data is that the transitive outcomes were the most frequent in all cases considered; indeed, they were three to four times as frequent as the most frequent intransitive outcome. This pattern offers hope that even if the overall probability of transitivity goes to zero, the most probable outcomes may indeed be transitive outcomes.

These empirical exercises have generated a huge amount of data; indeed, more than could be sufficiently analyzed in the course of this project. We hope that the data might prove useful to future researchers in the area, and may be used at some point to validate or test hypotheses concerning the analytic distribution. Based on our analysis of these data, we put forth two conjectures.

**Conjecture 2.6.2.** *Under simple majority rule, an arbitrary transitive societal preference outcome is more probable than the average intransitive societal preference outcome.*

Conjecture 2.6.2 is strongly supported by the data. Figure 2.5 is a plot of the ratioratio

$$\frac{F(T)}{F(IT_{avg})}$$

for three voters and three through seven alternatives, where $F(T)$ is the frequency of an arbitrary transitive societal preference outcome, $F(IT_{min})$ is the frequency of the least frequent societal preference outcome, and $F(IT_{avg})$ is the average frequency of all the intransitive societal preference outcomes.

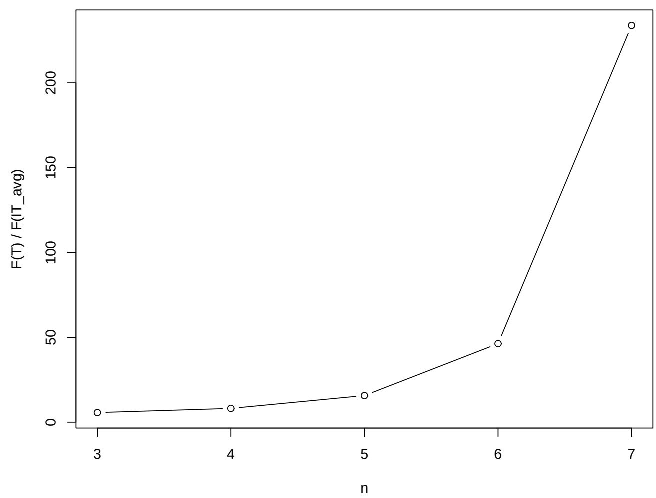


Figure 2.5: Ratio of Frequency of Transitive to Average Frequency of Intransitive Outcome

**Conjecture 2.6.3.** *Under simple majority rule, an arbitrary transitive societal preference outcome is more probable than an arbitrary intransitive societal preference outcome.*

Future research may provide the tools necessary to prove or disprove these conjectures. The available empirical evidence supports both, but an analytic proof has remained elusive.

# Chapter 3

# Borda Rules

## 3.1 Generalized Borda Rules

Qualitatively, generalized Borda rules produce a societal preference outcome via the following procedure:

1. Each voter is given $n$ tokens, where each token has a pre-defined value.
2. Each voter places the highest value token in the urn of their most-preferred alternative, the second-highest value token in the urn for their second-most-preferred alternative, etc.
3. After all the voters have distributed their tokens, the total value of the tokens in each urn is computed.
4. The alternatives are ranked in descending order according to the total value in their urns.

With this qualitative understanding in mind, we proceed with a formal definition.

Let $V = \{1, \ldots, m\}$ be the set of voters, and $A = \{a_1, \ldots, a_n\}$ be the set of alternatives. Define the ranking function $r : A \times L^A \longrightarrow \{1, \ldots, n\}$ by

$$r(a_i, p(j)) = k \quad \Leftrightarrow \quad p_k(j) = a_i.$$

When evaluated at $(a_i, p(j))$, the ranking function therefore gives the rank of alternative $a_i$ in the voter preference ordering $p(j)$, where $p$ is some profile and $j \in V$. Next define a real-valued scoring function $s(k)$ for $k = 1, \ldots, n$ such that

$$k_1 < k_2 \quad \Rightarrow \quad s(k_1) > s(k_2).$$

In our earlier qualitative description, the scoring function assigns the value to each of the $n$ tokens.

Define a vector $\Sigma(p) \in \mathbb{R}^n$ by

$$\Sigma(p)_i = \sum_{j=1}^{m} s(r(a_i, p(j))).$$

Again in reference to our qualitative description, $\Sigma(p)_i$ corresponds to the total value of the tokens in the urn associated with alternative $a_i$. We may therefore compute the societal preference ordering of a generalized Borda Rule at a profile $p$ by

- $a_i \succ_{g_{B_s}(p)} a_j$ if and only if $\Sigma(p)_i > \Sigma(p)_j$.
- $a_i \sim_{g_{B_s}(p)} a_j$ if and only if $\Sigma(p)_i = \Sigma(p)_j$.

**Lemma 3.1.1.** *Generalized Borda rules are unanimous, anonymous, neutral, and consistent.*

*Proof.* Let $V = \{1, \ldots, m\}$ be the set of voters, $A = \{a_1, \ldots, a_n\}$ be the set of alternatives, and $s$ be a scoring function for a generalized Borda rule $g$.

Let $p$ be a profile at which there exists $a_i \in A$ such that $a_i \in p_1(j)$ for all $j \in V$. Then $r(a_i, p(j)) = 1$ for all $j \in V$ so that $s(r(a_i, p(j))) \geq s(r(a_k, p(j)))$ for all $a_k \in A \setminus \{a_i\}$ and all $j \in V$, by monotonicity of $s$. Therefore,

$$\Sigma(p)_i = \sum_{j=1}^{m} s(r(a_i, p(j))) \geq \sum_{j=1}^{m} s(r(a_k, p(j))) = \Sigma(p)_k,$$

for all $a_k \in A \setminus \{a_i\}$. By the definition of $g$, we therefore have $a_i \in g_1(p)$ so $g$ is unanimous.

Let $\sigma_V$ be a permutation of the voters. Then for all $i \in \{1, \ldots, n\}$,

$$\Sigma(\sigma(p))_i = \sum_{j=1}^{m} s(r(a_i, p(\sigma_V(j)))) = \sum_{j=1}^{m} s(r(a_i, p(j))) = \Sigma(p)_i,$$

since addition is commutative. Therefore, $g$ is anonymous.

Next let $\sigma_A$ be a permutation of the alternatives, with $\hat{\sigma}_A$ the corresponding permutation on the indices $\{1, \ldots, n\}$. Then

$$\Sigma(\sigma_A(p))_i = \sum_{j=1}^{m} s(r(\sigma(a_i), p(j))) = \sum_{j=1}^{m} s(r(a_{\hat{\sigma}_A(i)}, p(j))) = \Sigma(p)_{\hat{\sigma}_A(i)},$$

so $g$ is neutral.

Let $V_1$ ($|V_1| = m_1$) and $V_2$ ($|V_2| = m_2$) be two disjoint sets of voters. Let profile $p_1$ contain preference orderings for voters in $V_1$, and profile $p_2$ contain preference orderings for voters in $V_2$. Let $p$ be the profile on $V = V_1 \cup C_2$ such that $p$ coincides with $p_1$ on $V_1$ and with $p_2$ on $V_2$. Then

$$\Sigma(p)_i = \sum_{j \in V} s(r(a_i, p(j))) = \sum_{j \in V_1} s(r(a_i, p(j))) + \sum_{j \in V_2} s(r(a_i, p(j))) = \Sigma(p_1)_i + \Sigma(p_2)_i.$$

If

$$\Sigma(p_1)_i = \Sigma(p_2)_i > \Sigma(p_2)_j = \Sigma(p_1)_j$$

for some $i, j$, then

$$\Sigma(p)_i = \Sigma(p_1)_i + \Sigma(p_2)_i > \Sigma(p_1)_j + \Sigma(p_2)_j = \Sigma(p)_j,$$

so $g$ is consistent. □

As with the generalized majority rule, generalized Borda rules must fail to satisfy one of the conditions of Arrow's Impossibility Theorem. We next state this failure here as a proposition; we will prove it by means of example in the case of the standard Borda rule in the next section.

**Proposition 3.1.2.** *Generalized Borda rules do not satisfy inde-*

*pendence of irrelevant alternatives.*

## 3.2 Standard Borda Rule

Define $g_B$ to be the standard Borda rule (Borda's rule), for which $s(k) = n - k + 1$. As a generalized Borda rule, Borda's rule displays all of the properties of a generalized Borda rule. In the previous section we stated, but did not prove, that generalized Borda rules do not satisfy independence of irrelevant alternatives. Next we consider an example of this failure.

**Example 3.2.1.** *Let $V = \{1, 2, 3, 4\}$ be the set of voters, $A = \{a_1, a_2, a_3\}$ be a set of alternatives, and $A^+ = \{a_1, a_2, a_3, a_4\}$ be another set of alternatives. Consider the following profile $p$ on $A$,*

$$p = \begin{bmatrix} \underline{1} & \underline{2} & \underline{3} & \underline{4} \\ a_1 & a_2 & a_3 & a_2 \\ a_3 & a_1 & a_2 & a_3 \\ a_2 & a_3 & a_1 & a_1 \end{bmatrix}.$$

*At this profile $\Sigma(p) = (7, 9, 8)$ so that $g_B(p) = (a_2 \succ a_3 \succ a_1)$. Next consider the profile $p^+$ on $A^+$*

$$p^+ = \begin{bmatrix} \underline{1} & \underline{2} & \underline{3} & \underline{4} \\ a_1 & a_2 & a_3 & a_2 \\ a_3 & a_1 & a_4 & a_3 \\ a_4 & a_3 & a_2 & a_1 \\ a_2 & a_4 & a_1 & a_4 \end{bmatrix},$$

*which is formed from profile $p$ by inserting alternative $a_4$ into each voter's preference ordering. At this profile, we have $\Sigma(p^+) = (10, 11, 12, 7)$ so that*

$$g_B(p^+) = (a_3 \succ a_2 \succ a_1 \succ a_4).$$

*Therefore, we have $a_2 \succ_{g_B(p)} a_3$ and $a_3 \succ g_B(p^+)a_2$, so Borda's rule does not satisfy independence of irrelevant alternatives.*

Borda's rule is also susceptible to manipulation, as the following example illustrates.

**Example 3.2.2.** *Let* $V = \{1, 2, 3\}$ *and* $A = \{a_1, a_2, a_3, a_4\}$. *Consider the profile*

$$p = \begin{bmatrix} \underline{1} & \underline{2} & \underline{3} \\ a_2 & a_1 & a_3 \\ a_1 & a_3 & a_4 \\ a_3 & a_4 & a_2 \\ a_4 & a_2 & a_1 \end{bmatrix},$$

*at which* $\Sigma(p) = (8, 7, 9, 6)$ *so that* $g_B(p) = (a_3 \succ a_1 \succ a_2 \succ a_4)$. *Next, consider the profile*

$$q = \begin{bmatrix} \underline{1} & \underline{2} & \underline{3} \\ a_1 & a_1 & a_3 \\ a_2 & a_3 & a_4 \\ a_4 & a_4 & a_2 \\ a_3 & a_2 & a_1 \end{bmatrix},$$

*at which* $q(i) = p(i)$ *for* $i = 2, 3$ *and* $\Sigma(q) = (9, 6, 8, 7)$ *so that* $g_B(q) = (a_1 \succ a_3 \succ a_4 \succ a_2)$.

*Note that alternative* $a_3$ *is first-ranked in* $g_B(q)$ *and alternative* $a_1$ *is first-ranked in* $g_B(p)$, *and that* $a_1 \succ_{p(i)} a_3$. *Therefore, individual 1 will manipulate at profile* $p$ *via the preference ordering* $q(1)$ *to ensure the promotion of alternative* $a_1$ *to the top of the societal preference outcome.*

Whereas failure to satisfy independence of irrelevant alternatives is suggested by Arrow's theorem, the manipulability of Borda's rule is suggested by the Gibbard–Satterthwaite theorem. Although both are considered undesirable properties of a social welfare rule, susceptibility to strategic manipulation is often considered a more grievous failure. We will therefore establish the framework for the quantification of manipulability of Borda's rule.

### 3.2.1 Measure of Manipulation

We wish to quantify the extent to which Borda's rule is susceptible to manipulation as a function of the number of voters and the number of alternatives. We will therefore establish a measure of local manipulability; that is, to what extent can a voter manipulate at a specific profile. To form an overall measure of manipulability we will average the local measures over the profile space, assuming that each profile is equally likely.

Given a profile $p$ on $m$ voters, our local measure of manipulability computes the proportion of alternatives not top-ranked in the societal preference outcome that can be promoted to top-ranked by a voter entering the society. A formal definition of this measure follows.

Let $V_1 = \{1, \ldots, m-1\}$ be set of voters, $V_2 = \{m\}$ be a singleton set consisting of one voter, and $A = \{a_1, \ldots, a_n\}$ be the set of alternatives. If $p$ is some profile containing the preference orderings of the voters in $V_1$ and $q(m)$ is a preference ordering of individual $m$, define $p_q$ to a profile such that $p_q(i) = p(i)$ for $i = 1, \ldots, m-1$ and $p_q(m) = q(m)$.

Next define the set $\Lambda(p)$ by

$$\Lambda(p) = \{a_i \ : \ \exists q(m) \ s.t. \ a_i \in g_1(p_q) \text{ and } a_i \notin g_1(p)\}.$$

That is, $\Lambda(p)$ is the set of alternatives that were not top-ranked in the societal preference outcome at $g(p)$, but could be made top-ranked in the societal preference outcome by voter $m$ contributing their preference ordering. Intuitively, this set corresponds to our notion of manipulability since voter $m$ can freely promote any of the alternatives in $\Lambda(p)$ to the top of the social preference outcome.

Next define the function $\lambda(p)$ by

$$\lambda(p) = \frac{|\Lambda(p)|}{|A \setminus g_1(p)|}.$$

As previously alluded to, $\lambda(p)$ is the proportion of non-top-ranked alternatives that can be promoted to the top of the societal preference ordering by voter $m$. By construction, $0 \leq \lambda(p) < 1$ for all profiles $p$.

There are many potential measure of manipulability, although there are none in the literature with which we are familiar. The appeal of the above measure is its congruence with our formal and intuitive concept of manipulation and the ease with which it can be computed. In the next section we take advantage of the computational ease of this measure.

## 3.3 Computation of Manipulability

As was the case with simple majority rule, analysis of the entire profile space was limited to a small number cases due to excessive runtimes; for $n$ voters and $m$ alternatives there are $n!^m$ profiles in the profile space $L_S(A)$. Analysis of the entire profile space was nevertheless conducted for two through four voters and three through seven alternatives. Analysis of the entire profile space utilized the following algorithm.

**Algorithm 3.3.1** (Exact Manipulation Score)**.** *Let $V = \{1, \ldots, m\}$ be the set of voters and $A = \{a_1, \ldots, a_n\}$ be the set of alternatives. For every profile $p \in L_S(A)^{(m-1)}$ on $m - 1$ voters,*

1. *Store the set $g_1(p)$.*
2. *Set $\Lambda(p) := \emptyset$.*
3. *For of the $n!$ each preference orderings $q(m)$ on $A$,*
    (a) *Construct $p_q$ by appending the preference ordering $q(m)$ to the profile $p$.*
    (b) *Determine the set $g_1(p_q)$.*
    (c) *For $i = 1, \ldots, n$, if $a_i \in g_1(p_q)$ and $a_i \notin g_1(p)$, set $\Lambda(p) := \Lambda(p) \cup \{a_i\}$.*
4. *Compute $\lambda(p) = \dfrac{|\Lambda(p)|}{|A \setminus g_1(p)|}$.*

| | $n=3$ | $n=4$ | $n=5$ | $n=6$ | $n=7$ |
|---|---|---|---|---|---|
| $m=2$ | 0.517 | 0.539 | 0.574 | 0.595 | 0.603 |
| $m=3$ | 0.454 | 0.482 | 0.488 | 0.494 | 0.500 |
| $m=4$ | 0.401 | 0.415 | 0.426 | 0.436 | 0.434 |

Table 3.1: Exact Manipulation Score for Borda's Rule

*To compute the average manipulation score, let*

$$\bar{\lambda} = \frac{1}{n!^{(m-1)}} \sum_{p \in L_S(A)^{(m-1)}} \lambda(p).$$

The MATLAB implementation of this algorithm is available in Appendix M, and the results of the algorithm for two through four voters and three through seven alternatives are available in Table 3.1. Two distinct trends emerge from the analysis of the entire profile space. First, we note that as the number of alternatives increase, Borda's rule is increasingly susceptible to manipulation by our measure. Moreover, since the manipulability score measures the proportion of non-top-ranking alternatives that can be promoted to the top of the social preference outcome, the score is not merely being inflated due to the large total number of alternatives.

Second, we note that as the number of voters increases, Borda's rule becomes less susceptible to manipulation by our measure. In a real voting setting we might attribute this decline to increased uncertainty on the part of the $m$th voter concerning the preference of the previous $m-1$ voters. However, this metric of manipulability assumes that the $m$th voter has complete knowledge of the consequence of reporting every of the $n!$ preference orderings at his or her disposal. Rather, the decrease in manipulability appears to reflect the decrease in the ratio of the number of points the $m$th voter can contribute to an alternative relative to the number of points the previous $m-1$ voters have already distributed.

When the size of the profile space grew too large to allow for exact computation of the average manipulation score, average scores were approximated by randomly sampling in the profile space. This was conducted for five through ten voters, and three through seven alternatives. Sampling utilized the following algorithm.

**Algorithm 3.3.2** (Approximation of Manipulation Score)**.** *Let* $V = \{1, \ldots, m\}$ *be the set of voters,* $A = \{a_1, \ldots, a_n\}$ *be the set of alternatives, and* $s$ *be the sample size. For* $i = 1, \ldots, s$*,*

1. *Randomly generate a profile* $p \in L_S(A)^{(m-1)}$ *on* $m - 1$ *voters.*
2. *Store the set* $g_1(p)$*.*
3. *Set* $\Lambda(i) := \emptyset$*.*
4. *For of the* $n!$ *each preference orderings* $q(m)$ *on* $A$*,*
   (a) *Construct* $p_q$ *by appending the preference ordering* $q(m)$ *to the profile* $p$*.*
   (b) *Determine the set* $g_1(p_q)$*.*
   (c) *For* $j = 1, \ldots, n$*, if* $a_j \in g_1(p_q)$ *and* $a_j \notin g_1(p)$*, set* $\Lambda(i) := \Lambda(i) \cup \{a_j\}$*.*
5. *Compute* $\lambda(i) = \dfrac{|\Lambda(i)|}{|A \setminus g_1(p)|}$*.*

*To compute the approximate average manipulation score, let*

$$\hat{\bar{\lambda}} = \frac{1}{S} \sum_{i=1}^{S} \lambda(i).$$

The MATLAB implementation of this algorithm is available in Appendix L. As sampling techniques are inherently less reliable than exhaustive techniques, we proceed with brief discussion of the sampling process.

Sampling was conducted without replacement. In the smallest case, the profile space consistent of over 7,000 profiles, so the probability of randomly sampling the sample profile more than once was small. Furthermore, the storage capacity needed to ensure that profiles were re-sampled would have significantly reduced this number of cases we were able to consider.

Profiles were randomly sampling by indexing the $n!$ preference orderings voters were permitted to report and randomly assigning

|  | $n=3$ | $n=4$ | $n=5$ | $n=6$ | $n=7$ |
|---|---|---|---|---|---|
| $m=5$ | 0.363 | 0.375 | 0.381 | 0.383 | 0.386 |
| $m=6$ | 0.342 | 0.346 | 0.346 | 0.349 | 0.349 |
| $m=7$ | 0.317 | 0.329 | 0.328 | 0.332 | 0.333 |
| $m=8$ | 0.294 | 0.305 | 0.308 | 0.309 | 0.309 |
| $m=9$ | 0.278 | 0.286 | 0.285 | 0.287 | 0.289 |
| $m=10$ | 0.263 | 0.271 | 0.273 | 0.272 | 0.276 |

Table 3.2: Simulated Manipulation Score for Borda's Rule

a preference ordering to each voter using the MATLAB random number generator with the seed set by the system clock. For each specification of $m$ voters and $n$ alternatives, a sample size of 15% of the profile space was selected ($s = 0.15 \times n!^{(m-1)}$). The threshold of 15% was selected through trial and error as the minimum proportion of the profile space that consistently produced approximations of the average manipulation score that were reliable to two decimal places.

Using this sampling procedure, ten replications of the program were run for each $(m, n)$ pair. The approximate averages from each of the ten replications were then averaged. The results are available in Table 3.2.

Although there the patterns are slightly less consistent in the approximated manipulation scores relative to the exact scores, the same patterns are apparent. Approximated manipulation scores for Borda's rule increase as the number of alternatives are increased, but decrease as the number of voters is decreased.

The consistent and intuitive pattern that arose from computation of manipulation scores for the standard Borda rule motivated an analysis of manipulability for other generalized Borda rules. In particular, one framework was chosen on which to focus additional analysis.

Let $s_1$ denote the scoring function of the standard Borda rule, and let $s_2$ denote an exponential scoring function, where $s_2(k) = e^{n-k+1}$, where $n$ is the number of alternatives.

**Example 3.3.3.** *Let $V = \{1, 2\}$ be the set of voters, $A = \{a_1, a_2, a_3\}$ denote the set of alternatives, and $s_2$ be the exponential scoring func-*

*tion. Then at the profile*

$$p = \begin{bmatrix} \underline{1} & \underline{2} \\ a_2 & a_1 \\ a_1 & a_3 \\ a_3 & a_2 \end{bmatrix},$$

*we have*

$$\Sigma(p) = \begin{pmatrix} e^2 + e^3 \\ e^3 + e^1 \\ e^1 + e^2 \end{pmatrix} \approx \begin{pmatrix} 27.47 \\ 22.80 \\ 10.11 \end{pmatrix}.$$

Using the standard and exponential scoring functions, we implemented a program that sampled randomly from the profile space, at each profile calculated the manipulation score of a number of Borda rules with weights that were linear combinations of $s_1$ and $s_2$. The program utilized the following algorithm.

**Algorithm 3.3.4.** *Let $V = \{1, \ldots, m\}$ be the set of voters, $A = \{a_1, \ldots, a_n\}$ be the set of alternatives, $S$ be the sample size, and $\delta \in (0, 1)$ be a step-size parameter that divides 1. For $i = 1, \ldots, S$,*

1. *Randomly generate a profile $p \in L_S(A)^{(m-1)}$ on $m - 1$ voters.*
2. *Store the set $g_1(p)$.*
3. *For of the $n!$ each preference orderings $q(m)$ on $A$:*
   - (a) *Construct $p_q$ by appending the preference ordering $q(m)$ to the profile $p$.*
   - (b) *For $d = 0, \delta, 2\delta, \ldots, 1$,*
     - i. *Set $\Lambda(i, d) = \emptyset$.*
     - ii. *Determine the set $g_1(p_q)$, where $g$ has scoring function $s = ds_1 + (1 - d)s_2$.*
     - iii. *For $j = 1, \ldots, n$, if $a_j \in g_1(p_q)$ and $a_j \notin g_1(p)$, set $\Lambda(i, d) := \Lambda(i, d) \cup \{a_j\}$.*
     - iv. *Compute $\lambda(i, d) = \dfrac{|\Lambda(p)|}{|A \setminus g_1(p)|}$.*

*To compute the approximate average manipulation score for the Borda rule with scoring function $s = ds_1 + (1-d)s_2$, let*

$$\hat{\lambda}_d = \frac{1}{S}\sum_{i=1}^{S} \lambda(i, d).$$

The sampling technique employed in this algorithm was identical to that of Algorithm 3.3.2. To isolate differences in manipulation scores from sampling variation, manipulation scores were computed at the same profiles for all Borda rules indexed by $d$.

Unlike the previous results, the approximated manipulation scores for the Borda rules with convex standard–exponential scoring functions displayed an unexpected result. Prior to running the simulation, we hypothesized that manipulation scores would be monotonic in $d$; that is, one of standard weights or exponential weights would have a higher average manipulation score. Instead we found an unexpected oscillatory behavior in the manipulation scores, depicted in Figure 3.1 and Figure 3.2.

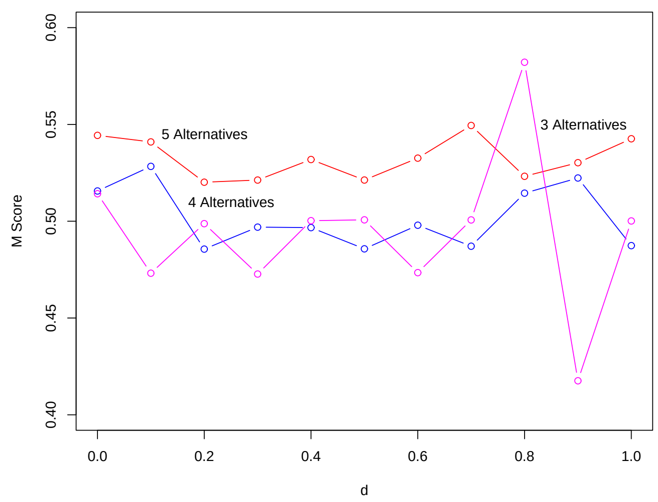


Figure 3.1: Convex Borda Manipulability Scores, 2 Voters

These results suggest the need for additional study of the impact

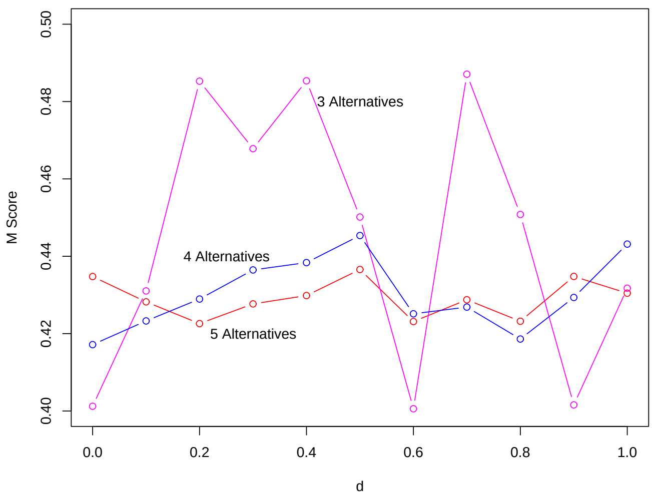


Figure 3.2: Convex Borda Manipulability Scores, 3 Voters

of Borda weights on manipulability. To date we have not found a satisfactory theoretical explanation for this behavior.

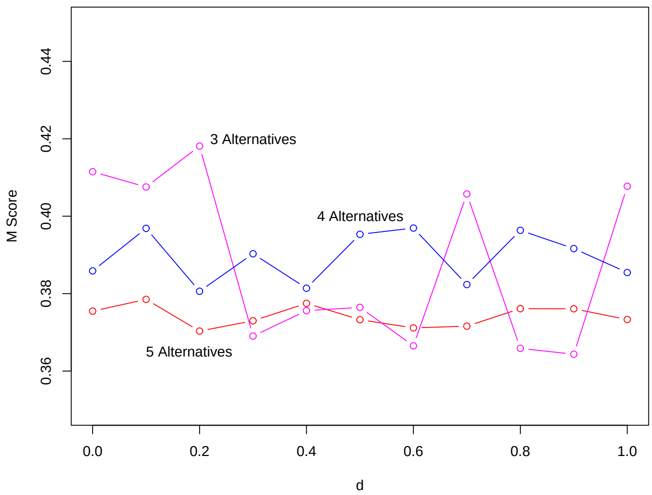


Figure 3.3: Convex Borda Manipulability Scores, 4 Voters

# Chapter 4

# A Matricial Approach to Social Choice Theory

Let $V = \{1, \ldots, m\}$ be the set of voters and $A = \{a_1, \ldots, a_n\}$ be the set of alternatives. Given a voter $k$'s preference ordering $p(k)$ on $A$, we can encode $p(k)$ into the $n \times n$ matrix $P(k)$ by the following procedure:

**Algorithm 4.0.5.** *Let $p(k)$ be a preference relation on $A$. Then define the matrix representation $P(k)$ by*

1. *If $a_i \succ_{p(k)} a_j$, let $P(k)_{ij} = 1$,*
2. *If $a_i \sim_{p(k)} a_j$, let $P(k)_{ij} = 0$,*

Given a profile $p$, we define the matrix representation of $p$ by

$$P = \sum_{k=1}^{m} P(k).$$

By construction, the elements of the matrix $P$ satisfy the following properties:

- $P_{ij} \geq 0$ for all $i, j \in \{1, \ldots, n\}$,
- $P_{ii} = 0$ for all $i \in \{1, \ldots, n\}$,

- $P_{ij} + P_{ji} = m$ for all $i, j \in \{1, \ldots, n\}$ with $i \neq j$.

We next consider an explicit example in terms of the matrix representation.

**Example 4.0.6** (The Matricial French Triple)**.** *Let* $V = \{1, 2, 3\}$ *and* $A = \{a_1, a_2, a_3\}$ *and consider the profile*

$$p = \begin{bmatrix} \underline{1} & \underline{2} & \underline{3} \\ a_1 & a_3 & a_2 \\ a_2 & a_1 & a_3 \\ a_3 & a_2 & a_1 \end{bmatrix}.$$

*Using the matrix representations, we have*

$$P(1) = \begin{bmatrix} 0 & 1 & 1 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix}, \qquad P(2) = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 0 \\ 1 & 1 & 0 \end{bmatrix}, \qquad P(3) = \begin{bmatrix} 0 & 0 & 0 \\ 1 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix},$$

*and therefore*

$$P = \begin{bmatrix} 0 & 2 & 1 \\ 1 & 0 & 1 \\ 2 & 1 & 0 \end{bmatrix}.$$

Many social choice rules can be formulated matricially, and in many cases the analysis of rules is simplified by this formulation. We will next revisit generalized majority and Borda rules from a matricial perspective, as well as introduce a new rule which can only be formulation in terms of voting matrices.

## 4.1 Simple Majority Rule

By Algorithm 1.3.1, we can represent each voter's preferences as a profile $p$ by a $\pm 1/0$-vector, $v(k)$. We can therefore define simple majority rule by

$$g_M(v) = \nu\left(\sum_{i=1}^{m} v(i)\right),$$

where

$$\nu(\alpha) = \begin{cases} 1 & \text{if } 0 < \alpha \\ 0 & \text{if } 0 = \alpha \\ -1 & \text{if } 0 > \alpha. \end{cases}$$

is a normalizing function.

We consider a concrete example using the matricial formulation of simple majority rule.

**Example 4.1.1.** *Let $V = \{1, 2\}$ be the set of voters and $A = \{a_1, a_2, a_3, a_4\}$ be the set of alternatives. Let $v$ be a profile such that*

$$v(1) = \begin{pmatrix} 1 \\ 1 \\ 1 \\ -1 \\ -1 \\ 1 \end{pmatrix}, \quad v(2) = \begin{pmatrix} -1 \\ 1 \\ -1 \\ 1 \\ -1 \\ -1 \end{pmatrix}$$

*That is,*

$$v^T = \begin{pmatrix} 0 & 2 & 0 & 0 & -2 & 0 \end{pmatrix},$$

*so that*

$$g_M(v)^T = \begin{pmatrix} 0 & 1 & 0 & 0 & -1 & 0 \end{pmatrix}.$$

*This vector profile corresponds to a profile $p$ where*

$$p = \begin{bmatrix} \underline{1} & \underline{2} \\ a_1 & a_4 \\ a_3 & a_2 \\ a_4 & a_1 \\ a_2 & a_3 \end{bmatrix}$$

*has the societal preference outcome depicted in Figure 4.1.*

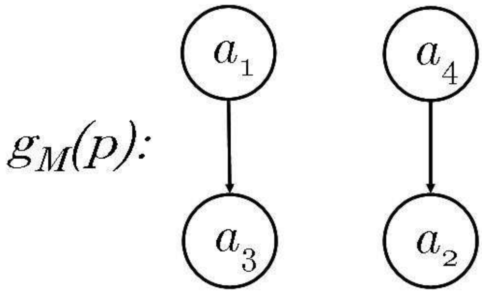


Figure 4.1: Societal Preference Outcome at Profile $p$

## 4.2 Standard Borda Rule

By Algorithm 1.3.1, we can represent each voter's preferences as a profile $p$ by a $\pm 1/0$-vector, $v(k)$. We can therefore define Borda's rule by

$$g_B(v) = r\left(M \times \sum_{i=1}^{m} v(i)\right),$$

where $r(s)$ is a ranking function that ranks the alternatives $a_i$ according to the value of $s_i$ and $M$ is an $n \times \binom{n}{2}$ matrix defined by the following algorithm.

**Algorithm 4.2.1.** *Let $M$ be an $n \times \binom{n}{2}$ matrix.*

*Set $M_{ij} := 0$ for all $i, j$.*

*Set $c := 1$.*

*For $j := 1, \ldots, n-1$,*

*For $i := j+1, \ldots, n$,*
*Set $M_{i,c} := 1$.*
*Set $M_{j,c} := -1$.*
*Set $c := c + 1$.*

Again, we consider a concrete example.

**Example 4.2.2.** *Let $V = \{1, 2, 3\}$ and $A = \{a_1, a_2, a_3\}$. Consider* *a profile p such that*

$$p = \begin{bmatrix} \underline{1} & \underline{2} & \underline{3} \\ a_1 & a_2 & a_1 \\ a_2 & a_1 & a_3 \\ a_3 & a_3 & a_2 \end{bmatrix},$$

*so that p has vector representation*

$$v(1) = \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix}, \quad v(2) = \begin{pmatrix} -1 \\ 1 \\ 1 \end{pmatrix}, \quad v(2) = \begin{pmatrix} 1 \\ 1 \\ -1 \end{pmatrix},$$

$$v^T = \begin{pmatrix} 1 & 3 & 1 \end{pmatrix}.$$

*Using our algorithm we have*

$$M = \begin{bmatrix} 1 & 1 & 0 \\ -1 & 0 & 1 \\ 0 & -1 & -1 \end{bmatrix},$$

*so that the Borda's rule societal preference outcome is*

$$g_B(v) = M \times v = \begin{bmatrix} 1 & 1 & 0 \\ -1 & 0 & 1 \\ 0 & -1 & -1 \end{bmatrix} \times \begin{bmatrix} 1 \\ 3 \\ 1 \end{bmatrix} = \begin{bmatrix} 4 \\ -2 \\ -2 \end{bmatrix}.$$

*Ranking the alternatives $a_i$ by their score $g_B(v)_i$ gives the societal preference outcome*

$$(a_1 \succ a_2 \succ a_3).$$

## 4.3 The Perron Rule

Let $V = \{1, \ldots, m\}$ be the set of voters and $A = \{a_1, \ldots, a_n\}$ be the set of alternatives. Define the matrix $A = P + J$, where $P$ is the matrix representation of a profile and where $J$ is the $n \times n$ matrix such that $J_{ij} = 1$ for all $i, j$. Therefore, we have that $P$ is a square matrix with $P_{ij} > 0$ for all $i, j$ and we may apply Perron's Theorem.

**Theorem 4.3.1** (Perron's Theorem). *If $A \in M_n$ and $A > 0$, then*

- $\rho(A) > 0$*;*
- $\rho(A)$ *is an eigenvalue of* $A$*;*
- *There is an* $x \in \mathbb{C}^n$ *with* $x > 0$ *and* $Ax = \rho(A)x$*;*
- $\rho(A)$ *is an algebraically simple eigenvalue of* $A$*;*
- $\rho(A)$ *is the unique eigenvalue of maximum modulus,*

*where* $\rho(A)$ *is the spectral radius of* $A$*.*

Given a matrix $A$, we may therefore compute the Perron eigenvector, that is the vector $v_A$ satisfying the equation

$$Av_A = \rho(A)v_A.$$

We therefore define Perron's rule as

$$g_P(P) = r\left(v_{P+J}\right),$$

where $r(s)$ is a ranking function that ranks the alternatives $a_i$ according to the value of $s_i$.

Borda's Rule gives a first approximation of Perron's Rule.

### 4.3.1 Analysis of Perron Manipulability

We wish to analyze the manipulability of Perron's rule as we analyzed the manipulability of Borda's rule. Analytic analysis of manipulability is hindered by the complex calculations needed to compute the Perron societal preference outcome. It is not at all clear how minor variations in the $P(k)$ matrices induce changes in the societal preference outcome $g_P(P)$. The relationship in manipulability of the two rules is of particular interest because Borda's rule provides a first approximation of Perron's rule [8].

To investigate the question of manipulability, computer simulation was employed. The sampling algorithm was as follows:

| | $n=3$ | $n=4$ | $n=5$ | $n=6$ | $n=7$ |
|---|---|---|---|---|---|
| $m=2$ | 0.725 | 0.893 | 0.978 | 0.993 | 0.999 |
| $m=3$ | 0.730 | 0.873 | 0.977 | 0.997 | 0.999 |
| $m=4$ | 0.758 | 0.798 | 0.933 | 0.987 | 0.998 |
| $m=5$ | 0.569 | 0.795 | 0.925 | 0.978 | 0.997 |
| $m=6$ | 0.650 | 0.745 | 0.889 | 0.960 | 0.993 |
| $m=7$ | 0.533 | 0.720 | 0.871 | 0.947 | 0.989 |
| $m=8$ | 0.580 | 0.686 | 0.838 | 0.927 | 0.981 |
| $m=9$ | 0.510 | 0.660 | 0.823 | 0.914 | 0.974 |
| $m=10$ | 0.541 | 0.640 | 0.788 | 0.893 | 0.965 |

Table 4.1: Simulated Perron's Rule Manipulability Score

**Algorithm 4.3.2** (Manipulability Comparison)**.** *For a fixed number of voters (m) and alternatives (n),*

1. *Obtain a profile, p, by sampling randomly from the profile space,*
2. *Compute the profile matrix $P$ from p,*
3. *Compute the societal preference orderings $g_P(P)$ and $g_B(p)$,*
4. *Evaluate the manipulability of each of $g_P(P)$ and $g_B(p)$ using the previous algorithm.*

The MATLAB code for this algorithm is available in Appendix L. To isolate differences in manipulability from sampling variation, manipulability of the Perron and Borda rules are calculated with respect to the same preference profile in each iteration. The sampling was conducted using the same techniques as previously discussed.

Table 4.1 contains estimates for the Perron rule manipulability score for a large number of cases. In comparison to the Tables 3.1 and 3.2, Perron's rule appears to be significantly more manipulable. Although we cannot identify any result in matrix theory that would suggest this pattern, it the high manipulability scores were consistent across numerous simulations.

The following figures depict estimated manipulability scores for the Borda and Perron rules in a number of settings.

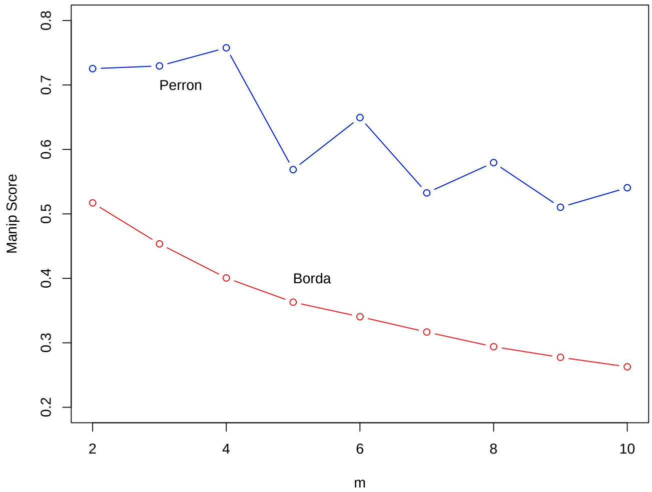


Figure 4.2: Borda vs. Perron Manipulability Scores, 3 Alternatives

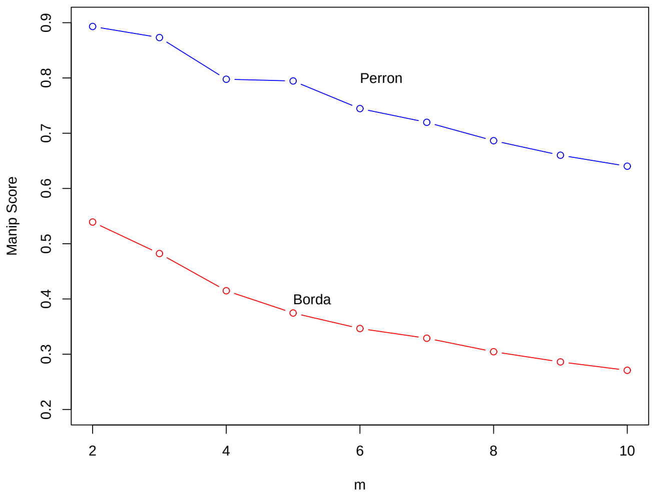


Figure 4.3: Borda vs. Perron Manipulability Scores, 4 Alternatives

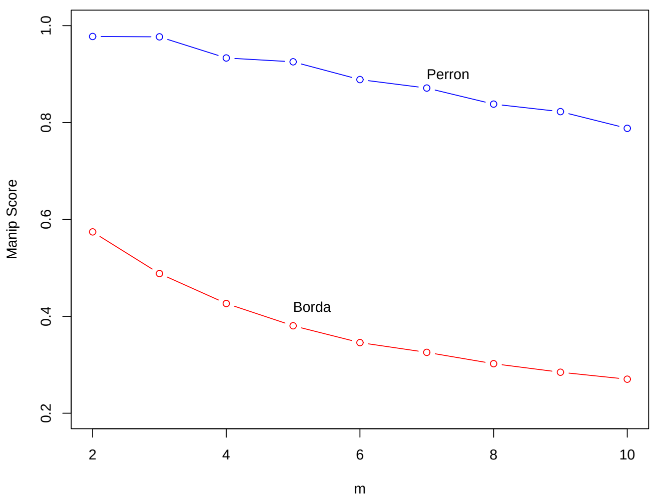


Figure 4.4: Borda vs. Perron Manipulability Scores, 5 Alternatives

# Chapter 5

# Appendices

## Appendix A: Inventory Analysis for 3 Voters, 3 Alternatives

| Frequency | Number of Distinct Labeled CADS | Transitivity Score |
|---|---|---|
| 6 | 2 | 0.3333 |
| 24 | 6 | 1 |

$$P[\text{Transitivity}] = \frac{24 \times 6}{3!^3} \approx 0.9444.$$

## Appendix B: Inventory Analysis for 3 Voters, 4 Alternatives

| Frequency | Number of Distinct Labeled CADS | Transitivity Score |
|---|---|---|
| 30 | 24 | 0.6667 |
| 102 | 16 | 0.6667 |
| 478 | 24 | 1 |

$$P[\text{Transitivity}] = \frac{728 \times 24}{4!^3} \approx 0.8299.$$

# Appendix C: Inventory Analysis for 3 Voters, 5 Alternatives

| Frequency | Number of Distinct Labeled CADS | Transitivity Score |
|---|---|---|
| 30 | 24 | 0.4 |
| 54 | 40 | 0.6 |
| 72 | 120 | 0.6 |
| 144 | 120 | 0.6 |
| 234 | 120 | 0.6 |
| 270 | 120 | 0.8 |
| 798 | 240 | 0.8 |
| 2286 | 80 | 0.8 |
| 2418 | 40 | 0.8 |
| 9730 | 120 | 1 |

$$P[\text{Transitivity}] = \frac{9730 \times 120}{5!^3} \approx 0.6757.$$

## Appendix D: Inventory Analysis for 3 Voters, 6 Alternatives

| Frequency | Number of Distinct Labeled CADS | Transitivity Score |
|---|---|---|
| 54 | 240 | 0.4667 |
| 60 | 1440 | 0.4667 |
| 132 | 720 | 0.6 |
| 156 | 1440 | 0.6 |
| 210 | 1440 | 0.6 |
| 252 | 480 | 0.6 |
| 282 | 720 | 0.7333 |
| 288 | 1440 | 0.6 |
| 324 | 1440 | 0.6 |
| 360 | 720 | 0.6 |
| 384 | 720 | 0.7333 |
| 534 | 1440 | 0.6 |
| 546 | 1440 | 0.6 |
| 672 | 240 | 0.4667 |
| 846 | 480 | 0.7333 |
| 996 | 1440 | 0.7333 |
| 1068 | 720 | 0.6 |
| 1230 | 288 | 0.6 |
| 1284 | 1440 | 0.7333 |
| 1734 | 1440 | 0.7333 |
| 2124 | 480 | 0.7333 |
| 2790 | 1440 | 0.7333 |
| 2814 | 720 | 0.7333 |
| 3000 | 1440 | 0.7333 |
| 3762 | 720 | 0.8667 |
| 5688 | 1440 | 0.7333 |
| 8472 | 1440 | 0.7333 |
| 10080 | 1440 | 0.8667 |
| 16272 | 80 | 0.7333 |
| 25050 | 1440 | 0.8667 |
| 27174 | 720 | 0.8667 |
| 65850 | 480 | 0.8667 |
| 70614 | 480 | 0.8667 |
| 264334 | 720 | 1 |

$$P[\text{Transitivity}] = \frac{264334 \times 720}{6!^3} \approx 0.5099.$$

# Appendix E: Inventory Analysis for 3 Voters, 7 Alternatives

| Frequency | Number of Distinct Labeled CADS | Frequency | Number of Distinct Labeled CADS |
|---|---|---|---|
| 42 | 960 | 558 | 15120 |
| 48 | 20160 | 576 | 20160 |
| 54 | 25200 | 612 | 10080 |
| 60 | 26880 | 618 | 10080 |
| 66 | 5040 | 630 | 10080 |
| 84 | 1680 | 636 | 5040 |
| 102 | 5040 | 642 | 5040 |
| 108 | 5040 | 648 | 5040 |
| 114 | 5040 | 654 | 20160 |
| 120 | 10080 | 660 | 15120 |
| 126 | 15120 | 696 | 10080 |
| 156 | 15120 | 702 | 11760 |
| 162 | 20160 | 708 | 10080 |
| 168 | 20160 | 714 | 10080 |
| 174 | 5040 | 720 | 10080 |
| 180 | 10080 | 774 | 10080 |
| 192 | 10080 | 810 | 10080 |
| 210 | 5040 | 834 | 10080 |
| 216 | 1680 | 876 | 10080 |
| 228 | 10080 | 918 | 5040 |
| 234 | 30240 | 936 | 30240 |
| 252 | 30240 | 960 | 15120 |
| 258 | 5040 | 1140 | 10080 |
| 264 | 20160 | 1146 | 20160 |
| 270 | 1008 | 1152 | 6720 |
| 282 | 10080 | 1158 | 10080 |
| 300 | 20160 | 1176 | 20160 |
| 306 | 10080 | 1206 | 10080 |
| 318 | 10080 | 1236 | 10080 |
| 324 | 20160 | 1272 | 10080 |
| 330 | 10080 | 1278 | 10080 |
| 360 | 10080 | 1296 | 1680 |
| 372 | 10080 | 1308 | 10080 |
| 378 | 5040 | 1320 | 10080 |
| 420 | 10080 | 1332 | 10080 |
| 426 | 10080 | 1350 | 3360 |
| 432 | 10640 | 1392 | 15120 |
| 450 | 10080 | 1440 | 5040 |
| 456 | 10080 | 1470 | 10080 |
| 468 | 10080 | 1518 | 10080 |
| 504 | 11760 | 1524 | 10080 |
| 540 | 3360 | 1590 | 5040 |
| 546 | 10080 | 1614 | 10080 |

| Frequency | Number of Distinct Labeled CADS | Frequency | Number of Distinct Labeled CADS |
|---|---|---|---|
| 1704 | 10080 | 4338 | 3360 |
| 1746 | 5040 | 4464 | 10080 |
| 1782 | 10080 | 5322 | 10080 |
| 1788 | 5040 | 5724 | 5040 |
| 1800 | 1680 | 5952 | 5040 |
| 1950 | 10080 | 6054 | 5040 |
| 1956 | 10080 | 6210 | 10080 |
| 1992 | 5040 | 6390 | 10080 |
| 2052 | 5040 | 6564 | 10080 |
| 2094 | 5040 | 6594 | 10080 |
| 2112 | 5040 | 6630 | 10080 |
| 2130 | 5040 | 6690 | 10080 |
| 2136 | 5040 | 6792 | 10080 |
| 2142 | 5040 | 6894 | 10080 |
| 2184 | 5040 | 7518 | 10080 |
| 2190 | 10080 | 7872 | 10080 |
| 2214 | 10080 | 8004 | 10080 |
| 2232 | 3360 | 8400 | 10080 |
| 2274 | 10080 | 8490 | 10080 |
| 2340 | 10080 | 8646 | 10080 |
| 2400 | 20160 | 9408 | 5040 |
| 2412 | 10080 | 9474 | 10080 |
| 2472 | 20160 | 10206 | 10080 |
| 2580 | 5040 | 10260 | 5040 |
| 2598 | 10080 | 10554 | 10080 |
| 2688 | 10080 | 11292 | 10080 |
| 2772 | 5040 | 11400 | 10080 |
| 2844 | 10080 | 11424 | 5040 |
| 3006 | 3360 | 12780 | 3360 |
| 3132 | 10080 | 12936 | 5040 |
| 3162 | 20160 | 13536 | 10080 |
| 3222 | 10080 | 14040 | 3360 |
| 3378 | 10080 | 14148 | 3360 |
| 3384 | 10080 | 14826 | 10080 |
| 3516 | 10080 | 14838 | 10080 |
| 3522 | 5040 | 14964 | 10080 |
| 3672 | 5040 | 15444 | 10080 |
| 3762 | 10080 | 17538 | 10080 |
| 3804 | 10080 | 17754 | 10080 |
| 3948 | 10080 | 17802 | 3360 |
| 3954 | 10080 | 17916 | 10080 |
| 3996 | 6720 | 18354 | 10080 |
| 4002 | 10080 | 18666 | 1680 |
| 4152 | 10080 | 19008 | 5040 |
| 4260 | 10080 | 19338 | 10080 |

| Frequency | Number of Distinct Labeled CADS | Frequency | Number of Distinct Labeled CADS |
|---|---|---|---|
| 19440 | 10080 | 74154 | 5040 |
| 20118 | 10080 | 85248 | 10080 |
| 20748 | 10080 | 87150 | 10080 |
| 21486 | 10080 | 90054 | 3360 |
| 27432 | 5040 | 100728 | 1680 |
| 27708 | 10080 | 117090 | 10080 |
| 28170 | 10080 | 129318 | 10080 |
| 28182 | 10080 | 130134 | 5040 |
| 28242 | 10080 | 139218 | 10080 |
| 28674 | 10080 | 141270 | 10080 |
| 31356 | 10080 | 183438 | 10080 |
| 38160 | 3360 | 241002 | 3360 |
| 42462 | 3360 | 241956 | 10080 |
| 43020 | 3360 | 271320 | 5040 |
| 43278 | 5040 | 340854 | 10080 |
| 44748 | 5040 | 371760 | 5040 |
| 48678 | 10080 | 414150 | 10080 |
| 49302 | 10080 | 456948 | 5040 |
| 50142 | 10080 | 606186 | 560 |
| 51558 | 5040 | 652716 | 1120 |
| 53460 | 2016 | 942720 | 10080 |
| 55062 | 10080 | 1047936 | 10080 |
| 55410 | 10080 | 2352612 | 3360 |
| 60510 | 1008 | 2526492 | 3360 |
| 68784 | 10080 | 2567736 | 1680 |
| 69084 | 10080 | 9076864 | 5040 |

$$P[\text{Transitivity}] = \frac{9076864 \times 5040}{7!^3} \approx 0.3573.$$

# Appendix F: Inventory Analysis for 5 Voters, 3 Alternatives

| Frequency | Number of Distinct Labeled CADS | Transitivity Score |
|---|---|---|
| 270 | 2 | 0.3333 |
| 1206 | 6 | 1 |

$$P[\text{Transitivity}] = \frac{1206 \times 6}{3!^5} \approx 0.9306.$$

# Appendix G: Inventory Analysis for 5 Voters, 4 Alternatives

| Frequency | Number of Distinct Labeled CADS | Transitivity Score |
|---|---|---|
| 22410 | 24 | 0.6667 |
| 71010 | 16 | 0.6667 |
| 262000 | 24 | 1 |

$$P[\text{Transitivity}] = \frac{262000 \times 24}{4!^5} \approx 0.7898.$$

# Appendix H: Serial FORTRAN Inventory Code, 3 Voters

```
Program Inventory
  Implicit none
  include "mpif.h"
  Integer :: i,j,k
  Integer,Allocatable :: POrder(:,:)
  Integer,Allocatable :: CAD_Index(:)
  Integer :: N
  Integer :: iargc,n,nchoosek
  Logical :: restart
  Integer :: POV
  Integer :: Pid, N_proc, ierr
  Integer :: TstartA,nlocal,deficit,startl,endA,factorial
  Integer :: startA
  Character argv*10
  Character nstr
  Character n3*8
  Character cadsID*27, ipos*10
  double precision starttime, endtime
  interface
    Integer Function str2int(nn)
      Character nn
      Integer :: N
    end function str2int
    Integer Function Nfact(N)
      Integer :: N
      Integer i,nfct
    end function Nfact
    Integer Function NC2(N)
      Integer, intent(in) :: N
    end function NC2
  end interface
  call MPI_INIT(ierr)
  call MPI_COMM_RANK(MPI_COMM_WORLD,Pid,ierr)
  call MPI_COMM_SIZE(MPI_COMM_WORLD,N_proc,ierr)
```

```
  n = iargc()
  restart = .FALSE.
  if (Pid .EQ. 0) then
     if ((n < 1) .AND. (n > 2)) then
        write(*,*)'  ERROR!! '
        write(*,*)'  The inventory executable requires an input arguement and optional restart'
        write(*,*)'   example:  inventory 3 restart  '
        write(*,*)' '
        STOP
     end if
     do i = 0,n
        call getarg(i,argv)
        if (i .EQ. 1) then
           nstr = argv
        endif
     end do
     if (n .EQ. 2) then
        restart = .TRUE.
     end if
  N = str2int(nstr)
  end if
  call MPI_Bcast(N,1,MPI_INTEGER,0,MPI_COMM_WORLD,ierr)
  call MPI_Bcast(nstr,1,MPI_CHARACTER,0,MPI_COMM_WORLD,ierr)
  cadsID = '_CAD_inventory.txt'
  ipos = '_IPOS.txt'
  cadsID = nstr//cadsID
  ipos = nstr//ipos
  nchoosek = NC2(N)
  Allocate(CAD_Index(1:2**nchoosek))
  Allocate(POrder(1:Nfact(N),1:N))
  CAD_Index = 0
202 format(I2,I2,I2)
  if (Pid .EQ. 0) then
     write(*,*) 'My pid is ',Pid,' ipos is ',ipos
     open(12,FILE = ipos)
     do i = 1,Nfact(N)
        read(12,*)  (POrder(i,j), j = 1,N)
     end do
     close(12)
  end if
  call MPI_BCAST(POrder,(Nfact(N)*N),MPI_INTEGER,0,MPI_COMM_WORLD,ierr)
  starttime = MPI_WTIME()
  call three_inventory(N,nchoosek,POrder,CAD_Index,cadsID,restart,N_proc,Pid)
 if (Pid .eq. 0) then
    open(14,FILE = cadsID)
       write(14,*) (CAD_Index(j), j = 1,2**nchoosek)
       write(14,*) ' '
       write(14,*) 'inventory time is ',endtime-starttime,' secs'
    close(14)
  end if
 deallocate(POrder)
 deallocate(CAD_INDEX)
```

```
 201 format (I2,I2,I2)
END Program Inventory

Subroutine three_inventory(N,nchoosek,PO,CAD_Index,fname,restart,N_proc,Pid)
  include "mpif.h"
  Integer, intent(in) :: N,nchoosek
  Integer, intent(in) :: N_proc,Pid
  Integer, Dimension(1:Nfact(N),1:N), Intent(in)  :: PO
  Integer, Dimension(1:2**nchoosek),Intent(inout)  :: CAD_Index
  Integer, Dimension(1:nchoosek) :: IPO1,IPO2,IPO3
  Integer, Dimension(1:nchoosek) :: SPO
  Integer, Dimension(1:2**nchoosek) :: Local_CAD_Index
  Logical, intent(in) :: restart
  Integer :: a,b,c,j,i
  Integer :: startA, startB, startC
  Integer :: B10_Rep
  Integer :: factorial
  Character, intent(in) :: fname*27
  Integer :: ierr
  Integer :: TstartA,nlocal,deficit,startl,endA
  Local_CAD_Index = 0
  endA = 0
  factorial = Nfact(N)
  nlocal= factorial/N_proc
  startA = (Pid*nlocal) + 1
  endA = nlocal*(Pid+1)
  counter = 0
  if (restart) then
     open(9,FILE = "restart.txt", STATUS = "OLD")
     read(9,*) startA
     read(9,*) (CAD_Index(i), i = 1,2**nchoosek)
     close(9)
  else
     startB = 1
     startC = 1
  end if
  startB = 1
  startC = 1
  do a = startA, endA
     MPI_ALLREDUCE(Local_CAD_Index,CAD_Index,2**nchoosek,MPI_INTEGER,MPI_SUM,MPI_COMM_WORLD,ierr)
     call generate_IPO(PO(a,1:N),N,IPO1,nchoosek)
     do b = startB, factorial
        call generate_IPO(PO(b,1:N),N,IPO2,nchoosek)
        do c = startC, factorial
           call generate_IPO(PO(c,1:N),N,IPO3,nchoosek)
           call generate_B2_vector(IPO1 + IPO2 + IPO3,nchoosek,SPO)
           B10_Rep = 0
           do i = 0,nchoosek-1
              B10_Rep = B10_Rep + SPO(nchoosek-i)*(2**i)
           end do
           Local_CAD_Index(B10_Rep+1) =  Local_CAD_Index(B10_Rep+1) + 1
        end do
```

```
    end do
 end do
 call MPI_ALLREDUCE(Local_CAD_Index,CAD_Index,2**nchoosek,MPI_INTEGER,MPI_SUM,MPI_COMM_WORLD,ierr)
end subroutine three_inventory

Integer Function NC2(N)
  Integer, intent(in) :: N
  NC2 = Nfact(N)/(2*Nfact(N-2))
end function NC2

Integer Function Nfact(N)
  Integer :: N
  Integer i,nfct
  nfct = 1
  do i = 1,N-1
     nfct = nfct*(i+1)
  end do
  Nfact = nfct
end function Nfact

Integer Function str2int(nn)
  character nn
  Integer :: N
  if (nn == '1') then
    N = 1
  elseif (nn == '2') then
    N = 2
  elseif (nn == '3') then
    N = 3
  elseif (nn == '4') then
    N = 4
  elseif (nn == '5') then
    N = 5
  elseif (nn == '6') then
    N = 6
  elseif (nn == '7') then
    N = 7
  elseif (nn == '8') then
    N = 8
  elseif (nn == '9') then
    N = 9
  end if
  str2int = N
end function str2int

Subroutine generate_B2_vector(IPO,nchoosek,b2_vect)
  Integer, Intent(in) :: nchoosek
  Integer, Dimension(1:nchoosek) :: IPO
  Integer, Dimension(1:nchoosek), Intent(out) :: b2_vect
  Integer :: i,j
  do i = 1,nchoosek
    if (IPO(i) > 0) then
```

```
      b2_vect(i) = 1
    else
      b2_vect(i) = 0
    end if
  end do
end subroutine generate_B2_vector

Subroutine generate_IPO(tPO,N,IPO_vector,nchoosek)
  Integer, Intent(in) :: N,nchoosek
  Integer, Dimension(1:N), Intent(in) :: tPO
  Integer, Dimension(1:nchoosek),Intent(out) :: IPO_vector
  Integer :: i,j,k,m,count
  Integer :: ranki,rankj
  count = 1
  do i=1,N
    do k =1,N
      if (tPO(k) == i) then
        ranki=k
        exit
      end if
    end do
    do j = i+1,N
      do m = 1,N
        if (tPO(m) == j) then
          rankj = m
          exit
        end if
      end do
      if (ranki < rankj) then
        IPO_vector(count) = 1
      else
        IPO_vector(count) = -1
      end if
      count = count + 1
    end do
 end do
end subroutine generate_IPO
```

# Appendix I: MATLAB Code for Computing Transitivity Score

```
function Transitive_Scores = tscores(N)
  clear global TCADS;
  clear global CADS;
  global TCADS;
  global CADS;
  length = nchoosek(N,2)
  for k=1:2^nchoosek(N,2)
    CADS_TScore(k)=0;
```

```
  end
  generate_TCADS(N);
  generate_CADS(N);
  for i=1:2^nchoosek(N,2)
    for j=1:factorial(N)
        CADS(:,i)
        TCADS(:,j)
        temp_score = dot(CADS(:,i),TCADS(:,j))/length
        if temp_score > CADS_TScore(i)
            CADS_TScore(i) = temp_score;
        end
    end
    'CAD OF 2^(N CHOOSE 2)';
    i;
  end
  csvwrite('7_TScores.txt', CADS_TScore');
  CADS_TScore'
end

function Generate_Transitive_CADS = generate_TCADS(N)
  global TCADS;
  IPO = perms(1:N);
  for i=1:factorial(N)
      count=1;
      for j=1:N-1
          for a=1:N
              if IPO(i,a)==j
                  rankj=a;
                  a=N;
              end
          end
          for k=j+1:N
              for b=1:N
                  if IPO(i,b)==k
                      rankk=b;
                      b=N;
                  end
              end
              if rankj < rankk
                  TCADS(count,i) = 1;
              else
                  TCADS(count,i) = -1;
              end
              count = count + 1;
          end
      end
  end
end

function Generate_All_CADS = generate_CADS(N)
  global CADS;
  recur_CAD(nchoosek(N,2));
```

```
end

function Recursive_CADS = recur_CAD(length)
  global CADS;
  if length==1
      CADS=[-1 1];
  else
      recur_CAD(length-1);
      for i=1:2^(length-1)
        temp(i) = -1;
        temp(i+2^(length-1)) = 1;
      end
      CADS = [CADS CADS];
      CADS = [temp ; CADS];
  end
  end
```

# Appendix J: SPlus Code for Inventory Output Analysis

```
x    <- scan("CADFreqV5A5.txt")
x    <- sort(x)
f    <- rep(0,1000)
f[1] <- x[1]
c    <- rep(0,1000)
j    <- 1
for(i in 1:length(x))
{ if(x[i] == f[j])
    c[j] <- c[j] + 1
  else
  { j    <- j + 1
    f[j] <- x[i]
    c[j] <- c[j] + 1
  }
}
```

# Appendix K: MATLAB Code for Borda–Perron Manipulability Comparison

```
function TestManipulation = main(V,A,S)
   clear global OutputMatrixB;
   global OutputMatrixB;
   clear global OutputMatrixP;
   global OutputMatrixP;
   clear global MScoresB;
   global MScoresB;
```

```
   clear global MScoresP;
   global MScoresP;
   for a=8:A
      N = num2str(a);
      file_name = strcat('IPOS',N);
      input_file = strcat(file_name, '.txt');
      IPOs = load(input_file);
      for v=2:V
         clear PO;
         for i=1:a
            for j=1:v
               PO(i,j) = 0;
            end
         end
         for i=1:1
            for j=1:S
               MScoresB(i,j) = 0;
               MScoresP(i,j) = 0;
            end
         end
         for i=1:S
            for j=1:v
               PO(:,j) = IPOs(ceil(factorial(a)*rand),:)';
            end
            ManipulationScoreB(PO,IPOs,a,v,i);
            ManipulationScoreP(PO,IPOs,a,v,i);
         end
         OutputMatrixB(v-1,a-2) = mean(MScoresB);
         OutputMatrixP(v-1,a-2) = mean(MScoresP);
       end
   end
   csvwrite('Manipulation_OutputMatrixB.txt', OutputMatrixB);
   csvwrite('Manipulation_OutputMatrixP.txt', OutputMatrixP);
end

function PartialSum = BordaOutcome(PO,V,A)
   for i=1:A
      PartialSum(i) = 0;
   end
   for i=1:V
      for j=1:A
         for k=1:A
            if (PO(j,i) == k)
               PartialSum(k) = PartialSum(k) + A - (j - 1);
               break;
            end
         end
      end
   end
end

function OrderedSum = Sort(PartialSum,A)
```

```
   OrderedSum = PartialSum;
   for i=1:A-1
      for j=1:A-i
         if (OrderedSum(j+1) > OrderedSum(j))
            temp = OrderedSum(j);
            OrderedSum(j) = OrderedSum(j+1);
            OrderedSum(j+1) = temp;
         end
      end
   end
end

function Manipulation = ManipulationScoreB(PO,IPOs,A,V,index)
   global MScoresB;
   MaxScore     = 0;
   MaxIndex     = 0;
   Multiplicity = 0;
   PartialSum = BordaOutcome(PO,V,A);
   OrderedSum = Sort(PartialSum,A);
   for i=1:A
      Tpoints(i) = 0;
   end
   for i=1:factorial(A)
      MaxScore     = 0;
      Multiplicity = 0;
      Outcome      = OrderedSum + IPOs(i,:);
      for j=1:A
         if (Outcome(j) > MaxScore)
            MaxScore = Outcome(j);
         end
      end
      for j=1:A
         if (Outcome(j) == MaxScore)
            Multiplicity = Multiplicity + 1;
         end
      end
      for j=1:A
         if (Outcome(j) == MaxScore)
            if (Tpoints(j) < 1/Multiplicity)
               Tpoints(j) = 1/Multiplicity;
            end
         end
      end
   end
   for i=1:A
      MScoresB(index) = MScoresB(index) + Tpoints(i);
   end
   MScoresB(index) = (MScoresB(index)-1)/(A-1);
end

function Manipulation = ManipulationScoreP(PO,IPOs,A,V,index)
   global MScoresP;
```

```
   MaxScore     = 0;
   MaxIndex     = 0;
   Multiplicity = 0;
   for i=1:A
       for j=1:A
           PartialPerronMatrix(i,j) = 0;
       end
   end
   for i=1:V
       PartialPerronMatrix = PartialPerronMatrix + GenerateMatrixPO(PO(:,i),A);
   end
   PartialPerronMatrix;
   for i=1:A
       Tpoints(i) = 0;
   end
   for i=1:factorial(A)
      MaxScore      = 0;
      Multiplicity  = 0;
      OutcomeMatrix = PartialPerronMatrix + GenerateMatrixPO(IPOs(i,:),A);
      [V, D] = eig(OutcomeMatrix);
      OutcomeVector = abs(V(:,A));
      for j=1:A
         if (OutcomeVector(j) > MaxScore)
            MaxScore = OutcomeVector(j);
         end
      end
      for j=1:A
         if (OutcomeVector(j) == MaxScore)
            Multiplicity = Multiplicity + 1;
         end
      end
      for j=1:A
         if (OutcomeVector(j) == MaxScore)
            if (Tpoints(j) < 1/Multiplicity)
               Tpoints(j) = 1/Multiplicity;
            end
         end
      end
   end
   for i=1:A
      MScoresP(index) = MScoresP(index) + Tpoints(i);
   end
   MScoresP(index) = (MScoresP(index)-1)/(A-1);
end

function MatrixPO = GenerateMatrixPO(PO,A)
   for a=1:A
      for b=1:A
         MatrixPO(a,b) = 0;
      end
   end
   for i=1:A
```

```
        for k=1:A
            if PO(k)==i
                ranki=k;
                break;
            end
        end
        for j=i+1:A
            for m=1:A
                if PO(m)==j
                    rankj=m;
                    break;
                end
            end
            if ranki < rankj
                MatrixPO(i,j) = 1;
                MatrixPO(j,i) = 0;
            else
                MatrixPO(j,i) = 1;
                MatrixPO(i,j) = 0;
            end
        end
    end
end
```

# Appendix L: MATLAB Code for Standard–Exponential Convex Borda Manipulation Scores

```
function WeightedBordaManipScore = main(V,A,S,D)
   clear global TotalScore;
   global TotalScore
   clear global MScores;
   global MScores;
   clear global output_file;
   global output_file;
   output_file = fopen(strcat(num2str(V),num2str(A),'_ConvexBordaManipScore.txt'),'w');
   for i=1:V
       for j=1:A
           for k = 1:(1/D+1)
               MScores(i,j,k)  = 0;
           end
       end
   end
   for i=1:A
      PartialSum(i) = 0;
      OrderedSum(i) = 0;
   end
   for voters = 2:V
      for alts = 3:A
         IPOs = load(strcat(strcat('IPOS',num2str(alts)), '.txt'));
```

```
        for i = 1:(1/D+1)
           TotalScore(i) = 0;
        end
        for j = 1:S
           for k=1:voters
              PO(:,k) = IPOs(ceil(factorial(alts)*rand),:)';
           end
           delta = 0;
           for i = 1:(1/D+1)
              PartialSum    = BordaOutcome(PO,voters,alts,delta);
              OrderedSum    = Sort(PartialSum,alts);
              TotalScore(i) = TotalScore(i) + GetM(OrderedSum,IPOs,voters,alts,delta);
              delta         = delta + D;
              MScores(voters,alts,i) = TotalScore(i)/S;
           end
        end
        clear PO;
     end
  end
  delta = 0;
  fprintf(output_file, 'Sample Size ...... = %1.0f \n', S);
  fprintf(output_file, 'Total Voters ..... = %1.0f \n', V);
  fprintf(output_file, 'Total Alternatives = %1.0f \n', A);
  fprintf(output_file, 'Step Size ........ = %1.5f \n', D);
  fprintf(output_file, 'Exponent Base .... = e \n\n');
  fprintf(output_file, 'Delta     V     A     ManipulationScore \n');
  for i = 1:(1/D+1)
     for voters = 2:V
        for alts = 3:A
           fprintf(output_file, '%1.3f     %1.0f     %1.0f     %1.5f \n',
                   delta, voters, alts, MScores(voters, alts, i));
         end
     end
     delta = delta + D;
 end
end

function PartialSum = BordaOutcome(PO,V,A,delta)
  for i=1:A
     StandardSum(i) = 0;
  end
  for i=1:V
     for j=1:A
        for k=1:A
           if (PO(j,i) == k)
              StandardSum(k) = StandardSum(k) + A - (j-1);
              break;
           end
        end
     end
  end
  for i=1:A
```

```
      ExponSum(i) = 0;
   end
   for i=1:V
      for j=1:A
         for k=1:A
            if (PO(j,i) == k)
               ExponSum(k) = ExponSum(k) + exp(A+1-j);
               break;
            end
         end
      end
   end
   for i=1:A
      PartialSum(i) = (1-delta)*StandardSum(i) + delta*ExponSum(i);
   end
end

function OrderedSum = Sort(PartialSum,A)
   OrderedSum = PartialSum;
   for i=1:A-1
      for j=1:A-i
         if (OrderedSum(j+1) > OrderedSum(j))
            temp            = OrderedSum(j);
            OrderedSum(j)   = OrderedSum(j+1);
            OrderedSum(j+1) = temp;
         end
      end
   end
end

function Manipulation = GetM(OrderedSum,IPOs,V,A,delta)
   global MScores;
   MaxScore     = 0;
   MaxIndex     = 0;
   Multiplicity = 0;
   Manipulation = 0;
   for i=1:A
      Tpoints(i) = 0;
   end
   for i=1:factorial(A)
      MaxScore     = 0;
      Multiplicity = 0;
      Outcome      = OrderedSum + BordaOutcome(IPOs(i,:)',1,A,delta);
      for j=1:A
         if (Outcome(j) > MaxScore)
            MaxScore = Outcome(j);
         end
      end
      for j=1:A
         if (Outcome(j) == MaxScore)
            Multiplicity = Multiplicity + 1;
         end
```

```
      end
      for j=1:A
         if (Outcome(j) == MaxScore)
            if (Tpoints(j) < 1/Multiplicity)
               Tpoints(j) = 1/Multiplicity;
            end
         end
      end
   end
   for i=1:A
      Manipulation = Manipulation + Tpoints(i);
   end
   Manipulation = (Manipulation-1)/(A-1);
end
```

# Bibliography


[1] Abello, J., Johnson, C.R. "How Large Are Transitive Simple Majority Domains?" *SIAM Journal on Algebraic and Discrete Methods*, 5.4 (1984): 603-618.

[2] Arrow, K.J. "A Difficulty in the Concept of Social Welfare." *Journal of Political Economy*, 58.4 (August 1950): 328-346.

[3] Baker, K.M. Condorcet: From Natural Philosophy to Social Mathematics. University of Chicago Press : Chicago, Illinois (1975).

[4] Dietzenbacher, Erik. "Aggregation in Multisector Models: Using the Perron Vector." *Economic Systems Research*, 4.1 (1992): 3-24.

[5] Fishburn, Peter C. The Theory of Social Choice. Princeton University Press : Princeton, New Jersey (1973).

[6] Gibbard, Allan. "Manipulation of Voting Schemes: A General Result." *Econometrica*, 41.4 (1973): 587-601.

[7] Handbooks in Economics: Handbook of Social Choice and Welfare, 19.1. Arrow, K., Sen, A., Suzumura, K, editors. North-Holland : New York, New York (2002).

[8] Horn, Roger, and Johnson, Charles. Matrix Analysis. Cambridge University Press: New York, New York (1985).

[9] Johnson, C.R. "Whither Two-Thirds Majorities?" Working Paper.

[10] Kelly, Jerry S. Arrow Impossibility Theorems. Economic Theory and Mathematical Economics : New York, New York (1978).

[11] Kelly, Jerry S. Social Choice Theory, An Introduction. Springer-Verlag : New York, New York (1988).

[12] Levenglick, A., Young, H.P. "A Consistent Extension of Condorcet's Voting Principle." *SIAM Journal on Applied Mathematics*, 35.2 (1978): 285-300.

[13] McGarvey, David C. "A Theorem on the Construction of Voting Paradoxes." *Econometrica*, 21.4 (1953): 608-610.

[14] Mirkin, Boris G. Group Choice. V.H. Winston & Sons : Washington, D.C. (1979).

[15] Reny, Philip, J. "Arrow's Theorem and the Gibbard–Satterthwaite Theorem: A Unified Approach." *Economic Letters*, 70.1 (2001): 99-105.

[16] Saari, Don. "The Symmetry and Complexity of Elections." Working Paper: Department of Mathematics, Northwestern University. Available online $< http : //www.colorado.edu/education/DMP/voting_b.html >$.

[17] Satterthwaite, Mark A. "Strategy-proofness and Arrow's Conditions: Existence and Correspondence Theorems for Voting Procedures and Social Welfare Functions." *Journal of Economic Theory*, 10 (April 1975): 187-217.

[18] Stearns, Richard. "The Voting Problem." *The American Mathematical Monthly*, 66.9 (November 1959): 761-763.

[19] Taylor, Alan D. *Social Choice and the Mathematics of Manipulation.* Cambridge: Cambridge University Press, 2005.

[20] Taylor, Alan D. "The Manipulability of Voting Systems." *The American Mathematical Monthly*, April 2002.

[21] Young, H.P. “A Note on Preference Aggregation.” *Econometrica*, 42.6 (1974): 1129-1131.

[22] Young, H. P. “Condorcet’s Theory of Voting.” *The American Political Science Review*, 82.4 (1988): 1231-1244.

[23] Young, H.P. “Optimal Voting Rules.” *The Journal of Economic Perspectives*, 9.1 (1995): 51-64.

[24] Young, H.P. “Social Choice Scoring Functions.” *SIAM Journal on Applied Mathematics*, 28.4 (1975): 824-838.