WASHINGTON UNIVERSITY

Sever Institute
School of Engineering and Applied Science

Department of Computer Science and Engineering

Dissertation Examination Committee:
Ron K. Cytron, Chair
Steven Brams
Jeremy Buhler
Robert Pless
Itai Sened
Aaron Stump

COMPUTATIONAL ASPECTS OF APPROVAL VOTING AND DECLARED-STRATEGY VOTING

by

Robert Hampton LeGrand III, B.S., M.C.S.

A dissertation presented to the
Graduate School of Arts and Sciences
of Washington University in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy

May 2008

St. Louis, Missouri

Acknowledgements

I will always be grateful to my research advisor, Ron K. Cytron, for his guidance, perspective, encouragement and patience. He has been a pleasure to work with and a great help in exploring and evaluating research ideas and directions. I am passionate about this research and feel very lucky to have had the opportunity to work with him on it.

I also appreciate Steven J. Brams of NYU for his groundbreaking ideas that inspired much of this work. He has been extraordinarily encouraging and generous with his time, and it was a great honor to have him serve on my dissertation examination committee.

I would also like to thank the rest of my committee, Jeremy Buhler, Robert Pless, Itai Sened and Aaron Stump, for their time and valuable comments, and the DOC Group, especially James Brodman, Delvin Defoe, Morgan Deters, Scott Friedman, Richard Hough, Tobias Mann and Justin Thiel, who made it such a fun lab to work in.

Finally, for their love, support and pride, I owe my parents, Robert H. LeGrand Jr. and Marsha LeGrand, the most thanks of all.

Robert Hampton LeGrand III

Washington University, St. Louis

May 2008

Contents

List of Tables

List of Figures

ABSTRACT OF THE DISSERTATION

Computational Aspects of Approval Voting
and Declared-Strategy Voting

by
Robert Hampton LeGrand III

May 2008
Washington University
St. Louis, Missouri

Professor Ron K. Cytron, Chairperson

Computational social choice is a relatively new discipline that explores issues at the intersection of social choice theory and computer science. Designing a protocol for collective decision-making is made difficult by the possibility of manipulation through insincere voting. In approval voting systems, voters decide whether to approve or disapprove available alternatives; however, the specific nature of rational approval strategies has not been adequately studied. This research explores aspects of strategy under three different approval systems, from chiefly a computational viewpoint.

While traditional voting systems elicit only the outcome of a voter's strategic thinking, a Declared-Strategy Voting (DSV) system accepts such strategies directly and applies them according to the voter's preferences over the available alternatives. Ideally, when rational strategies are employed on behalf of the voters, voters are discouraged from expressing insincere preferences. Approval voting is a natural fit for use with DSV, but, unlike for the common plurality voting system, there is no extant theory regarding the most effective approval strategies in a DSV context. We propose such a theory.

Approval-rating polls already serve an important role in assaying the views of an electorate on some subject of interest. Sites such as Rotten Tomatoes and Metacritic.com collect and display the results of approval-rating polls for movies and games. Moreover, sites such as Amazon and eBay

collect approval ratings to estimate the worthiness of their buyers and sellers. In these polls, a rational voter's approval or disapproval will sometimes be insincere so as to move the result in a desired direction. A nonmanipulable protocol would allow indication of a voter's ideal outcome and would never reward an insincere such indication. We present and analyze a large new class of such nonmanipulable protocols motivated by the DSV concept.

The minimax procedure is a multiwinner form of approval voting that aims to maximize the satisfaction with the outcome of the least satisfied voter. Unfortunately, computing the minimax winner set is computationally hard. We propose an approximation algorithm for this problem, a framework for polynomial-time heuristics that perform very well in practice, and a preliminary analysis of strategic voting under minimax.

Preface

Computer technology has been made to serve mankind in many ways. Today computers make simple many tasks that were previously more difficult or even impossible. Many public elections remained largely mechanical with little help from computers well into the information age, but computerized voting systems have received rapidly increasing attention since 2000 [40, 31]. Much has been written [3, 41] about real and theoretical computerized systems that verify voters, collect ballots and count votes, but there has been relatively little exploration of the possibility of computers assisting voters with making strategic voting decisions.

When voting in elections, voters often find that a sincere ballot is unlikely to be the most effective one. For example, imagine an election for a single winner with three alternatives in which each voter is allowed to give a vote to one alternative. If a specific voter prefers A to B and B to C, but estimates that A is likely to finish a distant third, that voter may decide that choosing B is more likely to have a positive effect on the outcome. The outcome of an election can depend greatly on the extent and quality of this kind of strategically insincere voting in the electorate.

But choosing the most effective ballot is not always straightforward for a human voter, especially when the voter has little information on the alternatives' relative strengths. Even when rich information regarding the current vote totals and other voters' preferences is available, the most effective ballot—that is, the one that is likeliest to lead to the optimal reachable outcome—may not be obvious to a human voter. What is needed, then, is a system that will carry out the calculations required to find an optimally effective ballot for the voter. Such a system would make both naïvely sincere and sophisticated voters equally effective.

Chapter 1

Introduction and Background

In this chapter we will introduce important concepts and review previous work relevant to our research directions.

1.1 Declared-Strategy Voting

In 1996, Lorrie Cranor and Ron K. Cytron [23] described a hypothetical voting system they called Declared-Strategy Voting (DSV). DSV arose from the desire to elicit richer, more sincere[1] preferences from voters by using that information to find a winning alternative in such a way that voters would be unlikely to gain a superior result by submitting insincere preferences.

DSV can be seen as a meta-voting system, in that it uses voters' expressed preferences among alternatives to vote rationally in their stead in repeated simulated elections. The repeated simulated elections are run according to the rules of some underlying voting protocol, which can be any protocol that accepts any kind of ballots and uses them to elect one winner. Cranor [22] explored using DSV with plurality, but DSV, as a meta-voting system, could conceivably work with any voting protocol.

[1]Unfortunately, the terms "sincerity" and "manipulation" are used with little consistency in the social-choice literature. We will review the various uses of these terms and define the senses in which we use them, "sincerity" in this chapter and "manipulation" in the next.

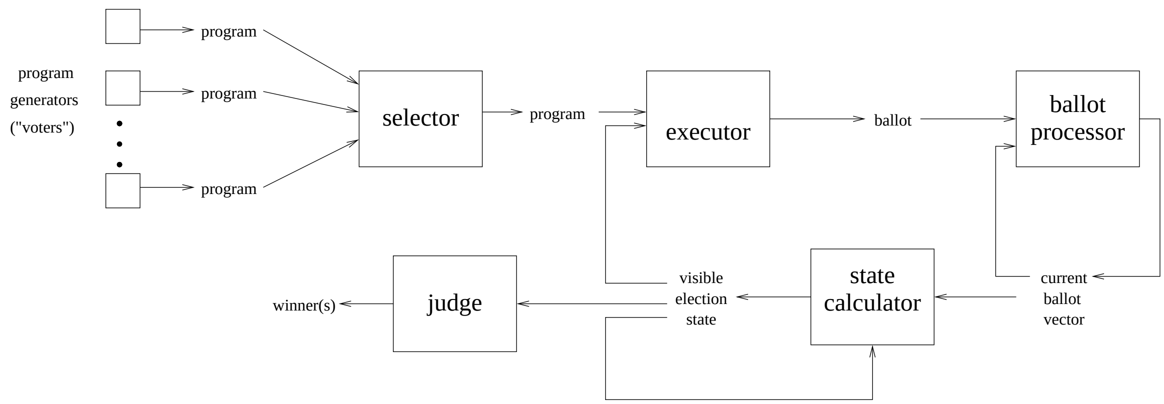


Figure 1.1: Outline of DSV operation

As depicted in Figure 1.1, a DSV system maintains a *ballot vector* and an *election state* through a series of rounds. The ballot vector is empty at the beginning of the election and is updated after each voter's program executes to hold all of the current ballots. The election state consists of a vector of rational numbers, one for each alternative in the election, that correspond to the current vote totals. It is updated using the ballot vector; the DSV mode (described below) of the election determines when and how that update takes place.

In a DSV election, a voter submits not a ballot but a DSV *program*. A program is essentially a function that takes some set of updated information about an election in progress as input and uses it to decide on a ballot to vote in the voter's stead. In the most general case, the program itself can be any algorithm taking in all input related to the election in progress, such as all ballots previously voted, the contents of all other submitted programs and the number of ballots processed so far. But few real-world voters are sophisticated enough to provide such a program directly.

In this research, a program is assumed to consist of a set of a voter's *cardinal preferences* (or utility ratings) over the alternatives and a rule, known as a *declared strategy*, that generates an appropriate ballot at that point in the election. The cardinal preferences (ratings) are constrained to rational numbers between 0 and 1 inclusively; a rating measures the utility of an alternative to a voter—the degree to which that alternative's victory is desirable. For example, if alternatives are candidates for public office, a rating can be interpreted as an estimate of the proportion of issues on which the voter and candidate agree, weighted by importance to the voter and likelihood of relevance during the period of representation. Ratings are assumed to scale linearly and otherwise behave as von Neumann–Morgenstern utilities [57].

The declared strategy is a precisely defined algorithm that takes as input only the current election state, the voter's cardinal preferences and the voter's previously voted ballot; notably, the declared strategy has no knowledge of when it will next be executed or which voters' programs have already executed. The declared strategy is expressed by submitting a well defined algorithm or choosing one from a predetermined list. The aim of the declared strategy is to maximize the result of the election according to the voter's submitted cardinal preferences.

Once all programs are submitted, a *selector* selects a program to be executed. The selector effectively determines the order in which the programs are executed. The ith *round* consists of the ith execution of each program and the processing of the resulting ballots. Each program must be selected i times before any program may be selected in round $(i+1)$, but apart from that restriction the selection is random and impartial.

An *executor* interprets the selected program and runs it with the current election state as input to produce a ballot as output. Only a finite number of computational steps is allowed to the program; if the program has not finished execution in the allotted number of steps, it is halted and an empty ballot results. This number can be made large enough for any reasonable program to complete comfortably but must be finite to guarantee progress and eventual completion of the election. The declared strategies considered in this research will be polynomial-time algorithms and will be assumed to finish in the allotted number of steps.

The ballot that is generated is then passed on to the *ballot processor*, which checks whether the ballot is well formed according to the underlying voting protocol. For example, if plurality were being used as the underlying voting protocol, the ballot $[1, 1, 0]$ would be rejected since plurality only allows one vote for at most one alternative. If the ballot is accepted as valid, the ballot processor then integrates it into the current ballot vector; the election's *mode* determines how that integration occurs.

The two modes described by Cranor [22] are ballot-by-ballot mode and batch mode. In this research, a mode can further be *non-cumulative* (like Cranor's modes) or *cumulative*, giving four possible DSV modes: ballot-by-ballot, batch, cumulative ballot-by-ballot and cumulative batch. If an election is run in one of the cumulative modes, the ballot processor adds a newly voted ballot to the ballot vector; if a non-cumulative mode is used, the ballot processor uses a newly voted ballot to replace that voter's previously voted ballot, so that each voter has at most one ballot in the ballot vector at any time.

The *state calculator* uses the current ballot vector to update the election state; the election's mode also determines how this update occurs. If the system is in a ballot-by-ballot mode, whether cumulative or not, the ballots in the ballot vector are summed and the summed vector replaces the previous election state after each ballot is processed. In a batch mode, the summed vector replaces the previous election state only after the last ballot of each round, that is, when each program has been executed the same number of times.

The underlying voting protocol (*e.g.*, plurality), the mode (*e.g.*, non-cumulative batch mode) and the number of rounds (*e.g.*, 30) are the only settings needed to specify a DSV voting system fully.

1.2 Approval voting

Approval voting is a simple single-winner voting protocol. It was used in the Republic of Venice and to elect the pope in the thirteenth through seventeenth centuries [47], and it was rediscovered independently in the 1970s by several authors, including Guy Ottewell [44], Robert Weber [59] and Steven Brams and Peter Fishburn [13]. Under approval voting each voter may approve any subset of the available alternatives, effectively recording a *yes* or *no* vote for each alternative. In this version of approval voting that elects exactly one of a finite number of discrete alternatives, the one alternative that receives the most approval votes is chosen as the winner.

We propose that approval voting is a good candidate for use with DSV. DSV with plurality has been previously explored [22]; a plurality vote that can be expected to maximize a voter's utility of the eventual outcome often deserts a favorite alternative to vote for another. More generally, it is sometimes rational to vote for A (and not B) even though B is preferred to A. Under approval voting a voter could vote fully for the same compromise alternative while also supporting fully his or her favorite. Assuming the favorite has a nonzero chance of winning, doing so will further increase expected utility of the outcome, so perhaps approval voting induces less insincere voting by some measure. For example, unlike under plurality, it may never be rational under approval voting to vote for A and not B when B is preferred to A.

1.3 Notions of sincerity

Under most voting systems, *insincere* voters can gain an advantageous outcome. Specifically, for most systems it sometimes happens that the most effective ballot contradicts a voter's true preferences. For example, if a voter's true preferences are for A over B and B over C and the other votes total 40 for A, 50 for B and 50 for C, the single most effective plurality ballot is one for B. It is reasonable to consider such a ballot insincere because it expresses a pairwise preference for B over A, contradicting the voter's sincere preferences.

Standard since Arrow's seminal impossibility result [4] in the realm of voting theory is to assume ordinal preferences and ordinal ballots. In such a world, a sincere ordinal ballot is simply one that exactly reflects the voter's ordinal preferences. (Of course, if voters may have tied preferences, then they must be allowed to vote tied rankings to maintain the possibility of sincerity.)

This notion of sincerity can be generalized to cardinal preferences and ballots in different ways, but we focus on two in this work.[2] First, If voters are assumed to have Von Neumann–Morgenstern [57] (cardinal and linearly scalable) utilities over the available alternatives, where $p(i)$ is a voter's cardinal preference (utility) for alternative i, and any particular allowed ballot is interpreted to assign a rating $v(i)$ to each alternative, then at least two notions of sincerity can be easily defined:

strong sincerity A ballot is strongly sincere if and only if, for all alternatives i and j,

$$v(i) > v(j) \longleftrightarrow p(i) > p(j).$$

[2]We assume that any voting protocol accepts ballots that can be interpreted to assign some rating to each alternative. For example, plurality only allows assigning the rating 1 to one alternative and 0 to the rest; approval voting allows assigning either 0 or 1 to each alternative; ranked-ballot voting systems allow any assignment of rational numbers to the alternatives, where alternatives given higher ratings are taken to be ranked ahead of those given lower ratings. So cardinal-ratings ballots nicely generalize a large class of ballots without loss of information, though each voting protocol has its own set of allowed ballots.

weak sincerity A ballot is weakly sincere if and only if, for all alternatives i and j, $v(i) > v(j) \longrightarrow p(i) > p(j)$. (This definition is equivalent to the definition of sincerity given by Brams and Fishburn [14, p. 29].)

So, for example, a voter with sincere utilities over three alternatives $[1, 0.8, 0]$ might vote the approval ballot $[1, 1, 0]$; such a ballot is weakly but not strongly sincere. (Note that no strongly sincere approval ballot exists for a voter with tri- or multichotomous [14, p. 17] preferences such as these.)

Merrill [38, p. 80] outlines different notions of sincerity specifically for approval voting. He describes an approval ballot that is not even weakly sincere as a *skipping* ballot, as such a ballot's approvals "skip" down the voter's preference ordering. For example, a voter who prefers A to B to C to D but approves only A and C is voting a skipping ballot. By this definition, an approval ballot is weakly sincere if and only if it is not skipping.

Notice that, if a voter knows exactly how every other voter will vote, a skipping ballot cannot be uniquely best. In other words, for any skipping ballot, there is some weakly sincere approval ballot that obtains an outcome which is no worse. For example, if C and D are tied for the win, then, for the voter mentioned above, approving B as well as A and C can only help; if A and B are tied for the win, then approving C as well as A can only hurt.

Also specifically for approval voting, Merrill defines "pure" sincerity:

pure sincerity An approval ballot is purely sincere if and only if, for all k alternatives i, $p(i) > \frac{\sum_j p(j)}{k} \longrightarrow v(i) = 1$ and $p(i) \leq \frac{\sum_j p(j)}{k} \longrightarrow v(i) = 0$.

Note that, according to this definition, if a voter assigns equal utilities to all alternatives, a purely sincere ballot would disapprove all alternatives.

Gibbard [30] and Satterthwaite [53] independently showed that, for any voting protocol that treats ballots and alternatives symmetrically, the most effective ballot is not always strongly sincere when there are at least three alternatives.[3] The two protocols most often given as examples to which the

[3] The Gibbard–Satterthwaite theorem considers protocols with fully ranked input, so every weakly sincere ballot is also strongly sincere. The theorem also effectively applies to protocols that accept ballots with tied ranks, but says only that *strong* (and not weak) sincerity must sometimes be violated by a rationally strategic voter.

Gibbard–Satterthwaite theorem does not apply are *random ballot* (a ballot is selected randomly; the highest rated alternative on it wins) and *random runoff* (two alternatives are randomly chosen; the one preferred to the other on more ballots wins). Neither treats ballots and alternatives symmetrically and thus are not generally considered appropriate for real-world elections.

So, every reasonable voting protocol sometimes rewards departing from strong sincerity, but, as we will further see below in section 1.4, approval voting can be said never to reward departing from *weak* sincerity. Other well-known protocols such as plurality, Hare (STV) [8] and Borda [51] cannot make the same claim.

1.4 Existing strategic approaches

While a DSV program can in general be any piece of code that a voter submits, rational program-writers will be attempting to generate a ballot that takes their cardinal preferences into account. Accordingly, we have assumed that a program consists of (1) cardinal preferences over the alternatives and (2) a declared strategy, which uses the election state and the cardinal preferences to find a ballot that is deemed likeliest to maximize the election result according to the preferences.

Several authors have investigated concepts very similar to what we call declared strategies. Brams and Fishburn [14, ch. 7] explore a concept they call the "poll assumption", which models changes in voters' strategy given the election state (the "poll") under both plurality and approval voting. One variation, which they call the "Poll Assumption (Approval)", is defined on page 115 (they use the term "strategy" to mean what we call "ballot"):

> After the poll, voters will adjust their voting strategies [ballots] to distinguish between the top two candidates, as indicated by the poll [election state], if they prefer one of these candidates to the other and their sincere, pre-poll strategies did not involve voting for exactly one of these choices. Given that they are not indifferent between the top two candidates in the poll, they will vote after the poll for their preferred candidate *and all candidates preferred to him (if any).* [14]

Below (Figure 1.4) we will call this poll assumption “strategy B” and define it more precisely. Brams and Fishburn give examples that show alternatives being hurt and helped when voters strategically respond to poll information using this poll assumption; we will see that there exist other reasonable ways to respond to polls that may result in different equilibria when all voters use them.

Chapter 5 of Merrill [38] constructs a general theory of strategy under uncertainty (where nothing is known about the alternatives' relative chances of winning) and risk (where each alternative's probability of winning is assumed to be known or estimated). His approach to strategy under risk assumes knowledge of pivot probabilities t_{ij}, where t_{ij} is the probability that, given that the election results in a tie between two alternatives, alternatives i and j are the participants in the tie. He uses these pivot probabilities to calculate a *strategic value* for each alternative: the expected benefit according to the voter's cardinal preferences of adding one vote for that alternative. He then finds, of all valid ballots, the one that maximizes the total strategic value.

Cranor [22] offers a theory of rational declared strategies and applies it especially to plurality DSV elections. She outlines several approaches to transforming an election state into pivot probabilities, the most prominent of which essentially treats each alternative's proportion of the votes in the current election state as the mean of a Gaussian distribution of that alternative's eventual proportion of the votes, assumes some particular variance for the distribution (S^2, which could be seen as a measure of uncertainty in the current vote totals), and then calculates the probability that each pair of alternatives will eventually tie for the win. When voters are randomly sampled to obtain the distribution, the variance can be modeled according to the Gaussian error of estimate based on the fraction of the electorate sampled so far. The resulting pivot probabilities are then used as by Merrill to find an optimal plurality ballot.

Cranor's approach attempts to define rational strategy directly by making certain reasonable assumptions but does not atttempt to show that, in practice, the strategies found lead to better results for voters that use them than any other strategy approaches.

1.5 Computationally simple approval strategies

Several existing styles of designing effective declared strategies were described in section 1.4. Some, such as the strategies of Brams and Fishburn, were intended to model approximately how typical real-world voters might vote when they have poll information. Such strategies are desirable for DSV for three reasons: they are based on results that have already appeared in the literature, they can be described simply enough for a human voter to understand easily and they can be executed quickly by a computer regardless of the numbers of voters and alternatives. Several such strategies are defined below; they all are computationally simple and can be found in extant literature. All of them always result in a weakly sincere approval ballot, essentially setting some cardinal cutoff over which alternatives are approved and under which they are disapproved, so skipping ballots never result.

To simplify the description of approval strategies, we will assume that DSV is run in non-cumulative batch mode, which means that the election state visible to a voter in round $r+1$ depends only on the ballots submitted in round r.

Notation varies widely in the literature, but we will describe election situations using the following. Any election has k alternatives numbered from 1 to k and n voters numbered from 1 to n. For notational convenience, $\mathbb{Z}_n$ is defined as the set $\{z \in \mathbb{Z} : 1 \leq z \leq n\}$ (the integers between 1 and n) and $\mathbb{Q}_{m \ldots n}$ is defined as the set $\{q \in \mathbb{Q} : m \leq q \leq n\}$ (the rational numbers between m and n).

The voters' sincere cardinal preferences are represented as the function $p : \mathbb{Z}_n \times \mathbb{Z}_k \to \mathbb{Q}$, where $p(i,j)$ = voter i's cardinal preference for alternative j. The ballots submitted throughout the DSV rounds are represented as the function $b : \mathbb{Z}_n \times \mathbb{Z}_k \times \mathbb{N} \to \mathbb{Q}_{0 \ldots 1}$, where $b(i,j,r)$ = voter i's vote for alternative j in round r; $b(i,j,0) = 0$ for all i and j.[4] The election states are represented as the function $s : \mathbb{Z}_k \times \mathbb{N} \to \mathbb{Q}_{0 \ldots n}$, where $s(j,r) = \sum_{i=1}^{n} b(i,j,r)$, alternative j's vote total at the end of round r.

The approval strategies defined below calculate the ballot for voter v at round $r+1$; that is, given v and r, they calculate $b(v,i,r+1)$ ($= 1$ for approval and 0 for disapproval) for each alternative i.

[4] Here $\mathbb{N}$ denotes the set of nonnegative integers: $\mathbb{N} = \{n \in \mathbb{Z} : n \geq 0\}$.

It will be useful to define two more functions. $\text{Top}_y(r)$ is the set of the y leading alternatives according to round r's election state:

$$\text{Top}_y(r) = \{i : |\{j : s(i,r) < s(j,r)\}| < y\}$$

$\text{PSum}_y(v,r)$ is the sum of voter v's cardinal preferences of the alternatives in $\text{Top}_y(r)$:

$$\text{PSum}_y(v,r) = \sum_{j \in \text{Top}_y(r)} p(v,j)$$

Also, we define the set $C(r)$ of "contending" alternatives at round r to be $\{i : s(i,r) \geq x\}$ where x is as large as possible with the constraint that $|C(r)| \geq 2$ for all r.

One of the simplest strategies effectively ignores the election state and assumes that each alternative has an equal probability of winning the election, approving each alternative the voter rates higher than his average rating over all alternatives. We call this approach strategy Z and define it precisely in Figure 1.2. Note that the definition uses Top and PSum to make clear its similarities with those of the other strategies we will consider, but as only Top_k and PSum_k are used the effect is to compare each alternative's utility to the average utility over all the alternatives, so the election state is effectively ignored.

Figure 1.2: Approval strategy Z

For voter v voting in round $r+1$,

- for each alternative i:
  - approve alternative i if and only if $p(v,i) \cdot |\text{Top}_k(r)| > \text{PSum}_k(v,r)$

Strategy Z is always purely sincere according to Merrill's definition above. It is also equivalent to the optimal approval strategy according to the Laplace method for making decisions under uncertainty Merrill [38] gives in chapter 5.

Another strategy based only on a voter's preferences and the alternatives' current vote totals is often given by advocates of approval voting as a good rule of thumb for real-world approval elections [47, p. 196]. Mike Ossipoff [43] wrote, "In Plurality, if you're sure that Smith & Jones will be the top 2 votegetters, then obviously you should vote for whichever of those 2 you prefer to the other. In Approval, vote for him and for everyone whom you like better." Equivalently, an approval cutoff is placed just below the preferred of the top two alternatives. We call this approach strategy T and define it more precisely in Figure 1.3.

Figure 1.3: Approval strategy T

For voter v voting in round $r+1$,

- for each alternative i:
  - find smallest y such that $p(v,i) \cdot |\mathrm{Top}_y(r)| \neq \mathrm{PSum}_y(v,r)$ ($y = k$ if none)
  - if $y \leq 2$:
    * approve alternative i if and only if $p(v,i) \cdot |\mathrm{Top}_1(r)| > \mathrm{PSum}_1(v,r)$ and $p(v,i) \cdot (|\mathrm{Top}_2(r)| - |\mathrm{Top}_1(r)|) > \mathrm{PSum}_2(v,r) - \mathrm{PSum}_1(v,r)$
  - else:
    * approve alternative i if and only if $p(v,i) \cdot |\mathrm{Top}_y(r)| > \mathrm{PSum}_y(v,r)$

The first of the two strategies Brams and Fishburn [14] describe we will call strategy B; the second we will call strategy J. Strategy B takes as input not only a voter's preferences and the alternatives' current vote totals but also the voter's previous ballot. We use the description of the Poll Assumption (Approval) and the ensuing examples to inspire the precise definition of Strategy B in Figure 1.4. Strategy B will change a voter's ballot only when necessary to differentiate among the contending alternatives.

Figure 1.4: Approval strategy B

For voter v voting in round $r+1$,

- if $r > 0$:
  - b_{-1} = voter v's ballot cast in round r
- else:
  - b_{-1} = ballot found by applying strategy Z
- if b_{-1} includes *all* alternatives in $C(r)$ or *none* of them:
  - apply strategy T
- else:
  - use voter v's previously cast ballot

Brams and Fishburn describe a variation on their Poll Assumption (Approval) on page 120, which can be generalized to what we call strategy J and define precisely in Figure 1.5. Strategy J will vote a purely sincere ballot whenever doing so would differentiate among the contending alternatives.

Figure 1.5: Approval strategy J

For voter v voting in round $r+1$,

- if the ballot found by using strategy Z includes *all* alternatives in $C(r)$ or *none* of them:
  - apply strategy T
- else:
  - apply strategy Z

It is worth pointing out that the approval strategies presented above (plus strategy A, defined in Figure 4.1) effectively use only the ordinal preferences of a voter. In other words, any cardinal preferences that impose the same preference order over the alternatives will result in identical ballots if any of these strategies is used. Therefore they can be used equally effectively whether preference input consists of cardinal utilities or a (perhaps partial) ordering of the alternatives.

In chapter 4 we will compare the effectiveness of these approval strategies according to different evaluation metrics.

Chapter 2

Manipulation (or, What You Will)

The developers [23] of Declared-Strategy Voting (DSV) elections posited that their election protocol would force rational voters to specify cardinal preferences *sincerely*, while still acting in the best interest of each voter at the moment that voter must pick an alternative. In this way they were attempting to avoid the possibility of *manipulation*, which is an unfortunately grossly overloaded term in voting literature. In section 2.1, we examine the nature of manipulation in voting systems, settling on a definition that suits our goals in this chapter.

The act of voting requires brain activity, and the Church–Turing hypothesis [18, 55, 33] says that activity can be captured by a program. That program P takes in basically three things: the election protocol, the voter's feelings about the alternatives and the voter's expectations concerning how everybody else will vote.

The protocol specifies the manner in which the election will be conducted: it is a mathematical function from a list of ballots to one or more winners. We assume that voters cannot (will not) act until the protocol has been set. With the protocol established, a voter will act based upon the other two factors: that voter's feelings about the alternatives, and that voter's expectation concerning how everybody else will vote.

If we examine only those two factors, then literature can be summarized as follows.

sincerity If P is based only on the voter's feelings, ignoring how others might vote, then P is acting *sincerely.* However, it is well known [30, 53] that acting so directly on the voter's feelings may result in a provably worse outcome for the voter in terms of who wins the election. In that sense, P is generally *irrational* and a better outcome is obtainable through *strategic* behavior.

irrationality If P is based only on how others vote, ignoring how the voter feels about the alternatives, then P is by definition *irrational.* Examples include bandwagon and underdog voting [22].

strategy If P takes into account both how the voter feels and how others might vote, then P is acting *strategically* (even though P may happen to output a ballot that is sincere in some sense). If P can be shown, by some objective measure perhaps similar to those explored in chapter 4, to obtain outcomes influenced favorably by the voter's preferences, then P can be said to be acting *rationally.*

In summary of DSV, most voting protcols compel voters to act strategically to be rational. DSV elections hope to cause voters to act *sincerely* to be rational—by moving the voter's insincere program into the DSV system itself.

2.1 Notions of manipulation

One of a democracy's goals is to conduct elections that are *fair*, in the sense that no individual has an *undue* influence on an election's outcome. Such influence is loosely called *manipulation*, and we next examine various mechanisms by which an individual (or group of collaborating individuals) can influence an election's outcome. Of course, every voter has the potential to influence an election's outcome; otherwise, voting would be a futile activity. In discussions of manipulation, it is therefore important to discern the nature of influence and to establish the conditions under which an individual's influence is unfair.

2.1.1 Election specification

Those who specify an election and its protocol have the ability to manipulate an election. For example, election officials could overtly exclude an alternative, either so that alternative cannot win or because, if included, that alternative would prevent the election of a more desirable alternative due to a vote-splitting effect [42]. They could also affect the outcome by determining the set of allowed voters [9], perhaps allowing only those who support the favored outcome or excluding those who support strong rival outcomes. More subtly, if the election officials have the chance to choose an election protocol, then one can be chosen so as to obtain (probabilistically or for certain) a given outcome [52], or at least one that is less likely to give a particular undesirable outcome.

2.1.2 Ballot choice of voters

More common in the literature is to identify the manipulation opportunities of voters themselves. Intuitively speaking, manipulation by a voter occurs when he or she changes a ballot in the expectation of effecting a superior outcome; in this research, we will use only this notion of manipulation. But when, more precisely, can manipulation be said to have occurred? We will consider two possibilities:

1. A voter or group of voters, using all available information, decides whether a specific alternative can be made to win with at least a given probability and, if so, votes in such a way to elect that alternative. (Such voters are acting *strategically* in the sense defined above.)
2. A voter submits an insincere ballot that results in an election outcome better for that voter than the one that would have resulted if he or she had voted sincerely. (Insincerity can be reckoned by weak sincerity or strong sincerity as defined in chapter 1.)

These two notions of manipulation address *strategy* and *insincerity*, respectively.

The first is useful to consider when a voter is deciding how to vote; if a corresponding decision problem is shown to be computationally hard, it may be reasonable to expect voters to default to voting sincerely. Decision problems of this sort will be investigated in the next section.

The second identifies when a voter has "gamed the system"; it is what many social-choice authors mean when they describe a voting system as manipulable. (By this definition, a voting system is *nonmanipulable* when any rationally strategic ballot is sincere.) One design motivation for DSV was to encourage the submission of sincere preferences. In particular, it was hoped that ballot-by-ballot mode, by randomizing voter order, would deter the submission of insincere preferences; if a voter determines that insincere preferences will gain a superior outcome given one voter order, the same preferences may be unlikely to gain the same outcome given another voter order.

2.2 Manipulation decision problems

There have already been attempts to characterize the difficulty of manipulating voting systems. The following decision problem is a generalization of several in the literature.

> **Existence of Probably Winning Coalition Ballots (EPWCB)**
>
> INSTANCE: Set of alternatives A and a distinguished member a of A; set of weighted cardinal-ratings ballots B_V; the weights of a set of ballots B_U which have not been cast; probability $0 < \pi \leq 1$
> QUESTION: Does there exist a way to cast the ballots B_U so that a has at least probability π of winning the election with the ballots $B_V \cup B_U$?

EPWCB is perhaps best explained by presenting its interesting subproblems.

> **Existence of a Winning Ballot (EWB)**
>
> INSTANCE: Set of alternatives A and a distinguished member a of A; set of cardinal-ratings ballots B
> QUESTION: Does there exist a way to cast a ballot b_0 so that a wins the election with the ballots $B \cup \{b_0\}$ outright?

EWB is identical to the decision problem Existence of a Winning Preference (EWP) presented and analyzed by Bartholdi, Tovey and Trick [7], except that EWP uses ordinal ballots, standard for the literature in this area. EWB essentially looks at a manipulation opportunity from the point of view of a DSV voter coming last in the voter order: All other ballots are considered to be cast, fixed and known, and the question is whether there is a ballot that the ultimate voter can cast to cause the election of a certain alternative. The assumed situation is very much like that of a DSV program computing the final ballot of a round on behalf of its voter. Bartholdi *et al.*'s reasoning for using EWP is that it assumes all relevant information is available—if one can show that, for a given voting system, EWP is computationally hard, then manipulating that voting system must be hard when less information is available.

Bartholdi, Tovey and Trick are able to show that a polynomial-time algorithm exists for solving EWP in general for a large class of voting protocols, which includes plurality and approval voting. However, they present a protocol they call Second-Order Copeland for which solving EWP is NP-hard, and in a later paper Bartholdi and Orlin [8] show that EWP is also NP-hard for the single transferable vote (STV) in its single-winner version—also known as Hare, Instant Runoff Voting (IRV) or the alternative vote. But when the number of alternatives is held constant, a polynomial-time algorithm solving EWP exists even for these protocols, so EWP's NP-hardness depends on a large slate of alternatives.

> **Existence of Winning Coalition Ballots (EWCB)**
>
> INSTANCE: Set of alternatives A and a distinguished member a of A; set of weighted cardinal-ratings ballots B_V; the weights of a set of ballots B_U which have not been cast
> QUESTION: Does there exist a way to cast the ballots B_U so that a wins the election outright?

EWCB is Constructive Coalitional Weighted Manipulation (CCWM), introduced by Conitzer and Sandholm [20], with cardinal instead of ordinal ballots as input. EWCB is a generalization of EWB that, surprisingly, is NP-hard for many protocols even given a constant number of alternatives.

Note that EWB and EWCB ask whether a specific alternative can be made to win. The more general and perhaps more intuitively useful problem of finding a voter's most-liked alternative that can be made to win is not significantly harder: one would simply test each alternative in decending order of cardinal preference; when alternative a can be made to win, alternatives not preferred to a need not be considered.

Conitzer and Sandholm [21] described three "tweaks" that, when added to a voting protocol, such as plurality or the Borda count [51], make that protocol computationally hard to manipulate. All three use a "preround" to determine a set of alternatives to be eliminated before the voting protocol is executed. The simplest, called a deterministic preround, publishes a pairing of the alternatives before the ballots are collected; for each pair, the loser of the pairwise comparison between them according to the ballots is eliminated—if the number of alternatives is odd, one alternative survives without a comparison—and the original protocol, such as plurality or Borda, is executed on the ballots over the remaining alternatives. Adding this deterministic preround to a large class of protocols, including plurality and Borda, renders them NP-hard to manipulate in the sense of EWP.

The second preround tweak pairs the alternatives randomly after the ballots have been collected; it makes a protocol #P-hard to manipulate, but in a special sense: Instead of asking whether a specific alternative a can be made to win, as in EWP, it asks whether a can be made to win with some given probability $0 < \pi \leq 1$.

Existence of a Probably Winning Ballot (EPWB)

INSTANCE: Set of alternatives A and a distinguished member a of A; set of cardinal-ratings ballots B; probability $0 < \pi \leq 1$

QUESTION: Does there exist a way to cast a ballot b_0 so that a has at least probability π of winning the election with the ballots $B \cup \{b_0\}$?

Perhaps the nondeterminism of DSV's ballot-by-ballot mode has a similar effect on manipulation, either making it computationally difficult in this probabilistic sense or making it so that any manipulation that would work given one voter order would backfire for some other voter order.

Recall from section 1.3 that the Gibbard–Satterthwaite theorem proves that any voting protocol, including a meta-voting system like DSV, that treats ballots and alternatives symmetrically is manipulable by strategic voters, and, further, that the two protocols (random ballot and random runoff) most often given as examples to which the Gibbard–Satterthwaite theorem does not apply have a large nondeterministic component. Could it be that there is generally a tradeoff between manipulability and determinism?

2.3 Strategic insincerity and DSV

One motivation for the design of DSV was to elicit the submission of sincere preferences by having the system strategize for the voter. The hope is that the voter's declared strategy will cast ballots in such a way that the eventual outcome of the election is optimized from the voter's point of view, giving the voter no reason to try to gain a better outcome by submitting insincere preferences. In other words, one hopes that the declared strategy will do everything in its power to make the winner of the election as good as possible so that submitting false preferences can only backfire.

Unfortunately, the Gibbard–Satterthwaite theorem dashes hope that any reasonable voting system will be found that is immune to strategic insincerity in all voting situations with at least three alternatives. So even DSV, assuming no bias towards some voters or some alternatives is built in, will have cases with opportunities for manipulation by submitting insincere cardinal preferences.

Different DSV systems—those with different underlying voting protocols and in different modes—may in practice present manipulation opportunities to voters more or less often. For example, it may be much easier to find an election example where plurality DSV rewards the submission of insincere cardinal preferences than for approval DSV. But even if DSV is sometimes manipulable, one might expect it to be generally difficult to find the preferences that would manipulate successfully.

2.3.1 An NP-hard result

As it turns out, DSV can be shown to be computationally hard to manipulate in a certain sense. EWB, the version of Bartholdi's and Orlin's [8] EWP with cardinal-preference ballots, captures the relevant notion of manipulability. They showed that EWP is NP-hard for Hare, the single-winner form of STV.

Hare takes ordinal ballots as input. The ballots are counted in a series of elimination rounds. In the first round, only the top-rank votes are counted. The alternative with the smallest top-rank total is eliminated from the ballots, possibly adding to other alternatives' top-rank totals. This step is repeated until exactly one winner is left.

If a specific choice of declared strategy is forcibly made for all voters, DSV in batch mode with plurality as the underlying voting protocol can be made to simulate Hare. If the imposed declared strategy is carefully chosen, DSV can be made to select the winner that Hare would given the ranked ballots corresponding to the voters' expressed cardinal preferences. So if voters are not free to choose their own declared strategies, DSV can be made NP-hard to manipulate.

Theorem 2.3.1. *If a declared strategy can be imposed on the voters, so that they submit only their cardinal preferences over the alternatives, DSV can be made to be NP-hard to manipulate in the EWB sense.*

Proof. We will always elect the Hare winner according to the ordinal ballots implied by the voters' cardinal preferences if we use DSV in batch mode with one-vote plurality as the underlying protocol and the following strategy for all voters:

$$\text{vote for alternative } i \text{ such that } p_i = \max(p_j : s_j \geq t)$$

where

$$t = \begin{cases} \min(U \setminus \{\min(U)\}) & \text{if} \quad |U| > 1 \\ \min(U) & \text{if} \quad |U| = 1 \\ 0 & \text{if} \quad |U| = 0 \end{cases}$$

and

$$U = \{s_j : s_j > 0\}$$

In the first round of the batch DSV election, when $(\forall i)\ s_i = 0$, the imposed strategy will vote for each voter's favorite alternative, just as in the first round of counting a Hare election. In subsequent DSV rounds, as long as $|U| > 1$ (there remain more than one uneliminated alternative), all voters vote for their most preferred alternative with a vote total at least t, a threshold set at the second-smallest vote total, effectively eliminating the alternative(s) with the lowest vote totals among the uneliminated alternatives. When $|U| = 1$ (only one alternative has a nonzero vote total), every voter votes for that alternative. The resulting election state does not change afterward, and that alternative, which is the Hare winner, thus wins the DSV election.

Bartholdi and Orlin [8] proved that Hare (the single-winner version of STV) is NP-hard to manipulate in the Existence of a Winning Preference (EWP) sense. EWP is a subproblem of Existence of a Winning Ballot (EWB), so Hare is NP-hard to manipulate in the EWB sense as well. Since batch DSV with the above imposed strategy is equivalent to Hare, DSV can be made NP-hard to manipulate in the EWB sense. □

So DSV is NP-hard to manipulate in the general case. But what does this mean? It means only that there is no algorithm that runs in time polynomial in the number of alternatives and the number of voters that is guaranteed to find a set of preferences that guarantees a given outcome. It does *not* mean that DSV is necessarily hard to manipulate in every case. There may be a simple heuristic that often (if not always) finds preferences that will manipulate successfully, or one that tends to lead to a better outcome than blind sincerity. It also does not imply that manipulating DSV is necessarily easy or hard when declared strategies can be freely chosen.

2.4 Generalizing hardness results to approval voting

Conitzer and Sandholm [19] showed that CCWM is in P for plurality for any constant number of alternatives but NP-hard for Borda and veto voting with three or more alternatives. (For every voting system under consideration, CCWM is in P when the number of alternatives is limited to two.) CCWM takes ranked ballots as input, so approval voting cannot be applied to CCWM. But EWCB works with approval voting; it is simply CCWM with cardinal-preference input. EWCB can be seen to be in P for approval voting by using the same manipulation algorithm that works for plurality: Approve on all ballots only a, the distinguished alternative that is to be made to win. There is a way to make a win if and only if this strategy makes a win.

2.5 Summary of contributions

In this research, we have accomplished the following.

1. Proved that manipulating DSV in general is NP-hard by describing a plurality DSV system that imposes a specified declared strategy on all voters in such a way that the Hare winner is elected.

2. Provided a polynomial-time algorithm for solving EWCB for approval voting.

Chapter 3

DSV and Approval-Rating Polls

We see DSV as a way to reduce or eliminate manipulation by voters' insincerity by embracing manipulation itself. By strategizing for the voter, a "perfect" DSV system would encourage sincere indication of preferences. In the next chapter we will investigate questions regarding the manipulability of DSV in the general case, where it is used to elect one of a static set of alternatives over which voters may hold any preferences. But the DSV framework can be used in other contexts. In particular, if certain assumptions are made about the available alternatives and the voters' preferences among them, stronger results regarding DSV are possible.

In this chapter we will investigate applying DSV to the problem of selecting a number from a specified range of numbers. We will show that this new application of DSV achieves the original goal of DSV: It eliminates the possibility of manipulation by making insincere voting no more effective than sincere voting. Intuitively, this is possible because we assume that voters can only have certain kinds of preferences over the possible outcomes, effectively reducing the space of possible elections. For example, if the alternatives consist of the rational numbers between 0 and 1, we can justifiably assume each voter's preferences over the considered range to be single-peaked. In other words, we can assume that no voter will prefer a over b and c over b if $a < b < c$.

This application of DSV may have fewer uses in real-world elections than a more general DSV system that allows arbitrary voter preferences over alternatives, but it can be seen as an illustration of the power of DSV in principle and an étude for the further study of DSV.

3.1 Approval ratings and their aggregation

Approval ratings are one mechanism that communities can use to offer incentive and reward for good behavior or service. The prospect of feedback following a given interaction presumably increases the accountability of that interaction for all parties involved. Publication of approval ratings then enables appropriate consequences to follow from positive or negative experiences.

It is interesting, however, to consider the form in which approval ratings can and should be published. While the greatest detail is afforded by publication of each participant's response to an approval rating poll, the resulting volume is typically unacceptable for the purposes of summarizing an electorate's experience. Thus, some form of aggregation is typically performed on approval ratings, and the result of that aggregation is then announced as the result of the poll.

In this chapter we consider several forms of aggregation and we show that some methods can reward insincerity while others cannot. We next provide several examples of approval rating systems and formulate a general form of an approval rating poll.

3.1.1 Examples of approval rating polls

Subscribers and observers of media frequently learn of the results of *approval rating polls* that attempt to discern how strongly a participating electorate endorses a person or a position of interest. For example, approval ratings concerning the performance of the United States President are published throughout a presidency; following events or policy decisions that affect an electorate, such polls are often conducted as a means of evaluating the electorate's support for the President's actions.

As another example, several web sites post varous forms of approval ratings for movies and games. Specifically, Rotten Tomatoes [2] posts the results of two polls for each movie:

- Each review from a set of accredited critics is turned into approval (fresh rating) or disapproval (rotten rating) of the reviewed material, in terms of whether the material merits viewing. The percentage of fresh reviews is reported as the movie's *Tomatometer*. In effect, each review is turned into a 0 or 1 value, and the Tomatometer is the average of those values expressed as a percentage. Putative viewers might consult a movie's Tomatometer value to determine whether they should see that movie.
- Each critic can also rate a movie's overall quality on a 1–10 scale. Rotten Tomatoes then publishes the average of all such ratings. Similarly, Metacritic [1] computes a weighted average of its accredited reviewers' approval ratings for a given movie, supplied on a 0–100 scale.

Finally, consider the electronic marketplace, in which participants are asked to rate the honesty and effectiveness of merchants and customers. Sites such as eBay poll their participants concerning how strongly they approve of the behavior of the marketplace members they encounter in transactions. Upon completion of a transaction, the involved parties are asked to rate each other. An aggregation of an indvidual's approval ratings is posted for public view, so that members can consider such information before engaging that individual in a transaction.

Based on the above, some approval ratings are formulated more incrementally than others. For example, the ratings published by Rotten Tomatoes and Metacritic are collected and then analyzed *en masse*, while the approval rating of an eBay participant (merchant or customer) can be updated after every interaction involving that participant. As we show below, knowledge concerning how others approve of a given issue can influence a particpant's expressed approval rating.

3.1.2 Formulation

We next define a general instance of an approval rating poll to facilitate presentation of our results.

- An electorate of n participants is polled. Based on the participants' response and the aggregation protocol at hand, the result of the poll will be published as a rational number in the interval $[0, 1]$.

- Each participant i has in mind a sincere preference rating $r_i, 0 \leq r_i \leq 1$ that can be construed as that participant's dictatorial preference. The tuple of all participants' sincere ratings is denoted by the vector $\vec{r}$.

  We further make the reasonable assumption that voter i's preferences are single-peaked and non-plateauing: a voter's utilities for the outcomes are monotonically decreasing when moving away from r_i in either direction. It follows, for example, that any voter that prefers 0.2 to 0.5 must also prefer 0.5 to 0.8. More formally:

  - If $a < b \leq r_i$, then voter i must prefer b to a;
  - If $r_i \leq c < d$, then voter i must prefer c to d.

  Based on the above, r_i sufficiently characterizes a voter's outcome utilities for our purposes.

- Each participant i has also in mind a probability density function p_i that models the probabilistic outcome of the poll, excluding i's rating. For the purposes of this chapter, the outcome from i's point of view is simply an expected value o_i. While a more general treatment could be the subject of future work, we therefore assume p_i is the Dirac δ function:

$$p_i = \delta(t - o_i) = \begin{cases} \infty & \text{if} \quad t = o_i \\ 0 & \text{if} \quad t \neq o_i \end{cases}$$

  with the area under p_i summing by definition to unity.

  - In situations where preference data accrues incrementally and the poll's results are updated continually, o_i is readily available before the ith participant expresses approval. Such is the case in eBay when a buyer provides an approval rating for a merchant.
  - In other cases, preliminary polls or other information sources may provide sufficient information to provide a likely value for o_i.

  While estimations of o_i could be inaccurate, faulty or based on purposefully falsified information, the presence of such information can affect an electorate as discussed below.

- Finally, voter i participates in the poll by expressing a rating preference of v_i, which may or may not be the same as r_i. In fact, we are particularly interested in situations where $v_i \neq r_i$ due to p_i. For example, the expression of an individual's approval rating could well be affected by knowledge (perceptions, estimations, or actualities) of how others approve of the issue at hand. For example, consider an eBay customer who undertakes a transaction with a highly approved merchant. If the customer becomes disgruntled with the merchant, then the customer's resulting rating of the merchant might be overly negative, precisely because of the merchant's otherwise high rating.

  The tuple of all expressed approval ratings is denoted by the vector $\vec{v}$.

This chapter considers an approach that can account for, mitigate, or prevent the use of insincerity to increase a participant's effectiveness in an approval rating poll.

3.1.3 Aggregating approval ratings

The results of an approval rating poll are typically reported by an aggregation procedure that is disclosed *a priori*. In this section, we consider two popular aggregation schemes: average and median.

Average aggregation Here, the result of the approval rating poll is computed as the average of the participants' expressed approval ratings:

$$\bar{v} = \frac{\sum_{j=1}^{n} v_j}{n}$$

While the Average aggregation function is sensitive to each voter's input, it has an important disadvantage: Voters can often gain by voting insincerely. For example, the 1983 film *Videodrome* has five critics' ratings on Metacritic. If we assume that these critics rated the film sincerely (that each would prefer that the average rating of the film be his or her rating), we have

$$\vec{r} = [0.4, 0.7, 0.8, 0.8, 0.88]$$

If these preferences are actually expressed sincerely in an Average aggregation context, then we have

$$\vec{v} = [0.4, 0.7, 0.8, 0.8, 0.88]$$

and the Average aggregation yields 0.716.

Consider voter 5, whose ideal outcome is $r_5 = 0.88$. That voter could achive a better outcome by *not* expressing the sincere preference $v_5 = 0.88$ and instead voting $v_5 = 1$. The resulting Average aggregation yields the outcome 0.74, which, being closer to 0.88, is preferred by voter 5 to 0.716.

Median aggregation (n odd) Another possible aggregation function computes a *median* of $\vec{v}$: $\tilde{v}$ is a value that satisfies

$$|\{i : \tilde{v} < v_i\}| \leq \frac{n}{2} \quad \text{and} \quad |\{i : \tilde{v} > v_i\}| \leq \frac{n}{2}$$

or, equivalently,

$$|\{i : \tilde{v} < v_i\}| \leq \frac{n}{2} \leq |\{i : \tilde{v} \leq v_i\}|$$

The above definition does not necessarily prescribe a unique aggregation when n is even; we address this issue below.

According to the *median voter theorem* [10, 24], when n is odd, Median aggregation becomes the unique, Condorcet-compliant [38] rating system, yielding a result that is preferred by some majority of voters to every other outcome.

Unfortunately, Median aggregation can effectively ignore almost half of the voters. In other words, majority rule can mean majority tyranny. Given the following tuple of votes

$$\vec{v} = [0, 0, 0, 1, 1]$$

the 1-voters are effectively ignored when the median, 0, is chosen as the outcome. Majority tyranny could be quite undesirable for polls of this type, especially when the goal of aggregating ratings is to represent a satisfactory consensus for all voters. The Average outcome of the above tuple, 0.4, arguably provides such a much better consensus.

In contrast with Average aggregation, Median aggregation is nonmanipulable by insincere voters—at least when n is odd: a voter i can never improve the outcome from his or her point of view by voting $v_i \neq r_i$.

Theorem 3.1.1. *When n is odd, each voter i obtains his or her best outcome by voting $v_i = r_i$.*

Proof. Consider the relation of any voter i's sincere preference r_i to the Median outcome $\tilde{v}$, with the following three cases:

- $r_i = \tilde{v}$. With i's sincere preference as the outcome, no better result could obtain by changing r_i.
- $r_i < \tilde{v}$. Because n is odd, the median vote is uniquely determined. Thus, decreasing r_i cannot affect $\tilde{v}$; increasing r_i could only increase $\tilde{v}$, which would produce a less desirable outcome for voter i.
- $r_i > \tilde{v}$. A symmetric argument based on the above holds here as well.

$\square$

Thus, Median aggregation does not reward insincerity for an odd number of participants.

Median aggregation (generalized) The conventional method in statistics for computing the Median of an even number of values is to compute the average of the middle two values. In such a situation, the voter who cast one of those two values could pull the outcome in a beneficial direction by voting insincerely.

Fortunately, there are many methods to eliminate such manipulation; examples include the following:

- One of the two middle values could be chosen at random.
- If 0.5 lies between the two middle values, then 0.5 is chosen; otherwise, the one of the two that is nearer 0.5 is chosen.

Note that the outcome $\tilde{v}$ given by any of these median functions minimizes $\sum_i |v_i - \tilde{v}|$, in contrast to the average function, $\bar{v}$, which minimizes $\sum_i (v_i - \bar{v})^2$.

Without losing nonmanipulability, the Median function can be generalized to give the outcome

$$^b\tilde{v} \quad \text{where} \quad |\{i : {}^b\tilde{v} < v_i\}| \leq bn \leq |\{i : {}^b\tilde{v} \leq v_i\}|$$

for any $0 \leq b \leq 1$. (In this notation, the b is intended as a parameter modifying the tilde.) If bn is an integer, there may be more than one $0 \leq \phi \leq 1]$ that satisfies

$$|\{i : \phi < v_i\}| \leq bn \leq |\{i : \phi \leq v_i\}|$$

In that case, define Φ as the set of all such ϕ. Then

$$^b\tilde{v} \equiv \begin{cases} \min(\Phi) & \text{if} \quad b < \min(\Phi) \\ b & \text{if} \quad \min(\Phi) \leq b \leq \max(\Phi) \\ \max(\Phi) & \text{if} \quad \max(\Phi) < b \end{cases}$$

or, equivalently,

$$^b\tilde{v} \equiv \begin{cases} \min(\Phi) & \text{if} \quad (\forall \phi \in \Phi)\ \phi > b \\ b & \text{if} \quad b \in \Phi \\ \max(\Phi) & \text{if} \quad (\forall \phi \in \Phi)\ \phi < b \end{cases}$$

This order-statistic outcome equals $\max(\vec{v})$ when $b = 0$, the third quartile when $b = \frac{1}{4}$, the Median outcome when $b = \frac{1}{2}$, the first quartile when $b = \frac{3}{4}$ and $\min(\vec{v})$ when $b = 1$.

Summary For an approval-rating poll, the choice of aggregation mechanism affects the nature of the outcome and the reward for voter insincerity. The Average aggregation outcome can reward insincerity, but the outcome provides a reasonable consensus of the electorate. On the other hand, Median aggregation does not reward insincerity, but it allows for tyranny by a majority.

3.2 Rationally optimal strategy for Average aggregation

As shown in section 3.1, Average aggregation can reward insincerity. In this section, we develop a *rationally optimal* strategy: a computation by which a voter can achieve a result as close as possible to that voter's prefered outcome. As before, we assume an eletorate in which n voters will express preferences. We begin by considering a rationally optimal strategy from the perspective of a final, omniscient voter. We then consider the behavior of a system in which all voters use a rationally optimal strategy.

To facilitate exposition and analysis of our results, we begin by generalizing the scale on which preferences are expressed as follows. In an $[m, M]$-Average poll, voters are allowed to *express* preference ratings in the interval $[m, M], m \leq 0, 1 \leq M$. We continue to assume that *sincere* preference ratings are in the interval $[0, 1]$; the expanded range is therefore intended to allow voters more room to manipulate the outcome. We also assume that preferences are aggregated by computing the Average of the voters' expressed preferences.

3.2.1 Strategy for a final, omnisicent voter

Consider a $(-\infty, +\infty)$-Average poll in which voter v_n is the last voter to express an approval rating, and in which all other voters vote their sincere preference ratings: $(\forall i \neq n)\ v_i = r_i$. If voter n can see the expressed approval ratings of all voters, then the ideal outcome for voter n ($\bar{v} = r_n$) can be realized by voting

$$v_n = r_n n - \sum_{j \neq n} r_j$$

More generally, in an $[m, M]$-Average poll, voter n should express v_n to move the outcome as close to r_n as possible:

$$v_n = \min\left(\max\left(r_n n - \sum_{j \neq n} r_j, m\right), M\right) \tag{3.1}$$

The above is the *rationally optimal strategy* for voter n in an $[m, M]$-Average approval rating poll.

As an example, consider the $[0,1]$-Average system with sincere preferences from the *Videodrome* example above:

$$\vec{r} = [0.4, 0.7, 0.8, 0.8, 0.88]$$

After all other voters express their sincere preferences, v_5's rationally optimal preference rating is given by

$$\begin{aligned} v_5 &= \min\left(\max\left(r_5 n - \sum_{j \neq 5} r_j, 0\right), 1\right) \\ &= \min\left(\max\left(0.88 \cdot 5 - (0.4 + 0.7 + 0.8 + 0.8), 0\right), 1\right) \\ &= 1 \end{aligned} \tag{3.2}$$

achieving an outcome $\bar{v}$ of 0.74. No other choice for v_5 would achieve an outcome $\bar{v}$ closer to $r_5 = 0.88$.

After voter n has voted using Equation 3.1, exactly one of the following is true.

1. $\bar{v} < r_n$ and $v_n = M$
2. $\bar{v} = r_n$
3. $\bar{v} > r_n$ and $v_n = m$

In each case, either voter n's ideal outcome r_n has been realized, or voter n has moved the outcome as close to r_n as is immediately possible. Based on the three cases above, no other choice for r_n has that property.

Moreover, in each of the above three cases, $\bar{v} \in [0,1]$ even though $v_n \in [m, M]$. Recall that each sincere preference, including r_n, is in the interval $[0,1]$. In case (1), we have $\bar{v} < r_n \leq 1$. Thus we need only show $0 \leq \bar{v}$: Since $\bar{v}$ is computed as the average of $n-1$ numbers in the interval $[0,1]$ and one number $(v_n = M) \geq 1$, we obtain $0 \leq \bar{v}$. A symmetric argument holds for case (3). Case (2) follows directly since $\bar{v} = r_n$.

3.2.2 Equilibrium for n strategic voters

We have thus far allowed only voter n to use a rationally optimal strategy, requiring all other voters to express their sincere approval ratings. We now consider the properties of the more practical $[m, M]$-Average system in which each voter i uses a rationally optimal strategy to compute an expressed approval rating, based on i's sincere approval rating r_i and on the expressed votes of all other voters. When each voter i establishes v_i, other voters may wish to update their expressed approval ratings.

Returning to the *Videodrome* example, in which minimum vote $m = 0$ and maximum vote $M = 1$, we again have sincere preferences

$$\vec{r} = [0.4, 0.7, 0.8, 0.8, 0.88]$$

Hypothetically, let us say initial votes are assumed to be sincere:

$$\vec{v} = [0.4, 0.7, 0.8, 0.8, 0.88]$$

Then we allow all voters to revise their votes independently and simultaneously. Voter 2 deliberates:

$$\begin{aligned} v_2 &= \min\left(\max\left(r_2 n - \sum_{j \neq 2} r_j, 0\right), 1\right) \\ &= \min\left(\max\left(0.7 \cdot 5 - (0.4 + 0.8 + 0.8 + 0.88), 0\right), 1\right) \\ &= 0.62 \end{aligned}$$

and changes v_2 to 0.62; voter 5 decides to change v_5 to 1, the optimal strategy as calculated above (Equation 3.2). Similarly, v_3 and v_4 become 1 and v_1 becomes 0. The resulting vote vector is

$$\vec{v} = [0, 0.62, 1, 1, 1]$$

which is not an *equilibrium*—from this state, voter 2 in particular, in response to the other changed votes, would prefer to change v_2 again according to the optimal strategy. If the voters are given chances to re-revise their votes, again independently, then we have

$$\vec{v} = [0, 0.5, 1, 1, 1]$$

and finally reach an state from which no voter i would change v_i according to the optimal strategy. In this case the equilibrium is unique, giving a final outcome of $\bar{v} = 0.7$. Notice that voter 2, the only voter to vote in between the allowed extremes, achieved the ideal outcome of $\bar{v} = r_2$; all other voters are voting at the extremes in a vain attempt to pull the outcome in the desired direction.

Alternatively, let us again assume initial votes to be sincere:

$$\vec{v} = [0.4, 0.7, 0.8, 0.8, 0.88]$$

Then we allow voters to revise their votes in order, from voter 5 down to voter 1. First, voter 5 decides to change v_5 to 1 (Equation 3.2). Then voter 4 deliberates:

$$\begin{aligned} v_4 &= \min\left(\max\left(r_4 n - \sum_{j \neq 4} r_j, 0\right), 1\right) \\ &= \min\left(\max\left(0.8 \cdot 5 - (0.4 + 0.7 + 0.8 + 1), 0\right), 1\right) \\ &= 1 \end{aligned}$$

and changes v_4 to 1. The voters then in turn change v_3 to 0.9, v_2 to 0.2 and v_1 to 0. The resulting vote vector is

$$\vec{v} = [0, 0.2, 0.9, 1, 1]$$

which is not an equilibrium—from this state, both voters 2 and 3 would prefer to change their votes again according to the optimal strategy. If the voters are given chances to re-revise their votes, again from voter 5 down to 1, then we have

$$\vec{v} = [0, 0.2, 0.9, 1, \mathbf{1}]$$

$$\vec{v} = [0, 0.2, 0.9, \mathbf{1}, 1]$$

$$\vec{v} = [0, 0.2, \mathbf{1}, 1, 1]$$

$$\vec{v} = [0, \mathbf{0.5}, 1, 1, 1]$$

$$\vec{v} = [\mathbf{0}, 0.5, 1, 1, 1]$$

and finally reach the same equilibrium as before.

While there are many possible schemes that could accommodate iterative changes in expressed preferences, we examine the more general issue of reaching an equilibrium: each voter i has arrived at an expressed preference v_i such that the rationally optimal strategy recommends no change in v_i:

$$(\forall i)\ v_i = \min\left(\max\left(r_i n - \sum_{j \neq i} v_j, m\right), M\right) \tag{3.3}$$

So, at equilibrium,

$$(\forall i)\ (\bar{v} < r_i \wedge v_i = M) \vee (\bar{v} = r_i) \vee (\bar{v} > r_i \wedge v_i = m)$$

and it follows that

$$(\forall i)\ \bar{v} < r_i \longrightarrow v_i = M \tag{3.4}$$

and

$$(\forall i)\ \bar{v} > r_i \longrightarrow v_i = m \tag{3.5}$$

Equation 3.4 says that for every i such that $\bar{v} < r_i$, $v_i = M$. So we can place a lower bound on the sum of all v_is by assuming all other v_is are at the minimum:

$$m \cdot |\{i : \bar{v} \geq r_i\}| + M \cdot |\{i : \bar{v} < r_i\}| \leq \sum_{i=1}^{n} v_i = \bar{v}n$$

Similarly, Equation 3.5 says that for every i such that $\bar{v} > r_i$, $v_i = m$. So we can place an upper bound on the sum of all v_is by assuming all other v_is are at the maximum:

$$\bar{v}n = \sum_{i=1}^{n} v_i \leq m \cdot |\{i : \bar{v} > r_i\}| + M \cdot |\{i : \bar{v} \leq r_i\}|$$

So we have

$$m \cdot |\{i : \bar{v} \geq r_i\}| + M \cdot |\{i : \bar{v} < r_i\}| \leq \bar{v}n \leq m \cdot |\{i : \bar{v} > r_i\}| + M \cdot |\{i : \bar{v} \leq r_i\}|$$

Subtracting mn,

$$m \cdot |\{i : \bar{v} \geq r_i\}| + M \cdot |\{i : \bar{v} < r_i\}| - mn \leq \bar{v}n - mn \leq m \cdot |\{i : \bar{v} > r_i\}| + M \cdot |\{i : \bar{v} \leq r_i\}| - mn$$

and

$$m \cdot (|\{i : \bar{v} \geq r_i\}| - n) + M \cdot |\{i : \bar{v} < r_i\}| \leq (\bar{v} - m)n \leq m \cdot (|\{i : \bar{v} > r_i\}| - n) + M \cdot |\{i : \bar{v} \leq r_i\}|$$

Since $|\{i : \bar{v} < r_i\}| + |\{i : \bar{v} \geq r_i\}| = |\{i : \bar{v} \leq r_i\}| + |\{i : \bar{v} > r_i\}| = n$,

$$m \cdot (-|\{i : \bar{v} < r_i\}|) + M \cdot |\{i : \bar{v} < r_i\}| \leq (\bar{v} - m)n \leq m \cdot (-|\{i : \bar{v} \leq r_i\}|) + M \cdot |\{i : \bar{v} \leq r_i\}|$$

and

$$(M - m) \cdot |\{i : \bar{v} < r_i\}| \leq (\bar{v} - m)n \leq (M - m) \cdot |\{i : \bar{v} \leq r_i\}|$$

And since $M > m$,

$$|\{i : \bar{v} < r_i\}| \leq \frac{\bar{v} - m}{M - m} n \leq |\{i : \bar{v} \leq r_i\}|$$

Thus any average at equilibrium must satisfy the two equations

$$|\{i : \bar{v} < r_i\}| \leq \frac{\bar{v} - m}{M - m} n \tag{3.6}$$

and

$$\frac{\bar{v} - m}{M - m} n \leq |\{i : \bar{v} \leq r_i\}| \tag{3.7}$$

3.3 Multiple equilibria can exist

For some sincere-ratings vectors $\vec{r}$, multiple equilibria exist: There exist more than one $\vec{v}$ satisfying Equation 3.3. For example, if minimum vote $m = 0$, maximum vote $M = 1$ and

$$\vec{r} = [0.4, 0.7, 0.7, 0.8, 0.88]$$

(a slight tweak to the *Videodrome* example) then each of the following vectors satisfies Equation 3.3.

- $\vec{v} = [0, 0.5, 1, 1, 1]$
- $\vec{v} = [0, 0.6, 0.9, 1, 1]$
- $\vec{v} = [0, 0.75, 0.75, 1, 1]$

In fact, any $\vec{v} = [0, v_2, v_3, 1, 1]$ where $v_2 + v_3 = 1.5$ represents an equilibrium from which the optimal strategy would change no voter's vote.

In this case, at each possible equilibrium the outcome is $\bar{v} = 0.7$ (the ideal outcome of the two voters "conspiring" to keep it there). This is no coincidence; in general, it turns out that, even when multiple equilibria exist, the average at equilibrium is unique.

3.4 At most one equilibrium average rating can exist

We have seen that, given a length-n vector $\vec{r}$ of sincere ratings where $0 \leq r_i \leq 1$ for $1 \leq i \leq n$, any equilibrium $\vec{v}$ that results from every voter's using the optimal strategy will have a $\phi = \bar{v}$ that satisfies the inequalities

$$|\{i : \phi < r_i\}| \leq \frac{\phi - m}{M - m} n \tag{3.8}$$

and

$$\frac{\phi - m}{M - m} n \leq |\{i : \phi \leq r_i\}| \tag{3.9}$$

It turns out that at most one such ϕ exists for a given $\vec{r}$:

Theorem 3.4.1. *Given a vector $\vec{r}$ of length n where $0 \leq r_i \leq 1$ for $1 \leq i \leq n$,*

$$\begin{aligned} &|\{i : \phi_1 < r_i\}| \leq \tfrac{\phi_1 - m}{M - m} n \leq |\{i : \phi_1 \leq r_i\}| \quad \wedge \\ &|\{i : \phi_2 < r_i\}| \leq \tfrac{\phi_2 - m}{M - m} n \leq |\{i : \phi_2 \leq r_i\}| \quad \longrightarrow \quad \phi_1 = \phi_2 \end{aligned}$$

Proof. The proof considers two symmetric cases, $\phi_1 < \phi_2$ and $\phi_2 < \phi_1$, and shows by contradiction that each is impossible.

Case 1. Assume that $\phi_1 < \phi_2$. Then

$$(\forall i)\ \phi_2 \leq r_i \longrightarrow \phi_1 < r_i$$

Therefore

$$\{i : \phi_2 \leq r_i\} \subseteq \{i : \phi_1 < r_i\}$$

and then

$$|\{i : \phi_2 \leq r_i\}| \leq |\{i : \phi_1 < r_i\}|$$

We know that $\frac{\phi_2 - m}{M - m} n \leq |\{i : \phi_2 \leq r_i\}|$ and $|\{i : \phi_1 < r_i\}| \leq \frac{\phi_1 - m}{M - m} n$. So

$$\frac{\phi_2 - m}{M - m} n \leq |\{i : \phi_2 \leq r_i\}| \leq |\{i : \phi_1 < r_i\}| \leq \frac{\phi_1 - m}{M - m} n$$

Since n is positive, we have

$$\frac{\phi_2 - m}{M - m} \leq \frac{\phi_1 - m}{M - m}$$

and, since $M > m$,

$$\phi_2 - m \leq \phi_1 - m$$

and $\phi_2 \leq \phi_1$, contradicting the assumption that $\phi_1 < \phi_2$. Therefore the assumption must be false, and $\phi_1 \not< \phi_2$.

Case 2. Assume that $\phi_2 < \phi_1$. Then, for each i, $\phi_1 \leq r_i$ implies $\phi_2 < r_i$. Therefore $\{i : \phi_1 \leq r_i\} \subseteq \{i : \phi_2 < r_i\}$ and then

$$|\{i : \phi_1 \leq r_i\}| \leq |\{i : \phi_2 < r_i\}|$$

We know that $\frac{\phi_1 - m}{M - m} n \leq |\{i : \phi_1 \leq r_i\}|$ and $|\{i : \phi_2 < r_i\}| \leq \frac{\phi_2 - m}{M - m} n$. So

$$\frac{\phi_1 - m}{M - m} n \leq |\{i : \phi_1 \leq r_i\}| \leq |\{i : \phi_2 < r_i\}| \leq \frac{\phi_2 - m}{M - m} n$$

which gives us

$$\frac{\phi_1 - m}{M - m} \leq \frac{\phi_2 - m}{M - m}$$

and

$$\phi_1 - m \leq \phi_2 - m$$

and $\phi_1 \leq \phi_2$, contradicting the assumption that $\phi_2 < \phi_1$. Therefore the assumption must be false, and $\phi_2 \not< \phi_1$.

Conclusion. Since $\phi_1 \not< \phi_2$ and $\phi_2 \not< \phi_1$, it must be that $\phi_1 = \phi_2$. □

3.5 At least one equilibrium always exists

It does little good to show that all equilibria will have equal averages if an equilibrium does not always exist. Fortunately, for any set of sincere preferred outcomes $\vec{r}$, there will always be at least one equilibrium $\vec{v}$ such that no voter i would choose to change v_i according to the optimal Average strategy defined above.

We can show that a particular procedure will always find an equilibrium. Using the *Videodrome* example (with $m = 0$, $M = 1$) again for demonstration:

$$\vec{r} = [0.4, 0.7, 0.8, 0.8, 0.88]$$

This time, let us say initial votes are assumed to be, not sincere, but *zero* (the minimum allowed vote):

$$\vec{v} = [0, 0, 0, 0, 0]$$

Then we again allow voters to revise their votes in order, from voter 5 down to voter 1. (This particular order will prove significant.) First, voter 5 deliberates:

$$v_5 = \min\left(\max\left(r_5 n - \sum_{j \neq 5} r_j, 0\right), 1\right) = \min\left(\max\left(0.8 \cdot 5 - (0 + 0 + 0 + 0), 0\right), 1\right) = 1$$

and changes v_5 to 1. The voters then in turn reason similarly and change v_4 to 1, v_3 to 1, v_2 to 0.5 and v_1 to 0. The resulting vote vector,

$$\vec{v} = [0, 0.5, 1, 1, 1]$$

is indeed the same equilibrium found above in section 3.2.2, this time going through the voters only once.

This procedure inspires the following straightforward algorithm, which takes a $\vec{r}$ as input and outputs an equilibrium $\vec{v}$, assigning to each v_i exactly once. It orders the voters by decreasing r_i values, then uses the optimal strategy for each voter i in order, implicitly making the assumption that $v_j = m$ for $j > i$.

Algorithm 3.5.1.

FindEquilibrium$(\vec{r}, m, M)$:

sort $\vec{r}$ so that $(\forall i \leq j)\ r_i \geq r_j$

for $i = 1$ to n do

$v_i \leftarrow \min\left(\max\left(r_i n - \sum_{k<i} v_k - (n-i)m, m\right), M\right)$

return $\vec{v}$

Note that the algorithm assigns a value between m and M, inclusive, to each v_i exactly once, and that the assignment to v_i does not depend on the values of v_j where $j > i$. Therefore, after Algorithm 3.5.1 completes, it must be true that

$$(\forall i)\ v_i = \min\left(\max\left(r_i n - \sum_{k<i} v_k - (n-i)m, m\right), M\right)$$

but this is not enough to see that the resulting $\vec{v}$ is an equilibrium. To see that, we must show that

$$(\forall i)\ v_i = \min\left(\max\left(r_i n - \sum_{k\neq i} v_k, m\right), M\right)$$

To aid the proof that Algorithm 3.5.1 always reaches an equilibrium, we first prove a few lemmata that must hold true after the algorithm completes. The first intuitively says that the optimal strategy never recommends a vote v_i greater than m that would result in an average higher than the voter i's ideal outcome, with the assumption that the voters coming after voter i vote m:

Lemma 3.5.2. $(\forall i)\ v_i > m \longrightarrow r_i n \geq \sum_{k \leq i} v_k + (n-i)m$.

Proof. For any i, if

$$v_i = \min\left(\max\left(r_i n - \sum_{k<i} v_k - (n-i)m, m\right), M\right) > m$$

then it must be that

$$r_i n - \sum_{k<i} v_k - (n-i)m > m$$

Notice that whenever we have some x such that $x > m$, it must be true that $x \geq \min(\max(x, m), M)$. It follows that

$$r_i n - \sum_{k<i} v_k - (n-i)m \geq \min\left(\max\left(r_i n - \sum_{k<i} v_k - (n-i)m, m\right), M\right)$$

Therefore,

$$r_i n - \sum_{k<i} v_k - (n-i)m \geq v_i$$

and

$$r_i n \geq v_i + \sum_{k<i} v_k + (n-i)m = \sum_{k \leq i} v_k + (n-i)m$$

$\square$

Another lemma is substantially similar, saying that the optimal strategy never recommends a vote v_i less than M that would result in an average lower than the voter i's ideal outcome, with the assumption that the voters coming after voter i vote m:

Lemma 3.5.3. $(\forall i)\ v_i < M \longrightarrow r_i n \leq \sum_{k \leq i} v_k + (n-i)m$.

Proof. For any i, if

$$v_i = \min\left(\max\left(r_i n - \sum_{k<i} v_k - (n-i)m, m\right), M\right) < M$$

then it must be that

$$r_i n - \sum_{k<i} v_k - (n-i)m < M$$

Notice that whenever $x < M$ for some x, it must be true that $x \leq \min(\max(x, m), M)$. It follows that

$$r_i n - \sum_{k<i} v_k - (n-i)m \leq \min\left(\max\left(r_i n - \sum_{k<i} v_k - (n-i)m, m\right), M\right)$$

Therefore,

$$r_i n - \sum_{k<i} v_k - (n-i)m \leq v_i$$

and

$$r_i n \leq v_i + \sum_{k<i} v_k + (n-i)m = \sum_{k\leq i} v_k + (n-i)m$$

□

Finally, whenever Algorithm 3.5.1 assigns a value greater than the minimum m to a vote v_i, it must have assigned the maximum vote M to all v_j where $j < i$:

Lemma 3.5.4. $v_i > m \longrightarrow (\forall j < i)\ v_j = M.$

Proof. If

$$v_i = \min\left(\max\left(r_i n - \sum_{k<i} v_k - (n-i)m, m\right), M\right) > m$$

then, by Lemma 3.5.2, we have

$$r_i n \geq \sum_{k\leq i} v_k + (n-i)m = v_i + \sum_{k<i} v_k + (n-i)m$$

from which follows

$$r_i n - \sum_{k<i} v_k - (n-i)m \geq v_i$$

and, since $v_i > m$,

$$r_i n - \sum_{k<i} v_k - (n-i)m > m$$

or

$$r_i n > m + \sum_{k<i} v_k + (n-i)m = \sum_{k<i} v_k + (n-i+1)m$$

Then, since $\{k : k \leq j\} \subseteq \{k : k < i\}$ whenever $j < i$,

$$(\forall j < i)\ r_i n > \sum_{k \leq j} v_k + (n-i+1)m$$

Furthermore, $(\forall j < i)\ r_j \geq r_i$ and $(\forall j < i)\ r_j n \geq r_i n$, and so

$$(\forall j < i)\ r_j n > \sum_{k \leq j} v_k + (n-i+1)m$$

which means

$$(\forall j < i)\ (i-1)m > \sum_{k \leq j} v_k + nm - r_j n$$

Since $j < i$, $j \leq i-1$; since $m \leq 0$, $jm \geq (i-1)m$. So

$$(\forall j < i)\ jm > \sum_{k \leq j} v_k + nm - r_j n$$

and

$$(\forall j < i)\ r_j n > \sum_{k \leq j} v_k + (n-j)m$$

and, by applying the contrapositive of Lemma 3.5.3,

$$(\forall j < i)\ v_j \geq M$$

which means

$$(\forall j < i)\ v_j = M$$

since $(\forall i)\ m \leq v_i \leq M$. □

Next we define a property that captures a notion of "partial" equilibrium. The Boolean value StableUpTo(i) is true when, for $j \leq i$, all votes v_j are equal to their voters' optimal strategies if it is assumed that $(k > i)\ v_k = m$.

$$\text{StableUpTo}(i) \;\equiv\; (\forall j \leq i)\; v_j = \min\left(\max\left(r_j n - \sum_{k \leq i \wedge k \neq j} v_k - (n-i)m, m\right), M\right)$$

If we can show that StableUpTo(n) is necessarily true, then we will have succeeded in proving that $\vec{v}$ is an equilibrium. StableUpTo(0) is vacuously true; that StableUpTo(1) is true follows trivially from Algorithm 3.5.1's assignment to v_1. On the other hand, it is less obvious that StableUpTo(2) is true; specifically, it may be hard to see why

$$v_1 = \min\left(\max\left(r_1 n - \sum_{k \leq 2 \wedge k \neq 1} v_k - (n-2)m, m\right), M\right) = \min(\max(r_1 n - v_2 - (n-2)m, m), M)$$

(the $i = 2$, $j = 1$ case) must be true. To prove StableUpTo(i) for $2 \leq i \leq n$, we use a kind of induction.

Theorem 3.5.5. $(\forall i > 0)$ StableUpTo($i-1$) $\longrightarrow$ StableUpTo(i).

Proof. If StableUpTo($i-1$) is true, then

$$(\forall j < i)\; v_j = \min\left(\max\left(r_j n - \sum_{k < i \wedge k \neq j} v_k - (n-(i-1))m, m\right), M\right)$$

Algorithm 3.5.1 will assign a value between m and M, inclusive, to v_i. There are two cases: $v_i = m$ and $v_i > m$.

Case 1: $v_i = m$. Then $v_i - m = 0$, and

$$(\forall j < i) \sum_{k < i \wedge k \neq j} v_k = \sum_{k < i \wedge k \neq j} v_k + v_i - m = \sum_{k \leq i \wedge k \neq j} v_k - m$$

Therefore, substituting into StableUpTo($i-1$),

$$(\forall j < i)\; v_j = \min\left(\max\left(r_j n - \left(\sum_{k \leq i \wedge k \neq j} v_k - m\right) - (n-(i-1))m, m\right), M\right)$$

or

$$(\forall j < i)\ v_j = \min\left(\max\left(r_j n - \sum_{k \le i \wedge k \ne j} v_k - (n-i)m, m\right), M\right)$$

Case 2: $v_i > m$. Applying Lemma 3.5.2, we have

$$r_i n \ge \sum_{k \le i} v_k + (n-i)m$$

It follows that

$$(\forall j < i)\ r_i n \ge \sum_{k \le i \wedge k \ne j} v_k + v_j + (n-i)m$$

Since $v_i > m$, Lemma 3.5.4 tells us that $(\forall j < i)\ v_j = M$, and so

$$(\forall j < i)\ r_i n \ge \sum_{k \le i \wedge k \ne j} v_k + M + (n-i)m$$

Then, since $(\forall j < i)\ r_j n \ge r_i n$,

$$(\forall j < i)\ r_j n \ge \sum_{k \le i \wedge k \ne j} v_k + M + (n-i)m$$

or

$$(\forall j < i)\ r_j n - \sum_{k \le i \wedge k \ne j} v_k - (n-i)m \ge M$$

It follows that

$$(\forall j < i)\ \min\left(\max\left(r_j n - \sum_{k \le i \wedge k \ne j} v_k - (n-i)m, m\right), M\right) = M$$

Again we can use Lemma 3.5.4, $(\forall j < i)\ v_j = M$, and

$$(\forall j < i)\ \min\left(\max\left(r_j n - \sum_{k \le i \wedge k \ne j} v_k - (n-i)m, m\right), M\right) = v_j$$

Conclusion. So, whether $v_i = m$ or $v_i > m$,

$$(\forall j < i)\ v_j = \min\left(\max\left(r_j n - \sum_{k \leq i \wedge k \neq j} v_k - (n-i)m, m\right), M\right)$$

In addition, Algorithm 3.5.1 guarantees that v_i is assigned the value
$\min\left(\max\left(r_i n - \sum_{k<i} v_k - (n-i)m, m\right), M\right)$, which equals
$\min\left(\max\left(r_i n - \sum_{k \leq i \wedge k \neq i} v_k - (n-i)m, m\right), M\right)$, so

$$(\forall j \leq i)\ v_j = \min\left(\max\left(r_j n - \sum_{k \leq i \wedge k \neq j} v_k - (n-i)m, m\right), M\right)$$

which is precisely StableUpTo(i).

So, for any i, whether $v_i = m$ or $v_i > m$, StableUpTo($i-1$) $\longrightarrow$ StableUpTo(i). □

We are finally ready to prove that an equilibrium always exists.

Theorem 3.5.6. *For any $\vec{r}$, where $0 \leq r_i \leq 1$ for $1 \leq i \leq n$, the vote vector $\vec{v}$ returned by Algorithm 3.5.1 satisfies*

$$(\forall i)\ v_i = \min\left(\max\left(r_i n - \sum_{k \neq i} v_k, m\right), M\right)$$

Proof. If $\vec{v}$ is the vector returned by Algorithm 3.5.1, StableUpTo(0) is vacuously true, and so the truth of $(\forall i \leq n)$ StableUpTo(i) follows from Theorem 3.5.5. In particular, StableUpTo(n) must be true:

$$(\forall j \leq n)\ v_j = \min\left(\max\left(r_j n - \sum_{k \leq n \wedge k \neq j} v_k - (n-n)m, m\right), M\right)$$

or

$$(\forall j \leq n)\ v_j = \min\left(\max\left(r_j n - \sum_{k \neq j} v_k, m\right), M\right)$$

which is the optimal strategy for all voters. It directly follows that Algorithm 3.5.1, which is deterministic and always halts, will necessarily find an equilibrium from which no voter i would choose to change v_i using the rational Average strategy. □

So an equilibrium $\vec{v}$ must always exist for any input $\vec{r}$ and any $m \leq 0$ and $M \geq 1$.

We now know that, given some sincere-preference vector $\vec{r}$,

- at most one value ϕ satisfies Equations 3.8 and 3.9 (Theorem 3.4.1),
- any equilibrium $\vec{v}$ has average vote $\bar{v}$ satisfying Equations 3.6 and 3.7 (section 3.2.2), and
- at least one equilibrium $\vec{v}$ must exist (Theorem 3.5.6)

and so we can conclude that any ϕ that satisfies Equations 3.8 and 3.9 must equal the average vote $\bar{v}$ at all equilibria $\vec{v}$.

3.6 Average-Approval-Rating DSV

We have seen that Algorithm 3.5.1, *FindEquilibrium*, always finds an equilibrium for any sincere-preference vector $\vec{r}$. We also know that any equilibrium $\vec{v}$ will have the same average $\bar{v}$ (and that $0 \leq \bar{v} \leq 1$). It follows that the average at equilibrium is unique and can be defined as a function:

Algorithm 3.6.1.

$AverageAtEquilibrium(\vec{r}, m, M)$:

  $\vec{v} \leftarrow \text{FindEquilibrium}(\vec{r}, m, M)$

  return $\bar{v} = \frac{\sum_{i=1}^{n} v_i}{n}$

Even when $m < 0$ and/or $M > 1$, *AverageAtEquilibrium* will return an outcome between 0 and 1. In fact, the outcome returned will be within the range defined by the input vector of cardinal preferences:

Theorem 3.6.2. $(\forall m \leq 0, M \geq 1)\ \min(\vec{r}) \leq \text{AverageAtEquilibrium}(\vec{r}, m, M) \leq \max(\vec{r})$.

Proof. Say there is some $m \leq 0$ and some $M \geq 1$ such that $\min(\vec{r}) > \text{AverageAtEquilibrium}(\vec{r}, m, M)$. Then it follows that

$$(\forall i)\ r_i > \text{AverageAtEquilibrium}(\vec{r}, m, M)$$

and so

$$(\forall i)\ r_i > \bar{v} = \frac{\sum_{j=1}^{n} v_j}{n}$$

where $\vec{v} = \text{FindEquilibrium}(\vec{r}, m, M)$. According to Equation 3.4,

$$(\forall i)\ r_i > \bar{v} \longrightarrow v_i = M$$

and we can conclude that

$$(\forall i)\ v_i = M$$

which means that $\bar{v} = M \geq 1$. But $(\forall i)\ r_i > \bar{v}$, so

$$(\forall i)\ r_i > 1$$

which contradicts the fact that $(\forall i)\ 0 \leq r_i \leq 1$. Therefore there can be no $m \leq 0$ and $M \geq 1$ such that $\min(\vec{r}) > \text{AverageAtEquilibrium}(\vec{r}, m, M)$.

Now say there is some $m \leq 0$ and some $M \geq 1$ such that $\max(\vec{r}) < \text{AverageAtEquilibrium}(\vec{r}, m, M)$. Then it follows that

$$(\forall i)\ r_i < \text{AverageAtEquilibrium}(\vec{r}, m, M)$$

and so

$$(\forall i)\ r_i < \bar{v} = \frac{\sum_{j=1}^{n} v_j}{n}$$

where $\vec{v} = \text{FindEquilibrium}(\vec{r}, m, M)$. According to Equation 3.5,

$$(\forall i)\ r_i < \bar{v} \longrightarrow v_i = m$$

and we can conclude that

$$(\forall i)\ v_i = m$$

which means that $\bar{v} = m \leq 0$. But $(\forall i)\ r_i < \bar{v}$, so

$$(\forall i)\ r_i < 0$$

which contradicts the fact that $(\forall i)\ 0 \leq r_i \leq 1$. Therefore there can be no $m \leq 0$ and $M \geq 1$ such that $\max(\vec{r}) < \text{AverageAtEquilibrium}(\vec{r}, m, M)$.

So there is no $m \leq 0$ and $M \geq 1$ such that $\min(\vec{r}) > \text{AverageAtEquilibrium}(\vec{r}, m, M)$ or $\max(\vec{r}) < \text{AverageAtEquilibrium}(\vec{r}, m, M)$. Therefore it must be that

$$(\forall m \leq 0, M \geq 1)\ \min(\vec{r}) \leq \text{AverageAtEquilibrium}(\vec{r}, m, M) \leq \max(\vec{r})$$

$\square$

These bounds are tight; in fact,

$$(\forall M \geq 1)\ \lim_{m \to -\infty} \text{AverageAtEquilibrium}(\vec{v}, m, M) = \min(\vec{v})$$

and

$$(\forall m \leq 0)\ \lim_{M \to +\infty} \text{AverageAtEquilibrium}(\vec{v}, m, M) = \max(\vec{v})$$

3.6.1 A new class of rating systems

The Average and Median protocols necessarily take a vote vector $\vec{v}$ as input—voters' sincere preference information cannot be directly and reliably elicited, so $\vec{r}$ is not generally available. If the Average system is used and voters are rationally strategic (and are allowed to keep changing their votes until all decide to stand pat), the outcome can reasonably be expected to equal $\text{AverageAtEquilibrium}(\vec{r}, 0, 1)$. But instead of using Average on the vote vector $\vec{v}$ and relying on the voters to use optimally rational strategy when deciding on their votes v_i, $\text{AverageAtEquilibrium}(\vec{v}, 0, 1)$ can be calculated and taken as the outcome, implicitly and effectively using the DSV [23] framework with Average as the underlying voting protocol. In fact, we are not limited to AverageAtEquilibrium$(\vec{v}, 0, 1)$; we have just seen that AverageAtEquilibrium$(\vec{v}, m, M)$ lies between 0 and 1 for any $m \leq 0$ and $M \geq 1$ and so can serve as a rating system as well.

For illustration, we reuse the *Videodrome* example and assume sincere voters:

$$\vec{v} = [0.4, 0.7, 0.8, 0.8, 0.88]$$

Suppose we want to take as the outcome of this election not the average vote $\bar{v}$ or the median vote $\tilde{v}$ but AverageAtEquilibrium$(\vec{v}, 0, 1)$. First we calculate FindEquilibrium$(\vec{v}, 0, 1)$, which we have seen in section 3.2.2 to be

$$\vec{w} = \text{FindEquilibrium}(\vec{v}, 0, 1) = [0, 0.5, 1, 1, 1]$$

Then we see that

$$\bar{w} = \frac{\sum_{i=1}^{5} w_i}{5} = \frac{0 + 0.5 + 1 + 1 + 1}{5} = 0.7$$

giving the outcome as 0.7, which equals neither the Average outcome ($\bar{v} = 0.716$) nor the Median outcome ($\tilde{v} = 0.8$).

Alternatively, we can let $m = -99$ and $M = 100$. Then the equilibrium we find turns out to be

$$\vec{w} = \text{FindEquilibrium}(\vec{v}, -99, 100) = [-99, -99, 2, 100, 100]$$

And then

$$\bar{w} = \frac{\sum_{i=1}^{5} w_i}{5} = \frac{-99 + (-99) + 2 + 100 + 100}{5} = 0.8$$

This time the power to determine the outcome fell to voter 3 rather than voter 2, giving the Median outcome of 0.8. (We will see that if $m + M = 1$ and $M - m$ is allowed to become large enough, the resultant outcome will equal the Median outcome.)

It turns out that in neither of these cases will any voter be able to gain from voting insincerely. For example, if voter 3 in the $m = -99$, $M = 100$ case voted anything but 2, the outcome would deviate from the ideal $r_3 = 0.8$; if voter 2 voted anything other than -99, the outcome would be larger than 0.8, moving farther away from $r_2 = 0.7$. This is no coincidence.

We will now prove that this Average-Approval-Rating (AAR) DSV system has three intuitively desirable properties: a kind of monotonicity (Theorem 3.6.3), immunity to Average-like strategy (Theorem 3.6.4) and a general nonmanipulability (Theorem 3.6.5). The first two will imply the third.

3.6.2 Monotonicity of AAR DSV

First, the monotonicity property: When some input votes are increased and none is decreased, the outcome never decreases.

Theorem 3.6.3. *If* $\vec{v} = [v_1, v_2, \ldots v_n]$ *and* $\vec{v}' = [v'_1, v'_2, \ldots v'_n]$ *where* $(\forall i)\ v_i \leq v'_i$*, then* $\text{AverageAtEquilibrium}(\vec{v}, m, M) \leq \text{AverageAtEquilibrium}(\vec{v}', m, M)$.

Proof. Equations 3.6 and 3.7 say that

$$|\{i : \bar{w} < v_i\}| \leq \frac{\bar{w} - m}{M - m} n \leq |\{i : \bar{w} \leq v_i\}|$$

where $\vec{w} = \text{FindEquilibrium}(\vec{v}, m, M)$.

Therefore, if $\hat{v} = \text{AverageAtEquilibrium}(\vec{v}, m, M)$ and $\hat{v}' = \text{AverageAtEquilibrium}(\vec{v}', m, M)$, it must be that

$$|\{i : \hat{v} < v_i\}| \leq \frac{\hat{v} - m}{M - m} n \leq |\{i : \hat{v} \leq v_i\}|$$

and

$$|\{i : \hat{v}' < v'_i\}| \leq \frac{\hat{v}' - m}{M - m} n \leq |\{i : \hat{v}' \leq v'_i\}|$$

Also, since $(\forall i)\ v_i \leq v'_i$, we have that

$$(\forall i)\ \hat{v} \leq v_i \longrightarrow \hat{v} \leq v'_i$$

Assume for now that $\hat{v} > \hat{v}'$. Then

$$(\forall i)\ \hat{v} \leq v_i \longrightarrow \hat{v}' < v_i'$$

and so

$$\{i : \hat{v} \leq v_i\} \subseteq \{i : \hat{v}' < v_i'\}$$

which means that

$$|\{i : \hat{v} \leq v_i\}| \leq |\{i : \hat{v}' < v_i'\}|$$

But, since $\frac{\hat{v}-m}{M-m}n \leq |\{i : \hat{v} \leq v_i\}|$ and $|\{i : \hat{v}' < v_i'\}| \leq \frac{\hat{v}'-m}{M-m}n$, it must be true that

$$\frac{\hat{v}-m}{M-m}n \leq \frac{\hat{v}'-m}{M-m}n$$

n is positive, so

$$\frac{\hat{v}-m}{M-m} \leq \frac{\hat{v}'-m}{M-m}$$

and $M > m$, so

$$\hat{v} - m \leq \hat{v}' - m$$

and thus

$$\hat{v} \leq \hat{v}'$$

contradicting the assumption that $\hat{v} > \hat{v}'$. Therefore $\hat{v} > \hat{v}'$ must be false, so $\hat{v} \leq \hat{v}'$. $\square$

3.6.3 AAR DSV is immune to Average-style strategy

Another desirable property of AAR DSV is that its outcome is unaffected by voters' using Average-style strategy, trying to move the outcome in the desired direction by moving their votes in that direction.

Theorem 3.6.4. *If $\vec{v} = [v_1, v_2, \ldots v_n]$ and $\vec{v}' = [v_1', v_2', \ldots v_n']$ where, for all $1 \leq i \leq n$,*

- $v_i' \leq v_i$ *if* AverageAtEquilibrium$(\vec{v}, m, M) > v_i$

- $v_i' = v_i$ *if* $\text{AverageAtEquilibrium}(\vec{v}, m, M) = v_i$
- $v_i' \geq v_i$ *if* $\text{AverageAtEquilibrium}(\vec{v}, m, M) < v_i$

then $\text{AverageAtEquilibrium}(\vec{v}', m, M) = \text{AverageAtEquilibrium}(\vec{v}, m, M)$.

Proof. From the definition of $\vec{v}'$, we have that

$$(\forall i)\ \hat{v} > v_i \longrightarrow v_i' \leq v_i$$

$$(\forall i)\ \hat{v} = v_i \longrightarrow v_i' = v_i$$

$$(\forall i)\ \hat{v} < v_i \longrightarrow v_i' \geq v_i$$

where $\hat{v} = \text{AverageAtEquilibrium}(\vec{v}, m, M)$. These respectively imply that

$$(\forall i)\ \hat{v} > v_i \longrightarrow \hat{v} > v_i' \tag{3.10}$$

$$(\forall i)\ \hat{v} = v_i \longrightarrow \hat{v} = v_i' \tag{3.11}$$

$$(\forall i)\ \hat{v} < v_i \longrightarrow \hat{v} < v_i' \tag{3.12}$$

From 3.10 and 3.11 we can see that

$$(\forall i)\ \hat{v} \geq v_i \longrightarrow \hat{v} \geq v_i'$$

Combining the contrapositive of this result with 3.12 gives

$$(\forall i)\ \hat{v} < v_i \longleftrightarrow \hat{v} < v_i'$$

which means that

$$\{i : \hat{v} < v_i\} = \{i : \hat{v} < v_i'\}$$

Since $\hat{v} = \text{AverageAtEquilibrium}(\vec{v}, m, M)$, Equation 3.6 tells us that

$$|\{i : \hat{v} < v_i\}| \leq \frac{\hat{v} - m}{M - m} n$$

and so we can conclude that

$$|\{i : \hat{v} < v_i'\}| \leq \frac{\hat{v} - m}{M - m} n$$

Similarly, from 3.11 and 3.12 we can see that

$$(\forall i)\ \hat{v} \leq v_i \longrightarrow \hat{v} \leq v_i'$$

Combining this result with the contrapositive of 3.10 gives

$$(\forall i)\ \hat{v} \leq v_i \longleftrightarrow \hat{v} \leq v_i'$$

which means that

$$\{i : \hat{v} \leq v_i\} = \{i : \hat{v} \leq v_i'\}$$

Since $\hat{v} = \text{AverageAtEquilibrium}(\vec{v}, m, M)$, Equation 3.7 tells us that

$$\frac{\hat{v} - m}{M - m} n \leq |\{i : \hat{v} \leq v_i\}|$$

and so we can conclude that

$$\frac{\hat{v} - m}{M - m} n \leq |\{i : \hat{v} \leq v_i'\}|$$

Now we have that

$$|\{i : \hat{v} < v_i'\}| \leq \frac{\hat{v} - m}{M - m} n \leq |\{i : \hat{v} \leq v_i'\}|$$

and, if $\hat{v}' = \text{AverageAtEquilibrium}(\vec{v}', m, M)$, then $\hat{v}'$ satisfies

$$|\{i : \hat{v}' < v_i'\}| \leq \frac{\hat{v}' - m}{M - m} n \leq |\{i : \hat{v}' \leq v_i'\}|$$

Finally, by Theorem 3.4.1, $\hat{v} = \hat{v}'$, and so

AverageAtEquilibrium$(\vec{v}, m, M)$ = AverageAtEquilibrium$(\vec{v}', m, M)$. □

3.6.4 AAR DSV never rewards insincerity

For any voting system, it is desirable to show that a voter can never gain a better outcome by voting insincerely than by voting sincerely, however sincerity is defined. It turns out that, when AverageAtEquilibrium$(\vec{v}, m, M)$ is selected as the outcome, no voter i can gain an outcome closer to the ideal r_i by voting $v_i \neq r_i$ instead of $v_i = r_i$, guaranteeing a strong nonmanipulability property to AAR DSV:

Theorem 3.6.5. *If* $\vec{v} = [v_1, v_2, \ldots v_n]$ *where* $v_1 = r_1$ *and* $\vec{v}' = [v'_1, v'_2, \ldots v'_n]$ *where* $v'_1 \neq r_1$ *and* $(\forall i > 1)\ v'_i = v_i$*, then*
$|\text{AverageAtEquilibrium}(\vec{v}, m, M) - r_1| \leq |\text{AverageAtEquilibrium}(\vec{v}', m, M) - r_1|$.

Proof. Abbreviating AverageAtEquilibrium$(\vec{v}, m, M)$ as $\hat{v}$ and AverageAtEquilibrium$(\vec{v}', m, M)$ as $\hat{v}'$, if $\hat{v} = r_1$, then $|\hat{v} - r_1| = 0$, and so then we can conclude immediately that

$$|\hat{v} - r_1| \leq |\hat{v}' - r_1|$$

On the other hand, it may be that $\hat{v} \neq r_1$. Since it is also true that $v'_1 \neq r_1$, there are now four cases to consider:

1. $\hat{v} > r_1$ and $v'_1 > r_1$
2. $\hat{v} > r_1$ and $v'_1 < r_1$
3. $\hat{v} < r_1$ and $v'_1 < r_1$
4. $\hat{v} < r_1$ and $v'_1 > r_1$

Cases 2 and 4 might be said to represent reasonable attempts at effective strategy, while cases 1 and 3 entail insincere voting in the "wrong" direction. Considering them in order:

Case 1: $\hat{v} > r_1$ and $v_1' > r_1$. Then, since $v_1 = r_1$, $v_1' > v_1$; since $(\forall i > 1)\ v_i' = v_i$,

$$(\forall i)\ v_i \leq v_i'$$

and so Theorem 3.6.3 allows us to conclude that

$$\hat{v} = \text{AverageAtEquilibrium}(\vec{v}, m, M) \leq \text{AverageAtEquilibrium}(\vec{v}', m, M) = \hat{v}'$$

which means

$$\hat{v} - r_1 \leq \hat{v}' - r_1$$

We know that $\hat{v} > r_1$, and $\hat{v}' > r_1$ since $\hat{v} \leq \hat{v}'$, so $\hat{v} - r_1$ and $\hat{v}' - r_1$ are both positive. Thus

$$|\hat{v} - r_1| \leq |\hat{v}' - r_1|$$

Case 2: $\hat{v} > r_1$ and $v_1' < r_1$. Then, since $v_1 = r_1$, $\hat{v} > v_1$ and $v_1' < v_1$. We also know that $(\forall i > 1)\ v_i' = v_i$, so we can use Theorem 3.6.4 to get

$$\hat{v} = \hat{v}'$$

and it immediately follows that

$$|\hat{v} - r_1| \leq |\hat{v}' - r_1|$$

Case 3: $\hat{v} < r_1$ and $v_1' < r_1$. Then, since $v_1 = r_1$, $v_1' < v_1$; since $(\forall i > 1)\ v_i' = v_i$,

$$(\forall i)\ v_i \geq v_i'$$

and so Theorem 3.6.3 allows us to conclude that

$$\hat{v} = \text{AverageAtEquilibrium}(\vec{v}, m, M) \geq \text{AverageAtEquilibrium}(\vec{v}', m, M) = \hat{v}'$$

which means

$$\hat{v} - r_1 \geq \hat{v}' - r_1$$

We know that $\hat{v} < r_1$, and $\hat{v}' < r_1$ since $\hat{v} \geq \hat{v}'$, so $\hat{v} - r_1$ and $\hat{v}' - r_1$ are both negative. Thus

$$|\hat{v} - r_1| \leq |\hat{v}' - r_1|$$

Case 4: $\hat{v} < r_1$ and $v_1' > r_1$. Then, since $v_1 = r_1$, $\hat{v} < v_1$ and $v_1' > v_1$. We also know that $(\forall i > 1)\ v_i' = v_i$, so we can use Theorem 3.6.4 to get

$$\hat{v} = \hat{v}'$$

and it immediately follows that

$$|\hat{v} - r_1| \leq |\hat{v}' - r_1|$$

Conclusion. In each case we have found $|\hat{v} - r_1| \leq |\hat{v}' - r_1|$ to hold. The cases are exhaustive, so we can conclude that

$$|\text{AverageAtEquilibrium}(\vec{v}, m, M) - r_1| \leq |\text{AverageAtEquilibrium}(\vec{v}', m, M) - r_1|$$

□

So insincere voters under these AAR DSV systems cannot move the outcome closer to ideal. However, it should be mentioned that a nonmanipulable rating system cannot be easily generalized to give nonmanipulable single-winner voting protocols. For example, while it is always rational to vote sincerely when using Median aggregation, the “majority judgement” protocol of Balinski and Laraki [6] (under which each voter submit cardinal votes over a finite number of discrete alternatives and the one with the highest median vote wins) can be manipulated by voting insincerely. For example, say there are three voters and three alternatives, and the votes are

	A	B	C
voter 1	0.8	0.7	0.1
voter 2	0.6	0.2	0.4
voter 3	0	0.9	0.3

A's median vote is 0.6, B's is 0.7 and C's is 0.3, so B wins. But if voter 1 gave B an insincere vote of 0.5 instead, thus lowering B's median vote to 0.5, A would win and voter 1 would benefit.

3.7 A simpler AAR DSV algorithm

It is possible to find a quicker, more direct way of calculating AverageAtEquilibrium($\vec{v}, m, M$) given a vector $\vec{v}$. One promising approach is to use the property

$$|\{i : \phi < v_i\}| \leq \frac{\phi - m}{M - m} n \leq |\{i : \phi \leq v_i\}|$$

We know that, given $\vec{v}$, this property is true for exactly one value ϕ, which is equal to AverageAtEquilibrium($\vec{v}, m, M$). The problem is that ϕ may be any number between 0 and 1; testing every $0 \leq \phi \leq 1$ individually is impossible.

Happily, it turns out that testing a finite number of possibilities for ϕ is sufficient to guarantee finding one that satisfies the above property. In particular, AverageAtEquilibrium($\vec{v}, m, M$) is always equal to either v_i for some $1 \leq i \leq n$ or $m + \frac{M-m}{n} i$ for some $i \in \mathbb{Z}$, $0 \leq i \leq n$.

Theorem 3.7.1.
$(\exists i \in \mathbb{Z})$ $\text{AverageAtEquilibrium}(\vec{v}, m, M) = v_i \vee \text{AverageAtEquilibrium}(\vec{v}, m, M) = m + \frac{M-m}{n} i$.

Proof. Recall that, given some vote vector $\vec{v}$,

$$\text{AverageAtEquilibrium}(\vec{v}, m, M) = \frac{\sum_i \text{FindEquilibrium}(\vec{v}, m, M)_i}{n}$$

Define $\vec{w} \equiv \text{FindEquilibrium}(\vec{v}, m, M)$, the vector that results when $\vec{v}$ is treated as the voters' sincere preferences and optimal Average strategy is applied on behalf of all voters until an equilibrium is reached. We now proceed in two cases: Either there is some w_i such that $m < w_i < M$ or there is none.

Case 1. $(\exists i)\ m < w_i < M$. Take i to be any value such that $m < w_i < M$. Since $\vec{w} = \text{FindEquilibrium}(\vec{v}, m, M)$, we know by Theorem 3.5.6 that

$$(\forall j)\ w_j = \min\left(\max\left(v_j n - \sum_{k \neq j} w_k, m\right), M\right)$$

and, in particular for i,

$$w_i = \min\left(\max\left(v_i n - \sum_{k \neq i} w_k, m\right), M\right)$$

Since $m < w_i < M$,

$$w_i = v_i n - \sum_{k \neq i} w_k$$

and therefore

$$v_i n = w_i + \sum_{k \neq i} w_k = \sum_k w_k$$

and

$$v_i = \frac{\sum_k w_k}{n} = \text{AverageAtEquilibrium}(\vec{v}, m, M)$$

It follows that $(\exists i)\ v_i = \text{AverageAtEquilibrium}(\vec{v}, m, M)$. This case illustrates that any voter at equilibrium not voting at one of the extremes must have realized his or her ideal outcome.

Case 2. $(\forall i)\ w_i = m \vee w_i = M$. Then $\sum_k w_k$ must equal $jM + (n-j)m = nm + j(M-m)$ for integer $j = |\{i : w_i = M\}|$. So we have

$$\text{AverageAtEquilibrium}(\vec{v}, m, M) = \frac{\sum_k w_k}{n} = \frac{nm + j(M-m)}{n} = m + \frac{M-m}{n} j$$

and it follows that $(\exists i \in \mathbb{Z})\ \text{AverageAtEquilibrium}(\vec{v}, m, M) = m + \frac{M-m}{n} i$.

Conclusion. These two cases are exhaustive. Therefore,

$$(\exists i \in \mathbb{Z})\ \text{AverageAtEquilibrium}(\vec{v}, m, M) = v_i \vee \text{AverageAtEquilibrium}(\vec{v}, m, M) = m + \frac{M-m}{n} i$$

□

This fact would seem to motivate a direct and efficient algorithm for computing AverageAtEquilibrium$(\vec{v}, m, M)$. First, take $O(n \log n)$ time to sort $\vec{v}$ so that $v_1 \geq v_2 \geq \ldots \geq v_n$. Then binary searches can be used to test each v_i value and each $\frac{M-m}{n} i$ value. For each potential solution ϕ, if

$$\frac{\phi - m}{M - m} n > |\{i : \phi \leq v_i\}|$$

then we can rule out ϕ and every value greater than ϕ; if

$$|\{i : \phi < v_i\}| > \frac{\phi - m}{M - m} n$$

then we can rule out ϕ and every value less than ϕ. Each of these tests runs in $O(n)$ time, so the binary searches will finish and the correct outcome will be found (as Theorem 3.7.1 guarantees) in $O(n \log n)$ time.

Another efficient approach is to streamline Algorithm 3.6.1 directly:

AverageAtEquilibrium$(\vec{v}, m, M)$:

sort $\vec{v}$ so that $(\forall i \leq j)\ v_i \geq v_j$

for $i = 1$ to n do

$w_i \leftarrow \min\left(\max\left(v_i n - \sum_{k<i} w_k - (n-i)m, m\right), M\right)$

return $\bar{w} = \frac{\sum_{i=1}^{n} w_i}{n}$

The w_i assignment effectively assumes that $w_j = m$ for $j > i$ and applies the optimal strategy function for the ith voter:

Algorithm 3.7.2.

AverageAtEquilibrium$(\vec{v}, m, M)$:

sort $\vec{v}$ so that $(\forall i \leq j)\ v_i \geq v_j$

for $i = 1$ to n do

$w_i \leftarrow m$

for $i = 1$ to n do

$w_i \leftarrow \min\left(\max\left(v_i n - \sum_{k \neq i} w_k, m\right), M\right)$

return $\bar{w} = \frac{\sum_{i=1}^{n} w_i}{n}$

Instead of calculating $\sum_{k \neq i} w_k$ freshly each time, we can easily keep track of $w_{sum} = \sum_k w_k$ and subtract $w_i = m$ from it. For the first loop iteration w_{sum} is equal to $\sum_k m = nm$, and from the ith to the $(i+1)$th iteration it increases by $w_i - m$, as the new value of w_i replaces the old value, m:

Algorithm 3.7.3.

AverageAtEquilibrium$(\vec{v}, m, M)$:

sort $\vec{v}$ so that $(\forall i \leq j)\ v_i \geq v_j$

$w_{sum} \leftarrow nm$

for $i = 1$ to n do

$w_i \leftarrow \min\left(\max\left(v_i n - w_{sum} + m, m\right), M\right)$

$w_{sum} \leftarrow w_{sum} + w_i - m$

return $\bar{w} = \frac{\sum_{i=1}^{n} w_i}{n}$

Now we can get rid of $\vec{w}$ entirely:

Algorithm 3.7.4.

AverageAtEquilibrium$(\vec{v}, m, M)$:

sort $\vec{v}$ so that $(\forall i \leq j)\ v_i \geq v_j$

$w_{sum} \leftarrow nm$

for $i = 1$ to n do

$w_{sum} \leftarrow w_{sum} + \min\left(\max\left(v_i n - w_{sum}, 0\right), M - m\right)$

return $\frac{w_{sum}}{n}$

The sorting step can be done in $O(n \log n)$ time; each w_{sum} update takes constant time, so everything after the sorting step finishes in $O(n)$ time.

3.8 Parameterizing AAR DSV

Given a vote vector $\vec{v}$, we have seen that the outcome $\hat{v} = \text{AverageAtEquilibrium}(\vec{v}, 0, 1)$ of "pure" AAR DSV, which is DSV applied to the original $[0, 1]$-Average system, satisfies the property

$$|\{i : \hat{v} < v_i\}| \leq \hat{v}n \leq |\{i : \hat{v} \leq v_i\}|$$

We also know that the order-statistic outcome ${}^b\tilde{v}$ (the generalization of Median from section 3.1.3), for constant $0 \leq b \leq 1$, satisfies

$$|\{i : {}^b\tilde{v} < v_i\}| \leq bn \leq |\{i : {}^b\tilde{v} \leq v_i\}|$$

This looks like a job for interpolation! We define a new parameter a that varies between 0 (giving ${}^b\tilde{v}$) and 1 (giving $\text{AverageAtEquilibrium}(\vec{v}, 0, 1)$). The generalized outcome $\Phi_{a,b}(\vec{v})$ then will satisfy

$$|\{i : \Phi_{a,b}(\vec{v}) < v_i\}| \leq (a\Phi_{a,b}(\vec{v}) + (1-a)b)n \leq |\{i : \Phi_{a,b}(\vec{v}) \leq v_i\}| \tag{3.13}$$

and we have already found exactly such a function. We know that

$$|\{i : \hat{v} < v_i\}| \leq \frac{\hat{v} - m}{M - m} n \leq |\{i : \hat{v} \leq v_i\}|$$

where $\hat{v} = \text{AverageAtEquilibrium}(\vec{v}, m, M)$. So if

$$\frac{\Phi_{a,b}(\vec{v}) - m}{M - m} = a\Phi_{a,b}(\vec{v}) + (1-a)b$$

where $a = \frac{1}{M-m}$, $b = \frac{m}{1-M+m}$, $m = b - \frac{b}{a}$ and $M = b + \frac{1-b}{a}$, then we can define

$$\Phi_{a,b}(\vec{v}) \equiv \lim_{x \to a^+} \text{AverageAtEquilibrium}\left(\vec{v}, b - \frac{b}{x}, b + \frac{1-b}{x}\right)$$

and $\Phi_{a,b}(\vec{v})$ will satisfy Equation 3.13. (The limit is needed for the $a = 0$ case; as a approaches 0, $\Phi_{a,b}(\vec{v})$ approaches the ${}^b\tilde{v}$ outcome defined in section 3.1.3.) In fact, a value ϕ satisfies

$$|\{i : \phi < v_i\}| \leq (a\phi + (1-a)b)n \leq |\{i : \phi \leq v_i\}|$$

if and only if $\phi = \Phi_{a,b}(\vec{v})$, as we showed in section 3.5.

Theorem 3.8.1. $(\forall a, b)\ \min(\vec{v}) \leq \Phi_{a,b}(\vec{v}) \leq \max(\vec{v})$.

Proof. Whenever $0 \leq a, b \leq 1$, it is true that

$$ab \leq b$$

and so

$$\lim_{x \to a^+} b - \frac{b}{x} \leq 0 \tag{3.14}$$

It also must be that $a - 1 \leq 0$ and $b - 1 \leq 0$, so

$$(a-1)(b-1) = ab - a - b + 1 \geq 0$$

and

$$ab + 1 - b \geq a$$

which means

$$\lim_{x \to a^+} b + \frac{1-b}{x} \geq 1 \tag{3.15}$$

Theorem 3.6.2 says that

$$(\forall m \leq 0, M \geq 1)\ \min(\vec{v}) \leq \text{AverageAtEquilibrium}(\vec{v}, m, M) \leq \max(\vec{v})$$

Given this and Equations 3.14 and 3.15, it follows that

$$(\forall a, b)\ \min(\vec{v}) \leq \lim_{x \to a^+} \text{AverageAtEquilibrium}\left(\vec{v}, b - \frac{b}{x}, b + \frac{1-b}{x}\right) \leq \max(\vec{v})$$

By definition of the $\Phi_{a,b}$ function, we can conclude that

$$(\forall a, b)\ \min(\vec{v}) \leq \Phi_{a,b}(\vec{v}) \leq \max(\vec{v})$$

□

Therefore any Φ function's output can be neither higher than all inputs nor lower than all inputs. In particular, if $(\forall i)\ v_i = \omega$, then $(\forall a, b)\ \Phi_{a,b}(\vec{v}) = \omega$, so any Φ function satisfies an intuitively appealing unanimity property: when all voters agree, the outcome agrees with them.

3.9 Evaluation of AAR DSV systems

Any system that uses the outcome function $\Phi_{a,b}(\vec{v})$ where $0 \le a \le 1$ and $0 \le b \le 1$ has the property that no voter can gain by voting insincerely. But it does not follow that any values of a and b give equally desirable outcomes.

One approach to evaluating this continuous range of nonmanipulable systems is to take the Average system as a benchmark and determine which $\Phi_{a,b}$ function comes nearest, on average, to giving the Average outcome. Given a vote vector $\vec{v}$, we can calculate the Average outcome $\bar{v}$ and the outcome $\Phi_{a,b}(\vec{v})$ for many a, b combinations. For any particular a and b, we can calculate the squared error from $\bar{v}$:

$$\mathrm{SE}_{a,b}(\vec{v}) = (\Phi_{a,b}(\vec{v}) - \bar{v})^2$$

If $\mathbf{V} = \{\vec{v}_1, \vec{v}_2, \vec{v}_3 \ldots \vec{v}_N\}$ is a vector of N vote vectors, then we can find the root-mean-squared error from Average, weighted by the number of ratings in each vote vector $\vec{v}_i$:

$$\mathrm{RMSE}_{a,b}(\mathbf{V}) = \sqrt{\frac{\sum_{i=1}^{N} |\vec{v}_i| \cdot \mathrm{SE}_{a,b}(\vec{v}_i)}{\sum_{i=1}^{N} |\vec{v}_i|}}$$

Given some "training" vector $\mathbf{V}$ of vote vectors, we would like to choose a and b to minimize $\mathrm{RMSE}_{a,b}(\mathbf{V})$.

This approach requires a concrete source of vote-vector data or a distribution for generating such. The website Metacritic [1] offers ideal data for our purposes: Reviews for over 4000 films (plus many books, music albums, video games, etc.) are summarized into ratings between 0 and 100. For example, one film[1] has the seven ratings 70, 70, 80, 80, 88, 88 and 100, which are easily

[1] The 1978 film *Animal House*.

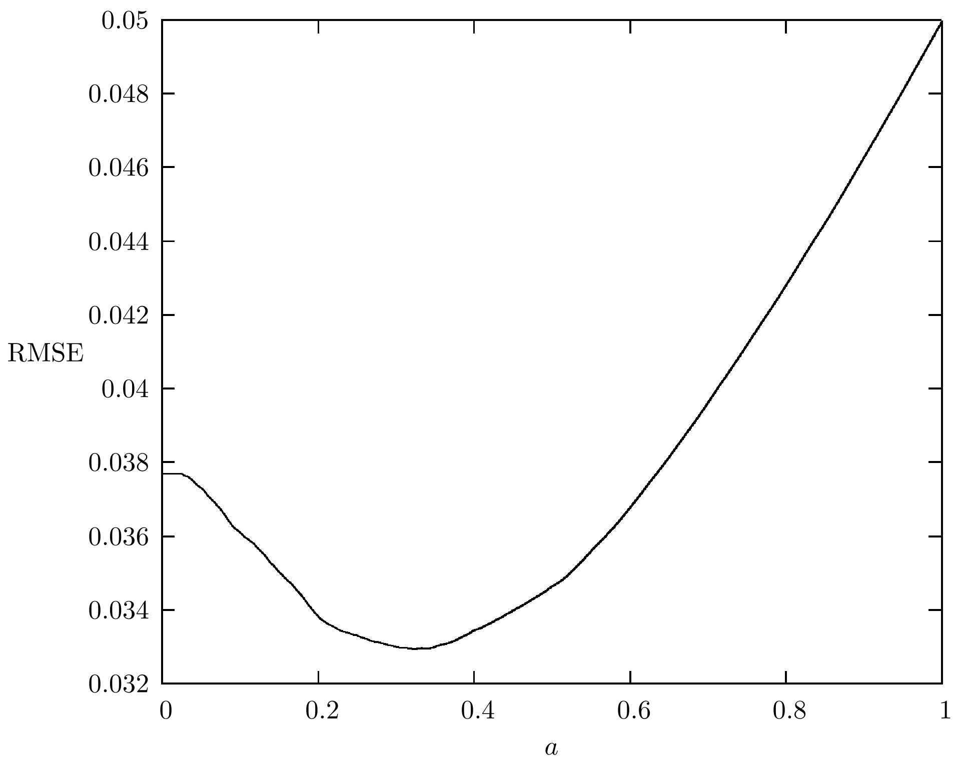
converted into the vote vector

$$\vec{v} = [0.7, 0.7, 0.8, 0.8, 0.88, 0.88, 1]$$

Converting all films on Metacritic the same way gives us a large vector $\mathbf{V}$ of vote vectors.[2]

Since there are two parameters, a and b, it is somewhat impractical to try all combinations. But it may be desired to fix $b = 0.5$ to ensure a kind of symmetry: If $(\forall i)\ v'_i = 1 - v_i$, then $(\forall a)\ \Phi_{a,0.5}(\vec{v}') = 1 - \Phi_{a,0.5}(\vec{v})$, so electorates that prefer low and high outcomes are treated symmetrically. Fixing $b = 0.5$ and trying all 10001 evenly spaced values of a, we find that $a = 0.3240$ (Figure 3.1) gives the minimum RMSE for the Metacritic data.

Figure 3.1: RMSE, varying a and fixing $b = 0.5000$



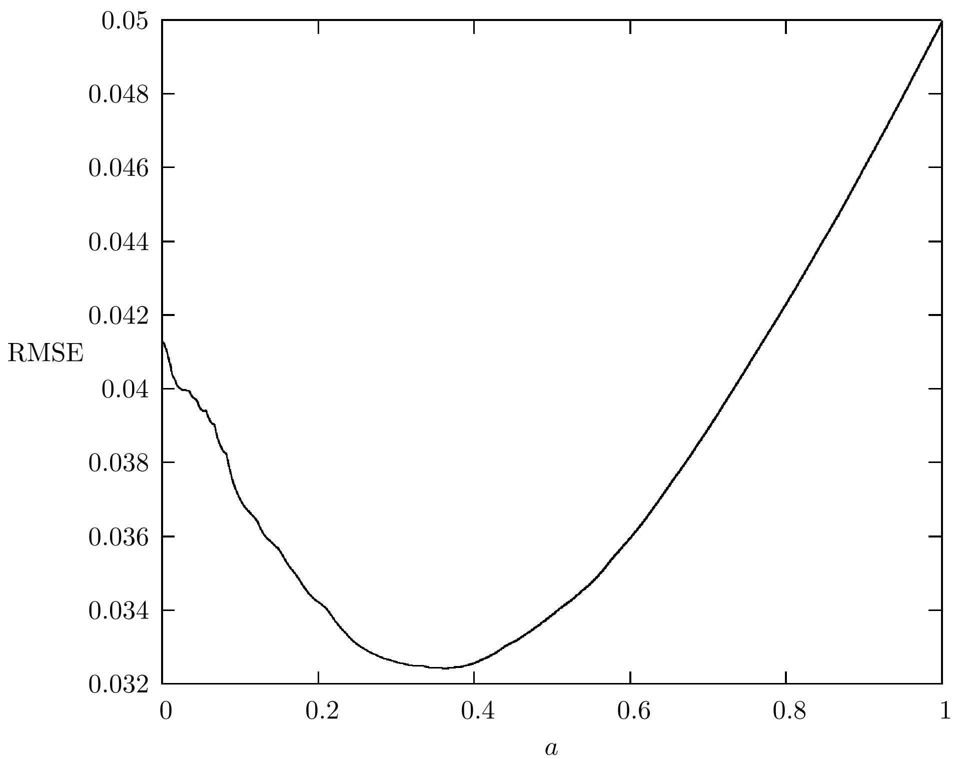
Having fixed $b = 0.5$ and found the value of a that minimizes RMSE (0.3240), we can now fix $a = 0.3240$ and find the value of b that minimizes RMSE, then fix b again accordingly and continue

[2]We use the data for the 4581 films mined from Metacritic on Thursday, 3 April 2008, that had at least three critics rate them.

in a hill-climbing fashion until we find a stable minimum. In practice, the procedure is guaranteed to halt because the RMSE decreases at each step for which either a or b changes.

Using this procedure on the Metacritic data and testing 10001 evenly spaced values of a or b at each step, whether we start with $a = 0$ or with $b = 0.5$, we find a local RMSE minimum (approximately 0.03242) at $a = 0.3647$, $b = 0.4820$ (Figures 3.2 and 3.3); such a system is equivalent to running an Average election with rationally optimal voters and allowing votes between $m \approx -0.8396$ and $M \approx 1.9023$.

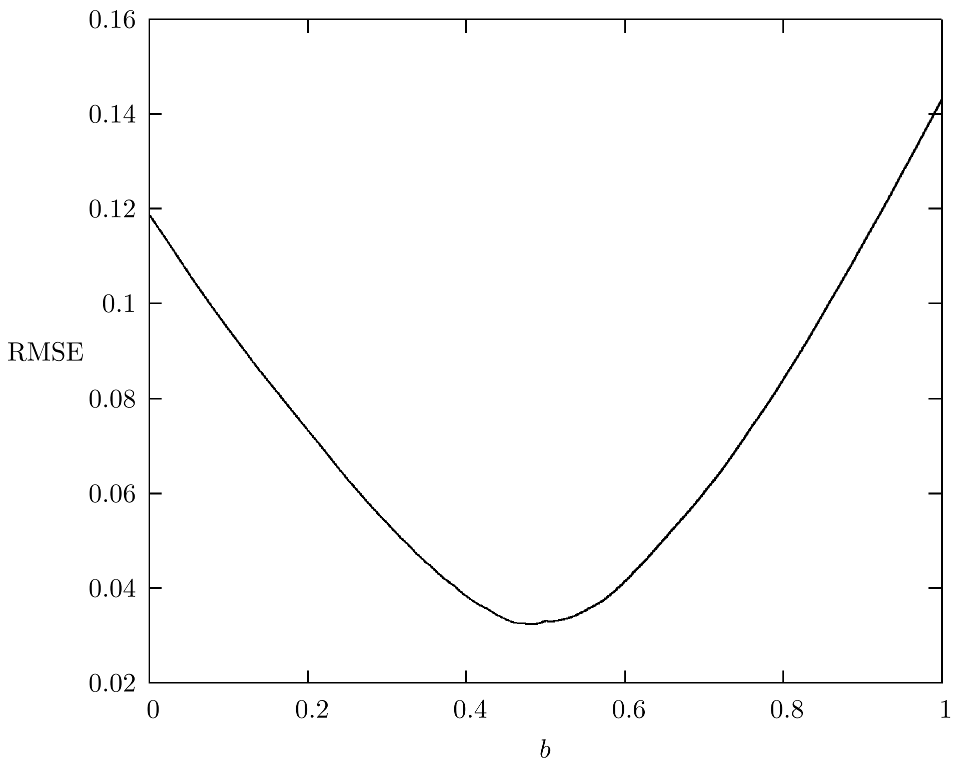
Figure 3.2: RMSE, varying a and fixing $b = 0.4820$



It can be seen in Figure 3.4 that $\Phi_{0.3647,0.4820}(\vec{v})$ and $\bar{v}$ do indeed correlate nicely for the Metacritic data.

To a good approximation, then, the AAR DSV system that comes closest to matching Average outcomes for the Metacritic data is one with $a = \frac{1}{3}$ and $b = \frac{1}{2}$, which corresponds to allowing votes between $m = -1$ and $M = 2$.

Figure 3.3: RMSE, fixing $a = 0.3647$ and varying b



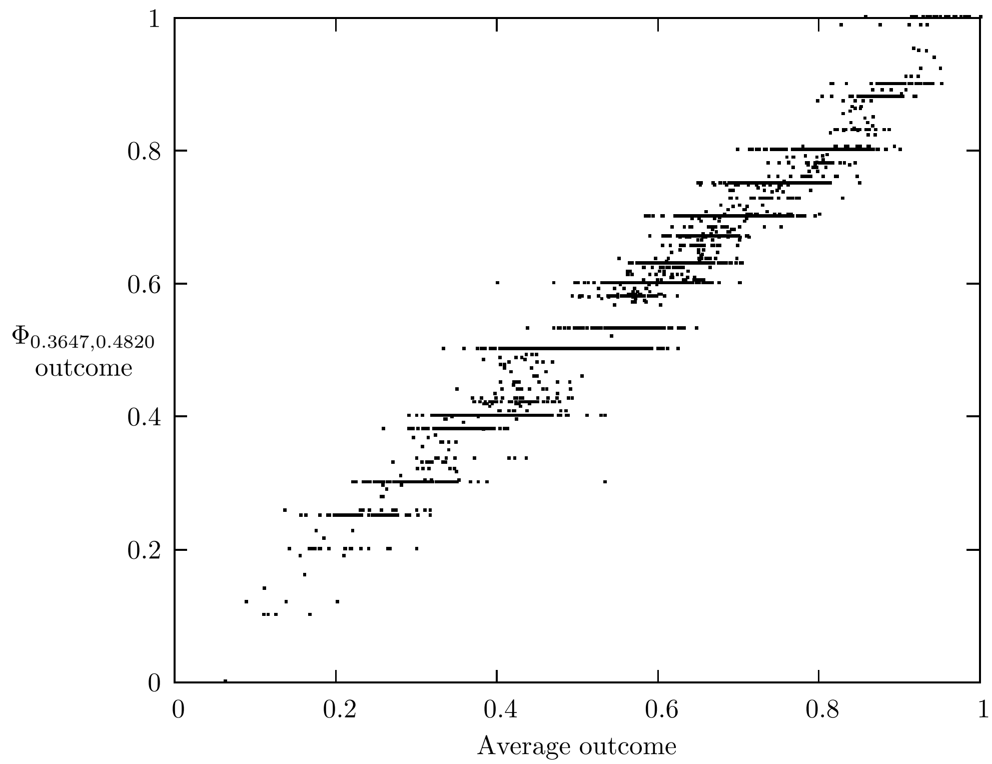
3.10 Generalizations to more dimensions

What if the votes and the outcome are in d-dimensional space, where $d > 1$ and each dimension is restricted between 0 and 1 (giving a hypercube)? For example, if the votes are $(0.3, 0.3)$, $(0.3, 0.3)$, $(0.3, 0.7)$, $(0.7, 0.3)$ and $(0.7, 0.7)$, what should the outcome be? This problem is very similar to single point estimation, the problem of finding a most "representative" point given a set of points.

The Average system is easily generalized to multiple dimensions by taking the average of each coordinate, effectively calculating the centroid, the center of mass given a set of unit masses. (Alternatively, one can imagine attaching Hookean springs of equal spring constants to each fixed input point, then gluing the other ends of the springs together; the glue point will come to rest at the centroid.) This generalization is equivalent to finding the point t that minimizes

$$\sum_{i=1}^{n} \mathrm{dist}(t, v_i)^2$$

Figure 3.4: $\Phi_{0.3647,0.4820}(\vec{v})$ vs. $\bar{v}$ scatterplot



where $\text{dist}(t, v_i) = \left(\sum_{j=1}^{d}(t_j - v_{ij})^2\right)^{1/2}$, the Euclidean distance between t and v_i (and the ℓ^2 norm of the vector $t - v_i$). (We must now conceptually allow all real numbers, not just rationals.) The resulting system is rotationally invariant and is equivalent to conducting d separate and independent Average elections, and the results above for strategic behavior under the one-dimensional Average system apply to the "election" for each coordinate. In particular, conducting a d-dimensional Average DSV election is equivalent to conducting d parallel one-dimensional Average DSV elections, and so gives a nonmanipulable system.

Generalizing Median to multiple dimensions by finding the median of each coordinate also results in a nonmanipulable system, but, unlike in the Average case, the result is not rotationally invariant (consider the points $(0, 0)$, $(0, 1)$ and $(1, 0)$, and then rotate them 45 degrees). But these are not the only ways to generalize the above one-dimensional systems.

First, the one-dimensional space between 0 and 1 (or -1 and 1) can be generalized in other ways than into hypercubes. For example, the d-dimensional space could be the zero-centered d-dimensional sphere of radius 1 (for example, $\{(x, y) \in \mathbb{R}^2 : x^2 + y^2 \leq 1\}$), and each voter can be assumed to prefer points with smaller Euclidean distance to his or her ideal point to those farther from it. Or it could be the d-dimensional surface of the $(d+1)$-dimensional sphere of radius 1 (for example, $\{(x, y) \in \mathbb{R}^2 : x^2 + y^2 = 1\}$). Perhaps even more interestingly, the outcome space could be the d-dimensional simplex (for example, $\{(x, y, z) \in \mathbb{R}^3 : x + y + z = 1\}$), which could describe the division of a limited resource among several uses (such as a committee allocating a fixed sum among budget items). Applying DSV to these problems may be addressed in future work.

Second, the Median system itself can be generalized in other interesting ways besides applying it independently to each coordinate. An alternative, and arguably superior, generalization of the Median system is found by finding the point t that minimizes

$$\sum_{i=1}^{n} \mathrm{dist}(t, v_i)$$

t is known as the Fermat–Weber point [58, 17]. The resulting system is rotationally invariant, unlike the median-in-each-dimension system. When $d > 1$, unlike in the one-dimensional case, it usually has a single optimum point even when n is even (the only exception is an even number of collinear points). Unfortunately, there is no computationally feasible exact algorithm to calculate the Fermat–Weber point in general [5], but numerical approximation is quite easy [56, 11].

The Fermat–Weber point does not change when a point v_i is moved farther away from t in the direction of the vector $\vec{tv_i}$ [54], so, in a sense, direction matters but not distance. Because of this property, a naïve Average-style strategy for manipulating this Fermat–Weber system fails, and any successful manipulation would have to move a sincere vote in some other direction.

Unfortunately, an insincere voter can indeed manipulate the Fermat–Weber point to move closer to his or her ideal outcome. Consider the square outcome space $\{(x, y) \in \mathbb{R}^2 : 0 \leq x, y \leq 1\}$. If there are four voters whose ideal outcomes are $(0.4, 0)$, $(0.4, 1)$, $(0.6, 0)$ and $(0.6, 1)$, and they all vote sincerely, then the Fermat–Weber point is $(0.5, 0.5)$. But then the fourth voter can insincerely vote $(0.4, 1)$, giving a Fermat–Weber point of $(0.4, 1)$, which is closer by Euclidean distance to the ideal $(0.6, 1)$ than is the previous outcome $(0.5, 0.5)$.

3.11 Summary of contributions

In this research, we have accomplished the following.

1. Described a rational voter's optimal strategy for Average aggregation.
2. Proved that at least one equilibrium will exist when all Average voters are optimally strategic.
3. Proved that all such equilibria will have the same Average outcome.
4. Defined a large new class of AAR DSV systems and showed that they are all immune to manipulation by insincere voters.
5. Provided an efficient algorithm to compute any AAR DSV outcome.
6. Used real-world data to choose an AAR DSV system that comes closest to matching Average outcomes over those data.
7. Showed that using the Fermat–Weber point as the outcome in a multi-dimensional space can be manipulated by insincere voters.

Chapter 4

Comparing Approval Strategies for DSV

DSV systems have been considered for plurality elections with the following goals in mind. First, it is known that voters will behave strategically, in that they will vote an insincere ballot if they are convinced a better outcome can be obtained. DSV systems attempt to distill the sincere cardinal preferences of a voter into the best possible strategic behavior for that voter. One obvious benefit is that the voter is motivated to provide sincere cardinal preferences, secure in the knowledge that his or her strategy will vote effectively according to those preferences. If DSV systems can derive provably best possible strategic voting behavior from those preferences, then providing other than sincere cardinal preferences would not be in the best interest of the voter. Second, reasoning about the best possible strategic behavior may be difficult and well beyond the means of some voters. A DSV system has the potential of affording the same advantages to all voters, so that no voter has disproportionate power in an election.

So a goal of DSV systems was to allow greater expression from a voter while providing that voter the best possible thinking in an election situation.

The above factors are also compelling arguments for elections using approval voting. Approval ballots are more expressive than plurality ballots, since favor can be expressed for multiple alternatives; in particular, a voter can approve both favorite and compromise alternatives. Thus, it may be the case that an effective approval ballot is, in some sense, more sincere than a plurality ballot of equal effectiveness. That is, there exists a (weakly, at least) sincere approval ballot that has the same or greater effectiveness as a correspondingly effective but insincere plurality ballot.

But Cranor's and Cytron's [23] approach to plurality strategy cannot be trivially extended to approval voting; that is, using their strategy to find the optimal plurality ballot and then converting it into an approval ballot simply by adding all alternatives preferred to the one already approved will not necessarily result in an optimal approval ballot. The problem is not simply finding the optimal alternative for which to vote but finding the optimal cutoff point between approved and disapproved alternatives; in other words, instead of maximizing a measure of effectiveness, a natural threshold above which approval is optimal must be found.

For example, if a voter prefers A to B to C, and C is judged to have far and away the best chance of winning with B the least likely to win, then the Cranor/Cytron approach to plurality strategy would recommend voting for A. Extending that plurality ballot trivially to an approval ballot by also approving all alternatives preferred to A results in approving only A, but it may well be that approving both A and B will give the voter a higher expected utility of the outcome. Indeed, "strategy A", presented in section 4.2 and later seen to be optimal under certain conditions (sections 4.4.6 and 4.5.1), would recommend approving both A and B.

Given extant work on DSV systems and on approval voting, it seems natural to examine the extent to which DSV systems can be used beneficially in approval elections. The following questions arise:

1. Is there a rationally optimal strategy for approval elections that is suitable under the same assumptions used for plurality elections (*i.e.*, Cranor/Cytron DSV pivot-probability computations and equations from social-choice literature concerning utilities)?

2. Under a wider set of assumptions, how do various approval voting strategies compare in terms of their cost, their effectiveness and their ability to elicit sincere cardinal preferences from voters?

In this chapter we begin the work of examining the above issues. A long-term goal of work in this area would be to establish strategies that work as well as any voter could in an approval election. While we may not fully reach that goal in this work, we propose methodologies for evaluating the effectiveness of an approval strategy and methods for comparing strategies.

4.1 The space of approval ballots

One might assume strategizing in an approval election to be more difficult than in a plurality election, since the number of allowed ballots is exponential in the number of alternatives rather than linear. But rarely does it make sense to vote an approval ballot that is not even weakly sincere by the definition in chapter 1, and in fact it never makes sense when all other ballots are known and fixed. More importantly from a DSV point of view, as we saw in section 1.4, only weakly sincere ballots make sense given only the information in an election state as set out in chapter 1.

There are only $k+1$ possible weakly sincere approval ballots over k alternatives (and fewer if abstention is ruled out or if the voter's cardinal preferences impose only a partial order on the alternatives). So only $O(k)$ ballots need to be considered for any one voter.

4.2 A new declared strategy for approval voting

Somewhat similar to approval strategy T (Figure 1.3) is strategy A [35], which essentially (when preferences and election state are tie-free) approves all alternatives preferred to the currently leading alternative, including that leading alternative if preferred to the currently second-place alternative. In other words, it places an approval cutoff next to the current leader on the side of the current second-placer. More precisely and generally, we define strategy A in Figure 4.1.

Notice that when there are no ties in the election state or in the voter's cardinal preferences, strategy A effectively approves alternative i if and only if $p_i \geq p_1$ when $p_1 > p_2$ and $p_i > p_1$ when $p_1 < p_2$.

Figure 4.1: Approval strategy A

For voter v voting in round $r+1$,

- for each alternative i:
  - find smallest y such that $p(v,i) \cdot |\mathrm{Top}_y(r)| \neq \mathrm{PSum}_y(v,r)$ ($y = k$ if none)
  - approve alternative i if and only if $p(v,i) \cdot |\mathrm{Top}_y(r)| > \mathrm{PSum}_y(v,r)$

4.3 Evaluating approval strategies

Our assumption is that all a voter ultimately cares about is the outcome of an election. Accordingly, perhaps the ideal measure of a particular strategy's effectiveness is, according to some distribution of sets of one focal voter's preferences and other voters' programs, the weighted average of the values of the outcomes that result when the focal voter uses that strategy. The difficulty of using such an approach to strategy evaluation is chiefly computational, but the approach also relies on finding such a distribution that is compellingly realistic.

4.3.1 Evaluating election states directly

Instead of trying to evaluate a strategy based on the eventual outcomes that obtain when it is used, then, one could simply find a way to evaluate the election states that immediately obtain. If, given a particular election state and set of preferences, one strategy produces a ballot that leads to an election state judged to be better by some measure than the election state found by another strategy, we could judge the first strategy superior to the second in that isolated case.

To illustrate this approach, here's one relatively simple way to evaluate a given election state: Chapter 6 of Merrill [38] inspires a way to estimate the expected value of the eventual election result by using only each alternative's current vote total. W_i, the probability of alternative i's winning, is estimated at $W_i = \frac{s_i^x}{\sum_{j=1}^{k} s_j^x}$ where s_i is alternative i's current vote total (alternative i's entry in the election state) and x is some constant greater than 1; Merrill (somewhat arbitrarily) suggests using $x = 2$.[1] W_i values estimated in this way can then be used to calculate an estimated

[1]Relatively speaking, using a smaller x spreads out the winning probability among the alternatives; using a larger x gives a relatively higher winning probability to the current leaders. Therefore it might make sense to vary x by increasing it as the election progresses, say by doubling it after each round.

value of the election from a particular voter's point of view by calculating $\sum_{i=1}^{k} p_i W_i$, where p_i is the cardinal preference that voter gives to alternative i.

For example, suppose the voter that submitted the program that executes next in a three-alternative approval DSV election rated the alternatives $\left[1, \frac{2}{3}, 0\right]$, and the current election state is $[14, 5, 10]$. Now the program must decide whether to vote $[1, 0, 0]$ or $[1, 1, 0]$, essentially choosing between the election states $[15, 5, 10]$ (if only the first alternative is approved) and $[15, 6, 10]$ (if the first two are approved).

Strategies T and J, described in chapter 1, disagree on the best ballot to cast in this situation; T recommends $[1, 0, 0]$ and J recommends $[1, 1, 0]$. So, according to this election-state-evaluation approach, T makes the right choice if $[15, 5, 10]$ is estimated to have a superior value to $[15, 6, 10]$ and J makes the right choice if the opposite is true.

Calculating the estimated value of each election state is straightforward. For $[15, 5, 10]$, the estimated probabilites of winning (assuming $x = 2$) are

$$\frac{\left[15^2, 5^2, 10^2\right]}{15^2 + 5^2 + 10^2} = \left[\frac{9}{14}, \frac{1}{14}, \frac{4}{14}\right] \approx [0.643, 0.071, 0.286]$$

and so the estimated value of this election state is[2]

$$\left[\frac{9}{14}, \frac{1}{14}, \frac{4}{14}\right] \cdot \left[1, \frac{2}{3}, 0\right] = \frac{9}{14} + \frac{1}{21} = \frac{29}{42} \approx 0.6905$$

For $[15, 6, 10]$, the estimated probabilites of winning are

$$\frac{\left[15^2, 6^2, 10^2\right]}{15^2 + 6^2 + 10^2} = \left[\frac{225}{361}, \frac{36}{361}, \frac{100}{361}\right] \approx [0.623, 0.100, 0.277]$$

and so the estimated value of this election state is

$$\left[\frac{225}{361}, \frac{36}{361}, \frac{100}{361}\right] \cdot \left[1, \frac{2}{3}, 0\right] = \frac{225}{361} + \frac{24}{361} = \frac{249}{361} \approx 0.6898$$

[2] The symbol $\cdot$ denotes vector dot product: $[a_1, a_2, \ldots a_n] \cdot [b_1, b_2, \ldots b_n] = [a_1 b_1, a_2 b_2, \ldots a_n b_n] = \sum_{i=1}^{n} a_i b_i$.

Since $0.6905 > 0.6898$, $[15, 5, 10]$ is judged to be a better election state for the voter than $[15, 6, 10]$ and so strategy T is the superior performer in this specific case.

A similar example situation shows that strategy T does not always perform better for a voter than strategy J by this measure. Suppose the next program's voter again rates the alternatives $\left[1, \frac{2}{3}, 0\right]$, but now the current election state is $[9, 5, 15]$. This time, again, strategy T recommends the ballot $[1, 0, 0]$, leading to $[10, 5, 15]$, and J recommends $[1, 1, 0]$, leading to $[10, 6, 15]$. For $[10, 5, 15]$, the estimated probabilites of winning, again assuming $x = 2$, are

$$\frac{\left[10^2, 5^2, 15^2\right]}{10^2 + 5^2 + 15^2} = \left[\frac{4}{14}, \frac{1}{14}, \frac{9}{14}\right] \approx [0.286, 0.071, 0.643]$$

and so the estimated value of this election state is

$$\left[\frac{4}{14}, \frac{1}{14}, \frac{9}{14}\right] \cdot \left[1, \frac{2}{3}, 0\right] = \frac{4}{14} + \frac{1}{21} = \frac{1}{3} \approx 0.3333$$

For $[10, 6, 15]$, the estimated probabilites of winning are

$$\frac{\left[10^2, 6^2, 15^2\right]}{10^2 + 6^2 + 15^2} = \left[\frac{100}{361}, \frac{36}{361}, \frac{225}{361}\right] \approx [0.277, 0.100, 0.623]$$

and so the estimated value of this election state is

$$\left[\frac{100}{361}, \frac{36}{361}, \frac{225}{361}\right] \cdot \left[1, \frac{2}{3}, 0\right] = \frac{100}{361} + \frac{24}{361} = \frac{124}{361} \approx 0.3435$$

Since $0.3435 > 0.3333$, $[10, 6, 15]$ is judged to be a better election state for the voter than $[10, 5, 15]$ and so strategy J now performs better in this case. Therefore, according to this election-state measure, neither strategy T nor J outperforms the other in every possible situation; both sometimes give inferior ballot recommendations.

This method of estimating alternatives' eventual winning probabilities by magnifying the vote totals of leading alternatives illustrates one way to evaluate strategies by evaluating election states. It essentially takes rules of thumb, which have been described in the literature and are easy to describe in English, and evaluates them according to a standard which has appeared in the literature and is mathematically precise and easily motivated and parameterized. But, still, it may seem somewhat arbitrary, not directly taking into account a probability distribution of future ballots. More compelling election-state-evaluation approaches are possible.

4.3.2 Evaluating election states by looking ahead

When a program is run during a DSV election, it has information on the ballots previously cast but cannot predict ballots to come with certainty. Any ballot that it chooses may turn out to have been the most effective one and it may not. The best that can be hoped for, therefore, is to maximize the expected value of the election for its voter, and to do so requires the program to make an assumption about the probabilities of the various possible outcomes.

Say an approval DSV election is run in ballot-by-ballot mode with only one round. A particular voter's program is about to be run; the ballot vector contains the ballots of those voters whose programs have already been run. The current program has no information about the ballots to come. One could imagine the situation as a large decision tree where each node corresponds to a set of voted ballots and a resulting election state. Each node's children are the states reachable from it by voting some specific ballot. The root of the tree is the current election state, and the leaves are those election states that result when all ballots have been cast and a winner can be crowned. The question then becomes, How likely is each leaf to be the one corresponding to the outcome eventually reached?[3] Knowing these probabilities, the program can calculate the expected value of the election for its voter for each possible ballot by weighting the outcomes according to their assumed probabilities of occurring and vote the ballot that maximizes that expected value.

There are at least two reasonable ways to estimate the leaf outcome probabilities. One, the agnostic approach, is to assume that at each node in the tree, the relevant program chooses each possible ballot with equal likelihood. All leaf outcomes reachable from the root node are therefore assumed to be equally likely to obtain. Another, the statistical approach, is to consider the ballots seen so far as a representative sample of the final set of ballots and assume that the probability of each future ballot approving an alternative is equal to the proportion of ballots seen so far that

[3]This question, in which we know a set of intermediate vote totals and ask about likely eventual totals, is much like the inverse of the "ballot problem" [50], in which we know the eventual vote total for each alternative and ask about intermediate vote totals as the ballots are counted one by one.

approve that alternative. (These two approaches could be combined; for example, programs run early in the DSV round might assume all outcomes to be relatively equally probable while those coming later in the round might tend toward the statistical-sample approach.)

Once an expected value of the election can be computed for each possible ballot, they can be directly compared. By this standard, a strategy is better than another in a specific instance if it chooses a ballot with a higher expected value of the election. Similarly, a strategy *election-state-dominates* another if it never chooses a ballot that leads to an election state with a lower expected value of the election than the other, no matter the current election state.

As an example, consider a single-round approval DSV election in ballot-by-ballot mode with three alternatives and 26 voters. Twenty-four of those voters' programs have already executed and voted their ballots, resulting in the election state $[10, 12, 12]$; therefore $\frac{5}{12}$ of those voters approved a_1 and $\frac{1}{2}$ of them approved each of a_2 and a_3. The penultimate voter's cardinal preferences are $[1, \frac{2}{5}, 0]$, so his or her program needs only to consider the ballots $[1, 0, 0]$ and $[1, 1, 0]$. If $[1, 0, 0]$ is chosen, the new election state will be $[11, 12, 12]$, and the winner of the election will depend on the ballot voted by the ultimate voter's program. There are eight possibilities:

ultimate voter's ballot	resulting final election state	value of winner(s) to penultimate voter	estimated probability of occurrence	expected value
$[0, 0, 0]$	$[11, 12, 12]$	$\frac{1}{5} = 0.2$	$\frac{7}{48} \approx 0.146$	$\frac{7}{240} \approx 0.029$
$[0, 0, 1]$	$[11, 12, 13]$	0	$\frac{7}{48} \approx 0.146$	0
$[0, 1, 0]$	$[11, 13, 12]$	$\frac{2}{5} = 0.4$	$\frac{7}{48} \approx 0.146$	$\frac{7}{120} \approx 0.058$
$[0, 1, 1]$	$[11, 13, 13]$	$\frac{1}{5} = 0.2$	$\frac{7}{48} \approx 0.146$	$\frac{7}{240} \approx 0.029$
$[1, 0, 0]$	$[12, 12, 12]$	$\frac{7}{15} \approx 0.467$	$\frac{5}{48} \approx 0.104$	$\frac{7}{144} \approx 0.049$
$[1, 0, 1]$	$[12, 12, 13]$	0	$\frac{5}{48} \approx 0.104$	0
$[1, 1, 0]$	$[12, 13, 12]$	$\frac{2}{5} = 0.4$	$\frac{5}{48} \approx 0.104$	$\frac{1}{24} \approx 0.042$
$[1, 1, 1]$	$[12, 13, 13]$	$\frac{1}{5} = 0.2$	$\frac{5}{48} \approx 0.104$	$\frac{1}{48} \approx 0.021$
			total expected value:	$\frac{41}{180} \approx 0.2278$

When two or more alternatives tie for the win, the tie is assumed to be broken with equal probability; for example, when the final election state is $[11, 13, 13]$, a_2 and a_3 tie for the win and so the penultimate voter's expected value of this outcome is $\frac{1}{2} \cdot \frac{2}{5} + \frac{1}{2} \cdot 0 = \frac{1}{5}$. The estimated probability of each of the ultimate voter's ballots is calculated by assuming that the ultimate voter approves a_i with probability equal to the proportion of a_i's approval among ballots seen so far. So the probability of the ultimate voter's ballot being $[0,1,0]$ equals $\frac{7}{12}$ (the proportion of ballots so far that did *not* approve a_1) times $\frac{1}{2}$ (the proportion of ballots so far that *did* approve a_2) times $\frac{1}{2}$ (the proportion of ballots so far that did *not* approve a_3). The total of the product column is the estimated value of the final outcome from the penultimate voter's point of view.

On the other hand, if the penultimate voter's program chooses the ballot $[1,1,0]$, the new election state will be $[11,13,12]$, and the possibilities become

ultimate voter's ballot	resulting final election state	value of winner(s) to penultimate voter	estimated probability of occurrence	expected value
$[0,0,0]$	$[11,13,12]$	$\frac{2}{5}=0.4$	$\frac{7}{48}\approx 0.146$	$\frac{7}{120}\approx 0.058$
$[0,0,1]$	$[11,13,13]$	$\frac{1}{5}=0.2$	$\frac{7}{48}\approx 0.146$	$\frac{7}{240}\approx 0.029$
$[0,1,0]$	$[11,14,12]$	$\frac{2}{5}=0.4$	$\frac{7}{48}\approx 0.146$	$\frac{7}{120}\approx 0.058$
$[0,1,1]$	$[11,14,13]$	$\frac{2}{5}=0.4$	$\frac{7}{48}\approx 0.146$	$\frac{7}{120}\approx 0.058$
$[1,0,0]$	$[12,13,12]$	$\frac{2}{5}=0.4$	$\frac{5}{48}\approx 0.104$	$\frac{1}{24}\approx 0.042$
$[1,0,1]$	$[12,13,13]$	$\frac{1}{5}=0.2$	$\frac{5}{48}\approx 0.104$	$\frac{1}{48}\approx 0.021$
$[1,1,0]$	$[12,14,12]$	$\frac{2}{5}=0.4$	$\frac{5}{48}\approx 0.104$	$\frac{1}{24}\approx 0.042$
$[1,1,1]$	$[12,14,13]$	$\frac{2}{5}=0.4$	$\frac{5}{48}\approx 0.104$	$\frac{1}{24}\approx 0.042$
			total expected value:	$\frac{7}{20}=0.3500$

The penultimate voter's estimated value of the final outcome is greater when voting $[1,1,0]$ than $[1,0,0]$, so in this situation a strategy that recommended $[1,1,0]$ would be judged better than one that recommended $[1,0,0]$. If the first strategy were never similarly judged inferior to the second strategy for any such situation, then it would be said to election-state-dominate the second.

This branching-probabilities approach to computing the expected value of an election state results in a very similar distribution of expected vote totals as Cranor's [22] pivot-probability approach. But instead of taking the parameter S^2 as an estimate of uncertainty, the branching-probabilities approach requires knowing how many ballots are still to come. This knowledge is possible when running a single-round ballot-by-ballot DSV election, as was assumed above. However, the same approach can be used for multi-round and batch DSV elections in at least two ways. First, it could be assumed that the current round is the last one and proceed accordingly. Second, and arguably more satisfying, the calculation could be carried out while allowing the number of future ballots to approach infinity—roughly equivalent to having S^2 approach zero in the pivot-probability approach. Note that this approach will in the limit assign to any election state with a clear leading alternative an expected value equal to the value of that alternative, but one election state will usually have a larger expected value than another as the number of future ballots is increased, even if the same alternative leads in both election states.

For example, imagine that the 20th voter in a three-alternative approval DSV election has the cardinal preferences $\left[1, \frac{1}{3}, 0\right]$ and is about to vote; the current election state is $[9, 6, 13]$. The voter's declared strategy considers voting either the ballot $[1, 0, 0]$, leading to the election state $[10, 6, 13]$, or the ballot $[1, 1, 0]$, leading to the election state $[10, 7, 13]$. When the number of future ballots is allowed to approach infinity, the expected value of each of the two considered election states approaches 0, making no distinction between the two. But if it can be shown that the expected value of $[10, 7, 13]$ is always greater than that of $[10, 6, 13]$ as the number of future ballots approaches infinity, then, in this case, the ballot $[1, 1, 0]$ could be said to dominate the ballot $[1, 0, 0]$.

But how can the effectiveness of two different strategies be compared when neither dominates the other by this measure? One approach would make more assumptions regarding the likelihoods of all possible election situations and calculate a weighted average of expected election values, but it may be difficult to make the problem computationally feasible without making extremely restrictive assumptions.

When neither of two strategies election-state-dominates the other, so that strategy α is better than strategy β in some situations and β is better than α in others, a new "überstrategy" can be created that is superior to both. Simply define strategy $\alpha\beta$ as the strategy that, in any particular situation, uses the election-state metric to decide whether α's or β's recommended ballot has a better expected election result and follow that recommendation. The resulting strategy $\alpha\beta$ necessarily election-state-dominates each of α and β. The computational cost of such a constructed überstrategy may, however, be substantial compared to that of the component strategies.

4.4 General results using the Merrill election-state metric

We've seen some approaches to comparing relative effectiveness of approval strategies in specific situations, but it may not be obvious how to use these election-state metrics to compare two strategies against each other in general. Nevertheless, it is possible to prove some relatively general results using the Merrill election-state metric defined above.

In this section we compare strategies by looking at decisions from one particular focal voter's point of view at a time, so the relevant voter v and the current round r can be treated as constants. As a notational convenience, we use p_i to mean $p(v,i)$, the focal voter's cardinal preference for alternative i, and s_i to mean $s(i, r-1)$, alternative i's vote total in the current visible election state.

4.4.1 Comparing strategies A and T in the three-alternative case

A compelling result can be obtained by focusing on elections with three alternatives[4] and comparing strategies A and T. We narrow consideration to situations in which no two alternatives are tied in the election state (further assuming with no loss of generality that alternative 1 is in the lead followed by alternative 2 and then alternative 3) and the focal voter has strictly ordered cardinal preferences.

Theorem 4.4.1. *In a three-alternative election with current election state $\vec{s} = [s_1, s_2, s_3]$ where $s_1 > s_2 > s_3$ and focal voter's cardinal preferences $\vec{p} = [p_1, p_2, p_3]$ where $p_i \neq p_j$ for $i \neq j$, employing strategy A will only lead the focal voter to a worse next election state than strategy T as measured using the Merrill election-state metric with parameter x when $p_2 > p_3 > p_1$ and $\frac{p_2 - p_3}{p_3 - p_1} > \left(\frac{s_1}{s_2+1}\right)^x$.*

Proof. When no ties exist in $\vec{s}$ or $\vec{p}$, the ballots recommended by strategies A and T are fully determined by the orderings of the alternatives in $\vec{s}$ and $\vec{p}$. When $s_1 > s_2 > s_3$, strategies A and T disagree in only one of the six possible orderings of $\vec{p}$:

[4] Note that in two-alternative elections, all of the approval strategies considered in this research are equivalent: approve the preferred of the two alternatives. In fact, this strategy dominates all others in the strict game-theoretical sense; see Brams and Fishburn [14, p. 25].

voter's preferences	strategy A's recommended ballot	strategy T's recommended ballot
$p_1 > p_2 > p_3$	$[1,0,0]$	$[1,0,0]$
$p_1 > p_3 > p_2$	$[1,0,0]$	$[1,0,0]$
$p_2 > p_1 > p_3$	$[0,1,0]$	$[0,1,0]$
$\boldsymbol{p_2 > p_3 > p_1}$	$[\mathbf{0},\mathbf{1},\mathbf{1}]$	$[\mathbf{0},\mathbf{1},\mathbf{0}]$
$p_3 > p_1 > p_2$	$[1,0,1]$	$[1,0,1]$
$p_3 > p_2 > p_1$	$[0,1,1]$	$[0,1,1]$

So, when $s_1 > s_2 > s_3$, strategies A and T can only lead to different next election states when $p_2 > p_3 > p_1$. To compare them according to the Merrill metric, we use the new election state found by voting the ballot recommended by each strategy, calculate each alternative's new estimated probability of winning, and use them to calculate the voter's estimated value of the election. The new estimated value of the election that results from voting $[0,1,1]$ is

$$V_{[0,1,1]} = [p_1, p_2, p_3] \cdot \frac{[s_1^x, (s_2+1)^x, (s_3+1)^x]}{s_1^x + (s_2+1)^x + (s_3+1)^x} = \frac{p_1 s_1^x + p_2(s_2+1)^x + p_3(s_3+1)^x}{s_1^x + (s_2+1)^x + (s_3+1)^x}$$

and $V_{[0,1,0]}$ is calculated similarly. To find when strategy A leads to a worse expectation of the election result than strategy T in terms of $\vec{s}$, $\vec{p}$ and x, we can set $V_{[0,1,1]} < V_{[0,1,0]}$:

$$\frac{p_1 s_1^x + p_2(s_2+1)^x + p_3(s_3+1)^x}{s_1^x + (s_2+1)^x + (s_3+1)^x} < \frac{p_1 s_1^x + p_2(s_2+1)^x + p_3 s_3^x}{s_1^x + (s_2+1)^x + s_3^x}$$

To simplify the derivation somewhat, we define $A = p_1 s_1^x + p_2(s_2+1)^x$ and $B = s_1^x + (s_2+1)^x$, so the previous inequality becomes

$$\frac{A + p_3(s_3+1)^x}{B + (s_3+1)^x} < \frac{A + p_3 s_3^x}{B + s_3^x}$$

Since $s_i \geq 0$ for all i, the denominators are positive, and so

$$\left(A + p_3(s_3+1)^x\right)\left(B + s_3^x\right) < \left(A + p_3 s_3^x\right)\left(B + (s_3+1)^x\right)$$

Expanding factors,

$$AB + As_3^x + p_3(s_3+1)^x B + p_3(s_3+1)^x s_3^x < AB + A(s_3+1)^x + p_3 s_3^x B + p_3 s_3^x (s_3+1)^x$$

Canceling the ABs and substituting for A and B gives

$$(p_1 s_1^x + p_2(s_2+1)^x)s_3^x + p_3(s_3+1)^x(s_1^x + (s_2+1)^x) + p_3(s_3+1)^x s_3^x <$$
$$(p_1 s_1^x + p_2(s_2+1)^x)(s_3+1)^x + p_3 s_3^x(s_1^x + (s_2+1)^x) + p_3 s_3^x(s_3+1)^x$$

Expanding factors and then canceling terms once again,

$$p_1 s_1^x s_3^x + p_2(s_2+1)^x s_3^x + p_3 s_1^x(s_3+1)^x + p_3(s_2+1)^x(s_3+1)^x <$$
$$p_1 s_1^x(s_3+1)^x + p_2(s_2+1)^x(s_3+1)^x + p_3 s_1^x s_3^x + p_3(s_2+1)^x s_3^x$$

Separating s_1^x terms and $(s_2+1)^x$ terms gives

$$p_3 s_1^x(s_3+1)^x - p_3 s_1^x s_3^x - p_1 s_1^x(s_3+1)^x + p_1 s_1^x s_3^x <$$
$$p_2(s_2+1)^x(s_3+1)^x - p_2(s_2+1)^x s_3^x - p_3(s_2+1)^x(s_3+1)^x + p_3(s_2+1)^x s_3^x$$

And, factoring each side,

$$(p_3 - p_1)s_1^x((s_3+1)^x - s_3^x) < (p_2 - p_3)(s_2+1)^x((s_3+1)^x - s_3^x)$$

Since $x > 1$, it is always true that $(s_3+1)^x - s_3^x > 0$, and so

$$(p_3 - p_1)s_1^x < (p_2 - p_3)(s_2+1)^x$$

Each of the steps that lead from $V_{[0,1,1]} < V_{[0,1,0]}$ to $(p_3 - p_1)s_1^x < (p_2 - p_3)(s_2+1)^x$ is reversible—each implication is two-way. So, if and only if $p_2 > p_3 > p_1$ and $\frac{p_2 - p_3}{p_3 - p_1} > \left(\frac{s_1}{s_2+1}\right)^x$, strategy T leads to a better next election state than strategy A. Conversely, it can be easily seen with similar logic that strategy A is superior when $p_2 > p_3 > p_1$ and $\frac{p_2 - p_3}{p_3 - p_1} < \left(\frac{s_1}{s_2+1}\right)^x$. (Recall that both strategies result in the same ballots when it is not the case that $p_2 > p_3 > p_1$ and so perform identically by any metric.) □

Corollary 4.4.2. *When the exponent x is taken to approach infinity and $s_1 > s_2 + 1$, strategy A dominates strategy T.*

Proof. If $s_1 > s_2 + 1$, then $\left(\frac{s_1}{s_2+1}\right)^x$ goes to infinity as $x \to \infty$, so strategy T dominates strategy A only when $\frac{p_3-p_1}{p_2-p_3} < 0$, which is impossible when $p_2 > p_3 > p_1$. □

Corollary 4.4.2 will be generalized to elections with more alternatives in section 4.4.6 below.

Corollary 4.4.3. *When $p_3 > \frac{p_1+p_2}{2}$, strategy A dominates strategy T.*

Proof. When it is not the case that $p_2 > p_3 > p_1$, strategies A and T agree. If $p_2 > p_3 > p_1$ and $p_3 > \frac{p_1+p_2}{2}$, then $p_3 - p_1 > p_2 - p_3$ and $\frac{p_2-p_3}{p_3-p_1} < 1$, and since $\left(\frac{s_1}{s_2+1}\right)^x \geq 1$, it is impossible to satisfy $\frac{p_2-p_3}{p_3-p_1} > \left(\frac{s_1}{s_2+1}\right)^x$, so strategy T cannot do better than strategy A. □

Notice that $p_3 > \frac{p_1+p_2}{2}$ is true if and only if $p_3 > \frac{p_1+p_2+p_3}{3}$ is true, so this corollary means that strategy A is never worse for a voter than strategy T when p_3 is higher than the voter's average cardinal preference.

To lend some intuition to these results, notice that it is possible for strategy T to do better than strategy A according to the Merrill metric only under limited circumstances. For example, say $\vec{p} = \left[0, 1, \frac{1}{12}\right]$ and $\vec{s} = [12, 9, 8]$. Then strategy A is better if $x > \frac{\ln 11}{\ln 6 - \ln 5} \approx 13.152$ and strategy T is better otherwise. Or, say $\vec{p} = [0, 1, p_3]$, $\vec{s} = [12, 9, 8]$ and $x = 2$. Then strategy A is better if $p_3 > \frac{25}{61} \approx 0.4098$ and strategy T is better otherwise. Generally and loosely speaking, strategy T can rate as better than strategy A only when s_1 and s_2 are relatively close, x is relatively small and p_3 is relatively close to p_1 compared to p_2. (And, of course, even then only when $p_2 > p_3 > p_1$.)

4.4.2 Comparing strategies A and J in the three-alternative case

A similar result can be obtained by focusing on elections with three alternatives and comparing strategies A and J. We again narrow consideration to situations in which no two alternatives are tied in the election state and the focal voter has strictly ordered cardinal preferences; further, we stipulate that none of the cardinal preferences should equal their average.

Theorem 4.4.4. *In a three-alternative election with current election state $\vec{s} = [s_1, s_2, s_3]$ where $s_1 > s_2 > s_3$ and focal voter's cardinal preferences $\vec{p} = [p_1, p_2, p_3]$ where $p_i \neq \frac{p_1+p_2+p_3}{3}$ for all i and $p_i \neq p_j$ for $i \neq j$, employing strategy A will only lead the focal voter to a worse next election state* *than strategy J as measured using the Merrill election-state metric with parameter x when* $p_2 > \frac{p_1+p_2}{2} > p_3 > p_1$ *and* $\frac{p_2-p_3}{p_3-p_1} > \left(\frac{s_1}{s_2+1}\right)^x$ *or when* $p_1 > p_3 > \frac{p_1+p_2}{2} > p_2$ *and* $\frac{p_1-p_3}{p_3-p_2} < \left(\frac{s_2}{s_1+1}\right)^x$.

Proof. When no ties exist in $\vec{s}$ or $\vec{p}$, the ballots recommended by strategy J are fully determined by the orderings of the alternatives in $\vec{s}$ and the relative ordering of the entries in $\vec{p}$ and their average $\bar{p} = \frac{p_1+p_2+p_3}{3}$. When $s_1 > s_2 > s_3$, strategies A and J disagree in two of the twelve possible orderings of $\vec{p}$ and $\bar{p}$:

voter's preferences	strategy A recommends	strategy J recommends
$p_1 > \bar{p} > p_2 > p_3$	$[1, 0, 0]$	$[1, 0, 0]$
$p_1 > p_2 > \bar{p} > p_3$	$[1, 0, 0]$	$[1, 0, 0]$
$p_1 > \bar{p} > p_3 > p_2$	$[1, 0, 0]$	$[1, 0, 0]$
$\boldsymbol{p_1 > p_3 > \bar{p} > p_2}$	$\mathbf{[1, 0, 0]}$	$\mathbf{[1, 0, 1]}$
$p_2 > \bar{p} > p_1 > p_3$	$[0, 1, 0]$	$[0, 1, 0]$
$p_2 > p_1 > \bar{p} > p_3$	$[0, 1, 0]$	$[0, 1, 0]$
$\boldsymbol{p_2 > \bar{p} > p_3 > p_1}$	$\mathbf{[0, 1, 1]}$	$\mathbf{[0, 1, 0]}$
$p_2 > p_3 > \bar{p} > p_1$	$[0, 1, 1]$	$[0, 1, 1]$
$p_3 > \bar{p} > p_1 > p_2$	$[1, 0, 1]$	$[1, 0, 1]$
$p_3 > p_1 > \bar{p} > p_2$	$[1, 0, 1]$	$[1, 0, 1]$
$p_3 > \bar{p} > p_2 > p_1$	$[0, 1, 1]$	$[0, 1, 1]$
$p_3 > p_2 > \bar{p} > p_1$	$[0, 1, 1]$	$[0, 1, 1]$

So, when $s_1 > s_2 > s_3$, strategies A and J can only lead to different next election states in the two opposite cases $p_1 > p_3 > \bar{p} > p_2$ and $p_2 > \bar{p} > p_3 > p_1$. We compare A and J according to the Merrill metric, as in section 4.4.1, considering the two cases separately.

Case 1: $p_1 > p_3 > \bar{p} > p_2$. Strategy A recommends the vote $[1, 0, 0]$ but strategy J recommends $[1, 0, 1]$. The new estimated value of the election that results from voting $[1, 0, 0]$ is

$$V_{[1,0,0]} = [p_1, p_2, p_3] \cdot \frac{[(s_1+1)^x, s_2^x, s_3^x]}{(s_1+1)^x + s_2^x + s_3^x} = \frac{p_1(s_1+1)^x + p_2 s_2^x + p_3 s_3^x}{(s_1+1)^x + s_2^x + s_3^x}$$

and $V_{[1,0,1]}$ is calculated similarly. To find when strategy A leads to a worse expectation of the election result than strategy J in terms of $\vec{s}$, $\vec{p}$ and x, we can set $V_{[1,0,0]} < V_{[1,0,1]}$:

$$\frac{p_1(s_1+1)^x + p_2 s_2^x + p_3 s_3^x}{(s_1+1)^x + s_2^x + s_3^x} < \frac{p_1(s_1+1)^x + p_2 s_2^x + p_3(s_3+1)^x}{(s_1+1)^x + s_2^x + (s_3+1)^x}$$

Following a derivation similar to the one in section 4.4.1, we find this inequality equivalent to

$$(p_1 - p_3)(s_1+1)^x < (p_3 - p_2)s_2^x$$

Case 2: $p_2 > \bar{p} > p_3 > p_1$. Strategy A recommends the vote $[0,1,1]$ but strategy J recommends $[0,1,0]$. We found in section 4.4.1 that $[0,1,1]$ is worse for the voter than $[0,1,0]$ if and only if

$$\frac{p_2 - p_3}{p_3 - p_1} > \left(\frac{s_1}{s_2+1}\right)^x$$

So, if and only if either $p_1 > p_3 > \bar{p} > p_2$ and $\frac{p_3-p_2}{p_1-p_3} > \left(\frac{s_1+1}{s_2}\right)^x$ or $p_2 > \bar{p} > p_3 > p_1$ and $\frac{p_2-p_3}{p_3-p_1} > \left(\frac{s_1}{s_2+1}\right)^x$, strategy J leads to a better next election state than strategy A. Conversely, it can be easily seen with similar logic that strategy A is superior when either $p_1 > p_3 > \bar{p} > p_2$ and $\frac{p_3-p_2}{p_1-p_3} < \left(\frac{s_1+1}{s_2}\right)^x$ or $p_2 > p_3 > p_1$ and $\frac{p_2-p_3}{p_3-p_1} < \left(\frac{s_1}{s_2+1}\right)^x$. $\square$

4.4.3 Comparing strategies T and J in the three-alternative case

A similar result can be obtained by focusing on elections with three alternatives and comparing strategies T and J. We again narrow consideration to situations in which no two alternatives are tied in the election state and none of the focal voter's cardinal preferences equals their average or another cardinal preference.

Theorem 4.4.5. *In a three-alternative election with current election state $\vec{s} = [s_1, s_2, s_3]$ where $s_1 > s_2 > s_3$ and focal voter's cardinal preferences $\vec{p} = [p_1, p_2, p_3]$ where $p_i \neq \frac{p_1+p_2+p_3}{3}$ for all i and $p_i \neq p_j$ for $i \neq j$, employing strategy T will only lead the focal voter to a worse next election state than strategy J as measured using the Merrill election-state metric with parameter x when $p_1 > p_3 > \bar{p} > p_2$ and $\frac{p_1-p_3}{p_3-p_2} < \left(\frac{s_2}{s_1+1}\right)^x$ or when $p_2 > p_3 > \bar{p} > p_1$.*

Proof. When no ties exist in $\vec{s}$ or $\vec{p}$, the ballots recommended by strategies T and J are fully determined by the orderings of the alternatives in $\vec{s}$ and the relative ordering of the entries in $\vec{p}$ and their average $\bar{p} = \frac{p_1+p_2+p_3}{3}$. When $s_1 > s_2 > s_3$, strategies T and J disagree in two of the twelve possible orderings of $\vec{p}$ and $\bar{p}$:

voter's preferences	strategy T recommends	strategy J recommends
$p_1 > \bar{p} > p_2 > p_3$	$[1, 0, 0]$	$[1, 0, 0]$
$p_1 > p_2 > \bar{p} > p_3$	$[1, 0, 0]$	$[1, 0, 0]$
$p_1 > \bar{p} > p_3 > p_2$	$[1, 0, 0]$	$[1, 0, 0]$
$\boldsymbol{p_1 > p_3 > \bar{p} > p_2}$	$[\mathbf{1, 0, 0}]$	$[\mathbf{1, 0, 1}]$
$p_2 > \bar{p} > p_1 > p_3$	$[0, 1, 0]$	$[0, 1, 0]$
$p_2 > p_1 > \bar{p} > p_3$	$[0, 1, 0]$	$[0, 1, 0]$
$p_2 > \bar{p} > p_3 > p_1$	$[0, 1, 0]$	$[0, 1, 0]$
$\boldsymbol{p_2 > p_3 > \bar{p} > p_1}$	$[\mathbf{0, 1, 0}]$	$[\mathbf{0, 1, 1}]$
$p_3 > \bar{p} > p_1 > p_2$	$[1, 0, 1]$	$[1, 0, 1]$
$p_3 > p_1 > \bar{p} > p_2$	$[1, 0, 1]$	$[1, 0, 1]$
$p_3 > \bar{p} > p_2 > p_1$	$[0, 1, 1]$	$[0, 1, 1]$
$p_3 > p_2 > \bar{p} > p_1$	$[0, 1, 1]$	$[0, 1, 1]$

So, when $s_1 > s_2 > s_3$, strategies T and J can only lead to different next election states in the two cases $p_1 > p_3 > \bar{p} > p_2$ and $p_2 > p_3 > \bar{p} > p_1$. We compare T and J according to the Merrill metric, considering the two cases separately.

Case 1: $p_1 > p_3 > \bar{p} > p_2$. Strategy T recommends the vote $[1, 0, 0]$ but strategy J recommends $[1, 0, 1]$. The new estimated value of the election that results from voting $[1, 0, 0]$ is

$$V_{[1,0,0]} = [p_1, p_2, p_3] \cdot \frac{[(s_1+1)^x, s_2^x, s_3^x]}{(s_1+1)^x + s_2^x + s_3^x} = \frac{p_1(s_1+1)^x + p_2 s_2^x + p_3 s_3^x}{(s_1+1)^x + s_2^x + s_3^x}$$

and $V_{[1,0,1]}$ is calculated similarly. To find when strategy A leads to a worse expectation of the election result than strategy J in terms of $\vec{s}$, $\vec{p}$ and x, we can set $V_{[1,0,0]} < V_{[1,0,1]}$:

$$\frac{p_1(s_1+1)^x + p_2 s_2^x + p_3 s_3^x}{(s_1+1)^x + s_2^x + s_3^x} < \frac{p_1(s_1+1)^x + p_2 s_2^x + p_3 (s_3+1)^x}{(s_1+1)^x + s_2^x + (s_3+1)^x}$$

Following a derivation similar to the one in section 4.4.1, we find this inequality equivalent to

$$(p_1 - p_3)(s_1 + 1)^x < (p_3 - p_2)s_2^x$$

Case 2: $p_2 > p_3 > \bar{p} > p_1$. Strategy J recommends the vote $[0, 1, 1]$ but strategy T recommends $[0, 1, 0]$. We found in section 4.4.1 that $[0, 1, 0]$ is worse for the voter than $[0, 1, 1]$ if and only if

$$\frac{p_2 - p_3}{p_3 - p_1} < \left(\frac{s_1}{s_2 + 1}\right)^x$$

Notice that if $p_3 > \bar{p} = \frac{p_1+p_2+p_3}{3}$, then

$$3p_3 > p_1 + p_2 + p_3$$

$$2p_3 > p_1 + p_2$$

$$p_3 - p_1 > p_2 - p_3$$

and, if $p_3 > p_1$, then

$$1 > \frac{p_2 - p_3}{p_3 - p_1}$$

But when $s_1 \geq s_2 + 1$ and $x > 0$, it must be that

$$\left(\frac{s_1}{s_2 + 1}\right)^x \geq 1$$

and so

$$\frac{p_2 - p_3}{p_3 - p_1} < \left(\frac{s_1}{s_2 + 1}\right)^x$$

and therefore strategy J is always judged better than strategy T when $p_2 > p_3 > \bar{p} > p_1$.

So, if and only if either $p_1 > p_3 > \bar{p} > p_2$ and $\frac{p_3 - p_2}{p_1 - p_3} > \left(\frac{s_1+1}{s_2}\right)^x$ or $p_2 > p_3 > \bar{p} > p_1$, strategy J leads to a better next election state than strategy T. Conversely, it can be easily seen with similar logic that strategy T is superior when $p_1 > p_3 > \bar{p} > p_2$ and $\frac{p_3 - p_2}{p_1 - p_3} < \left(\frac{s_1+1}{s_2}\right)^x$. $\square$

4.4.4 Comparing strategies A and Z in the three-alternative case

A similar result can be obtained by focusing on elections with three alternatives and comparing strategies A and Z. We again narrow consideration to situations in which no two alternatives are tied in the election state and none of the focal voter's cardinal preferences equals their average or another cardinal preference.

Theorem 4.4.6. *In a three-alternative election with current election state* $\vec{s} = [s_1, s_2, s_3]$ *where* $s_1 > s_2 > s_3$ *and focal voter's cardinal preferences* $\vec{p} = [p_1, p_2, p_3]$ *where* $p_i \neq \frac{p_1+p_2+p_3}{3}$ *for all* i *and* $p_i \neq p_j$ *for* $i \neq j$*, employing strategy A will only lead the focal voter to a worse next election state than strategy Z as measured using the Merrill election-state metric with parameter* x *when*

- $p_1 > p_2 > \frac{p_1+p_3}{2} > p_3$ *and* $\frac{p_2-p_3}{p_1-p_2} > \left(\frac{s_1+1}{s_3}\right)^x$,
- $p_1 > p_3 > \frac{p_1+p_2}{2} > p_2$ *and* $\frac{p_3-p_2}{p_1-p_3} > \left(\frac{s_1+1}{s_2}\right)^x$,
- $p_2 > p_1 > \frac{p_2+p_3}{2} > p_3$ *and* $\frac{p_1-p_3}{p_2-p_1} > \left(\frac{s_2+1}{s_3}\right)^x$,
- $p_2 > \frac{p_1+p_2}{2} > p_3 > p_1$ *and* $\frac{p_2-p_3}{p_3-p_1} > \left(\frac{s_1}{s_2+1}\right)^x$,
- $p_3 > \frac{p_2+p_3}{2} > p_1 > p_2$ *and* $\frac{p_3-p_1}{p_1-p_2} > \left(\frac{s_2}{s_3+1}\right)^x$, *or*
- $p_3 > \frac{p_1+p_3}{2} > p_2 > p_1$ *and* $\frac{p_3-p_2}{p_2-p_1} > \left(\frac{s_1}{s_3+1}\right)^x$.

Proof. When no ties exist in $\vec{p}$, the ballots recommended by strategy Z are fully determined by the relative ordering of the entries in $\vec{p}$ and their average $\bar{p} = \frac{p_1+p_2+p_3}{3}$; strategy Z ignores the $\vec{s}$ vector. When $s_1 > s_2 > s_3$, strategies A and Z disagree in six of the twelve possible orderings of $\vec{p}$ and $\bar{p}$:

voter's preferences	strategy A recommends	strategy Z recommends
$p_1 > \bar{p} > p_2 > p_3$	$[1, 0, 0]$	$[1, 0, 0]$
$\boldsymbol{p_1 > p_2 > \bar{p} > p_3}$	$\mathbf{[1, 0, 0]}$	$\mathbf{[1, 1, 0]}$
$p_1 > \bar{p} > p_3 > p_2$	$[1, 0, 0]$	$[1, 0, 0]$
$\boldsymbol{p_1 > p_3 > \bar{p} > p_2}$	$\mathbf{[1, 0, 0]}$	$\mathbf{[1, 0, 1]}$
$p_2 > \bar{p} > p_1 > p_3$	$[0, 1, 0]$	$[0, 1, 0]$
$\boldsymbol{p_2 > p_1 > \bar{p} > p_3}$	$\mathbf{[0, 1, 0]}$	$\mathbf{[1, 1, 0]}$
$\boldsymbol{p_2 > \bar{p} > p_3 > p_1}$	$\mathbf{[0, 1, 1]}$	$\mathbf{[0, 1, 0]}$
$p_2 > p_3 > \bar{p} > p_1$	$[0, 1, 1]$	$[0, 1, 1]$
$\boldsymbol{p_3 > \bar{p} > p_1 > p_2}$	$\mathbf{[1, 0, 1]}$	$\mathbf{[0, 0, 1]}$
$p_3 > p_1 > \bar{p} > p_2$	$[1, 0, 1]$	$[1, 0, 1]$
$\boldsymbol{p_3 > \bar{p} > p_2 > p_1}$	$\mathbf{[0, 1, 1]}$	$\mathbf{[0, 0, 1]}$
$p_3 > p_2 > \bar{p} > p_1$	$[0, 1, 1]$	$[0, 1, 1]$

We compare A and Z according to the Merrill metric, as in section 4.4.1, considering the six cases separately.

Case 1: $p_1 > p_2 > \bar{p} > p_3$. Strategy A recommends the vote $[1, 0, 0]$ but strategy Z recommends $[1, 1, 0]$. To find when strategy A leads to a worse expectation of the election result than strategy Z, we set $V_{[1,0,0]} < V_{[1,1,0]}$:

$$\frac{p_1(s_1+1)^x + p_2 s_2^x + p_3 s_3^x}{(s_1+1)^x + s_2^x + s_3^x} < \frac{p_1(s_1+1)^x + p_2(s_2+1)^x + p_3 s_3^x}{(s_1+1)^x + (s_2+1)^x + s_3^x}$$

Following a derivation similar to the one in section 4.4.1, we find this inequality equivalent to

$$\frac{p_2 - p_3}{p_1 - p_2} > \left(\frac{s_1+1}{s_3}\right)^x$$

Case 2: $p_1 > p_3 > \bar{p} > p_2$. Strategy A recommends the vote $[1, 0, 0]$ but strategy Z recommends $[1, 0, 1]$. We found in section 4.4.2 that $[1, 0, 0]$ is worse for the voter than $[1, 0, 1]$ if and only if

$$\frac{p_3 - p_2}{p_1 - p_3} > \left(\frac{s_1+1}{s_2}\right)^x$$

Case 3: $p_2 > p_1 > \bar{p} > p_3$. Strategy A recommends the vote $[0,1,0]$ but strategy Z recommends $[1,1,0]$. To find when strategy A leads to a worse expectation of the election result than strategy Z, we set $V_{[0,1,0]} < V_{[1,1,0]}$:

$$\frac{p_1 s_1^x + p_2(s_2+1)^x + p_3 s_3^x}{s_1^x + (s_2+1)^x + s_3^x} < \frac{p_1(s_1+1)^x + p_2(s_2+1)^x + p_3 s_3^x}{(s_1+1)^x + (s_2+1)^x + s_3^x}$$

Following a now-familiar style of derivation, we find this inequality equivalent to

$$\frac{p_1 - p_3}{p_2 - p_1} > \left(\frac{s_2+1}{s_3}\right)^x$$

Case 4: $p_2 > \bar{p} > p_3 > p_1$. Strategy A recommends the vote $[0,1,1]$ but strategy Z recommends $[0,1,0]$. We found in section 4.4.1 that $[0,1,1]$ is worse for the voter than $[0,1,0]$ if and only if

$$\frac{p_2 - p_3}{p_3 - p_1} > \left(\frac{s_1}{s_2+1}\right)^x$$

Case 5: $p_3 > \bar{p} > p_1 > p_2$. Strategy A recommends the vote $[1,0,1]$ but strategy Z recommends $[0,0,1]$. To find when strategy A leads to a worse expectation of the election result than strategy Z, we set $V_{[1,0,1]} < V_{[0,0,1]}$:

$$\frac{p_1(s_1+1)^x + p_2 s_2^x + p_3(s_3+1)^x}{(s_1+1)^x + s_2^x + (s_3+1)^x} < \frac{p_1 s_1^x + p_2 s_2^x + p_3(s_3+1)^x}{s_1^x + s_2^x + (s_3+1)^x}$$

We find this inequality equivalent to

$$\frac{p_3 - p_1}{p_1 - p_2} > \left(\frac{s_2}{s_3+1}\right)^x$$

Case 6: $p_3 > \bar{p} > p_2 > p_1$. Strategy A recommends the vote $[0,1,1]$ but strategy Z recommends $[0,0,1]$. To find when strategy A leads to a worse expectation of the election result than strategy Z, we set $V_{[0,1,1]} < V_{[0,0,1]}$:

$$\frac{p_1 s_1^x + p_2(s_2+1)^x + p_3(s_3+1)^x}{s_1^x + (s_2+1)^x + (s_3+1)^x} < \frac{p_1 s_1^x + p_2 s_2^x + p_3(s_3+1)^x}{s_1^x + s_2^x + (s_3+1)^x}$$

This inequality is equivalent to

$$\frac{p_3 - p_2}{p_2 - p_1} > \left(\frac{s_1}{s_3 + 1}\right)^x$$

So strategy Z leads to a better next election state than strategy A in only these six cases. □

Note that, when $s_1 > s_2 + 1 > s_3 + 2$ and x is large enough, none of these six cases will hold, so strategy A dominates strategy Z as x approaches infinity.

4.4.5 Comparing strategies A and T in the four-alternative case

We have generalized the result in section 4.4.1 to more strategy pairs; here we evaluate strategy A against strategy T in the four-alternative case.

Theorem 4.4.7. *In a four-alternative election with current election state $\vec{s} = [s_1, s_2, s_3, s_4]$ where $s_1 > s_2 > s_3 > s_4$ and focal voter's cardinal preferences $\vec{p} = [p_1, p_2, p_3, p_4]$ where $p_i \neq p_j$ for $i \neq j$, employing strategy A will only lead the focal voter to a worse next election state than strategy T as measured using the Merrill election-state metric with parameter x when*

- $p_2 > p_3 > p_1 > p_4$ *and* $(p_3 - p_1)s_1^x + (p_3 - p_4)s_4^x < (p_2 - p_3)(s_2 + 1)^x$,
- $p_2 > p_4 > p_1 > p_3$ *and* $(p_4 - p_1)s_1^x + (p_4 - p_3)s_3^x < (p_2 - p_4)(s_2 + 1)^x$,
- $p_2 > p_3 > p_1$ *and* $p_2 > p_4 > p_1$ *and* $((p_3-p_1)s_1^x-(p_2-p_3)(s_2+1)^x)((s_3+1)^x-s_3^x)+((p_4-p_1)s_1^x-(p_2-p_4)(s_2+1)^x)((s_4+1)^x-s_4^x) < (p_3 - p_4)(s_3^x(s_4 + 1)^x - (s_3 + 1)^x s_4^x)$,
- $p_3 > p_2 > p_4 > p_1$ *and* $(p_4 - p_1)s_1^x < (p_2 - p_4)(s_2 + 1)^x + (p_3 - p_4)(s_3 + 1)^x$, *or*
- $p_4 > p_2 > p_3 > p_1$ *and* $(p_3 - p_1)s_1^x < (p_2 - p_3)(s_2 + 1)^x + (p_4 - p_3)(s_4 + 1)^x$.

Proof. When no ties exist in $\vec{s}$ or $\vec{p}$, the ballots recommended by strategies A and T are fully determined by the orderings of the alternatives in $\vec{s}$ and $\vec{p}$. When $s_1 > s_2 > s_3 > s_4$, strategies A and T disagree in six of the 24 possible orderings of $\vec{p}$:

voter's preferences	strategy A's recommended ballot	strategy T's recommended ballot
$p_1 > p_2 > p_3 > p_4$	$[1, 0, 0, 0]$	$[1, 0, 0, 0]$
$p_1 > p_2 > p_4 > p_3$	$[1, 0, 0, 0]$	$[1, 0, 0, 0]$
$p_1 > p_3 > p_2 > p_4$	$[1, 0, 0, 0]$	$[1, 0, 0, 0]$
$p_1 > p_3 > p_4 > p_2$	$[1, 0, 0, 0]$	$[1, 0, 0, 0]$
$p_1 > p_4 > p_2 > p_3$	$[1, 0, 0, 0]$	$[1, 0, 0, 0]$
$p_1 > p_4 > p_3 > p_2$	$[1, 0, 0, 0]$	$[1, 0, 0, 0]$
$p_2 > p_1 > p_3 > p_4$	$[0, 1, 0, 0]$	$[0, 1, 0, 0]$
$p_2 > p_1 > p_4 > p_3$	$[0, 1, 0, 0]$	$[0, 1, 0, 0]$
$\boldsymbol{p_2 > p_3 > p_1 > p_4}$	$[\mathbf{0, 1, 1, 0}]$	$[\mathbf{0, 1, 0, 0}]$
$\boldsymbol{p_2 > p_3 > p_4 > p_1}$	$[\mathbf{0, 1, 1, 1}]$	$[\mathbf{0, 1, 0, 0}]$
$\boldsymbol{p_2 > p_4 > p_1 > p_3}$	$[\mathbf{0, 1, 0, 1}]$	$[\mathbf{0, 1, 0, 0}]$
$\boldsymbol{p_2 > p_4 > p_3 > p_1}$	$[\mathbf{0, 1, 1, 1}]$	$[\mathbf{0, 1, 0, 0}]$
$p_3 > p_1 > p_2 > p_4$	$[1, 0, 1, 0]$	$[1, 0, 1, 0]$
$p_3 > p_1 > p_4 > p_2$	$[1, 0, 1, 0]$	$[1, 0, 1, 0]$
$p_3 > p_2 > p_1 > p_4$	$[0, 1, 1, 0]$	$[0, 1, 1, 0]$
$\boldsymbol{p_3 > p_2 > p_4 > p_1}$	$[\mathbf{0, 1, 1, 1}]$	$[\mathbf{0, 1, 1, 0}]$
$p_3 > p_4 > p_1 > p_2$	$[1, 0, 1, 1]$	$[1, 0, 1, 1]$
$p_3 > p_4 > p_2 > p_1$	$[0, 1, 1, 1]$	$[0, 1, 1, 1]$
$p_4 > p_1 > p_2 > p_3$	$[1, 0, 0, 1]$	$[1, 0, 0, 1]$
$p_4 > p_1 > p_3 > p_2$	$[1, 0, 0, 1]$	$[1, 0, 0, 1]$
$p_4 > p_2 > p_1 > p_3$	$[0, 1, 0, 1]$	$[0, 1, 0, 1]$
$\boldsymbol{p_4 > p_2 > p_3 > p_1}$	$[\mathbf{0, 1, 1, 1}]$	$[\mathbf{0, 1, 0, 1}]$
$p_4 > p_3 > p_1 > p_2$	$[1, 0, 1, 1]$	$[1, 0, 1, 1]$
$p_4 > p_3 > p_2 > p_1$	$[0, 1, 1, 1]$	$[0, 1, 1, 1]$

We compare A and T according to the Merrill metric, considering the six cases separately.

Case 1: $p_2 > p_3 > p_1 > p_4$. Strategy A recommends the vote $[0, 1, 1, 0]$ but strategy T recommends $[0, 1, 0, 0]$. To find when strategy A leads to a worse expectation of the election result than strategy

T, we set $V_{[0,1,1,0]} < V_{[0,1,0,0]}$:

$$\frac{p_1 s_1^x + p_2(s_2+1)^x + p_3(s_3+1)^x + p_4 s_4^x}{s_1^x + (s_2+1)^x + (s_3+1)^x + s_4^x} < \frac{p_1 s_1^x + p_2(s_2+1)^x + p_3 s_3^x + p_4 s_4^x}{s_1^x + (s_2+1)^x + s_3^x + s_4^x}$$

To simplify the derivation, we now define $A = p_1 s_1^x + p_2(s_2+1)^x + p_4 s_4^x$ and $B = s_1^x + (s_2+1)^x + s_4^x$, so the previous inequality becomes

$$\frac{A + p_3(s_3+1)^x}{B + (s_3+1)^x} < \frac{A + p_3 s_3^x}{B + s_3^x}$$

Since $s_i \geq 0$ for all i, the denominators are positive, and so

$$\left(A + p_3(s_3+1)^x\right)\left(B + s_3^x\right) < \left(A + p_3 s_3^x\right)\left(B + (s_3+1)^x\right)$$

Expanding factors,

$$AB + As_3^x + p_3(s_3+1)^x B + p_3(s_3+1)^x s_3^x < AB + A(s_3+1)^x + p_3 s_3^x B + p_3 s_3^x (s_3+1)^x$$

Canceling the ABs and substituting for A and B gives

$$(p_1 s_1^x + p_2(s_2+1)^x + p_4 s_4^x) s_3^x + p_3(s_3+1)^x (s_1^x + (s_2+1)^x + s_4^x) + p_3(s_3+1)^x s_3^x <$$
$$(p_1 s_1^x + p_2(s_2+1)^x + p_4 s_4^x)(s_3+1)^x + p_3 s_3^x (s_1^x + (s_2+1)^x + s_4^x) + p_3 s_3^x (s_3+1)^x$$

Expanding factors and then canceling terms once again,

$$p_1 s_1^x s_3^x + p_2(s_2+1)^x s_3^x + p_4 s_3^x s_4^x + p_3 s_1^x (s_3+1)^x + p_3(s_2+1)^x (s_3+1)^x + p_3(s_3+1)^x s_4^x <$$
$$p_1 s_1^x (s_3+1)^x + p_2(s_2+1)^x (s_3+1)^x + p_4(s_3+1)^x s_4^x + p_3 s_1^x s_3^x + p_3(s_2+1)^x s_3^x + p_3 s_3^x s_4^x$$

Grouping s_1^x terms and s_4^x terms on the left side and $(s_2+1)^x$ terms on the right side gives

$$p_3 s_1^x (s_3+1)^x - p_3 s_1^x s_3^x - p_1 s_1^x (s_3+1)^x + p_1 s_1^x s_3^x + p_3(s_3+1)^x s_4^x - p_3 s_3^x s_4^x - p_4(s_3+1)^x s_4^x +$$
$$p_4 s_3^x s_4^x < p_2(s_2+1)^x (s_3+1)^x - p_2(s_2+1)^x s_3^x - p_3(s_2+1)^x (s_3+1)^x + p_3(s_2+1)^x s_3^x$$

Factoring gives us

$$(p_3 - p_1)s_1^x((s_3+1)^x - s_3^x) + (p_3 - p_4)s_4^x((s_3+1)^x - s_3^x) < (p_2 - p_3)(s_2+1)^x((s_3+1)^x - s_3^x)$$

Since $x > 1$, it is always true that $(s_3+1)^x - s_3^x > 0$, and so

$$(p_3 - p_1)s_1^x + (p_3 - p_4)s_4^x < (p_2 - p_3)(s_2+1)^x$$

Case 2: $p_2 > p_3 > p_4 > p_1$. Strategy A recommends the vote $[0,1,1,1]$ but strategy T recommends $[0,1,0,0]$. To find when strategy A leads to a worse expectation of the election result than strategy T, we set $V_{[0,1,1,1]} < V_{[0,1,0,0]}$:

$$\frac{p_1 s_1^x + p_2(s_2+1)^x + p_3(s_3+1)^x + p_4(s_4+1)^x}{s_1^x + (s_2+1)^x + (s_3+1)^x + (s_4+1)^x} < \frac{p_1 s_1^x + p_2(s_2+1)^x + p_3 s_3^x + p_4 s_4^x}{s_1^x + (s_2+1)^x + s_3^x + s_4^x}$$

To simplify the derivation, we now define $A = p_1 s_1^x + p_2(s_2+1)^x$ and $B = s_1^x + (s_2+1)^x$, so the previous inequality becomes

$$\frac{A + p_3(s_3+1)^x + p_4(s_4+1)^x}{B + (s_3+1)^x + (s_4+1)^x} < \frac{A + p_3 s_3^x + p_4 s_4^x}{B + s_3^x + s_4^x}$$

Since $s_i \geq 0$ for all i, the denominators are positive, and so

$$\left(A + p_3(s_3+1)^x + p_4(s_4+1)^x\right)\left(B + s_3^x + s_4^x\right) < \left(A + p_3 s_3^x + p_4 s_4^x\right)\left(B + (s_3+1)^x + (s_4+1)^x\right)$$

Expanding factors,

$$AB + As_3^x + As_4^x + p_3(s_3+1)^x B + p_3(s_3+1)^x s_3^x + p_3(s_3+1)^x s_4^x + p_4(s_4+1)^x B + p_4(s_4+1)^x s_3^x + p_4(s_4+1)^x s_4^x < AB + A(s_3+1)^x + A(s_4+1)^x + p_3 s_3^x B + p_3 s_3^x(s_3+1)^x + p_3 s_3^x(s_4+1)^x + p_4 s_4^x B + p_4 s_4^x(s_3+1)^x + p_4 s_4^x(s_4+1)^x$$

Subtracting $AB + p_3 s_3^x(s_3+1)^x + p_4 s_4^x(s_4+1)^x$ from each side and substituting for A and B gives

$$(p_1s_1^x+p_2(s_2+1)^x)s_3^x+(p_1s_1^x+p_2(s_2+1)^x)s_4^x+p_3(s_3+1)^x(s_1^x+(s_2+1)^x)+p_3(s_3+1)^xs_4^x+p_4(s_4+1)^x(s_1^x+(s_2+1)^x)+p_4(s_4+1)^xs_3^x<(p_1s_1^x+p_2(s_2+1)^x)(s_3+1)^x+(p_1s_1^x+p_2(s_2+1)^x)(s_4+1)^x+p_3s_3^x(s_1^x+(s_2+1)^x)+p_3s_3^x(s_4+1)^x+p_4s_4^x(s_1^x+(s_2+1)^x)+p_4s_4^x(s_3+1)^x$$

Expanding factors,

$$p_1s_1^xs_3^x+p_1s_1^xs_4^x+p_2(s_2+1)^xs_3^x+p_2(s_2+1)^xs_4^x+p_3s_1^x(s_3+1)^x+p_3(s_2+1)^x(s_3+1)^x+p_3(s_3+1)^xs_4^x+p_4s_1^x(s_4+1)^x+p_4(s_2+1)^x(s_4+1)^x+p_4s_3^x(s_4+1)^x<p_1s_1^x(s_3+1)^x+p_1s_1^x(s_4+1)^x+p_2(s_2+1)^x(s_3+1)^x+p_2(s_2+1)^x(s_4+1)^x+p_3s_1^xs_3^x+p_3(s_2+1)^xs_3^x+p_3s_3^x(s_4+1)^x+p_4s_1^xs_4^x+p_4(s_2+1)^xs_4^x+p_4(s_3+1)^xs_4^x$$

Grouping s_1^x and $(s_2+1)^x$ terms on the left side and terms with no mention of s_1 or s_2 on the right side gives

$$p_3s_1^x(s_3+1)^x-p_1s_1^x(s_3+1)^x-p_3s_1^xs_3^x+p_1s_1^xs_3^x-p_2(s_2+1)^x(s_3+1)^x+p_3(s_2+1)^x(s_3+1)^x+p_2(s_2+1)^xs_3^x-p_3(s_2+1)^xs_3^x+p_4s_1^x(s_4+1)^x-p_1s_1^x(s_4+1)^x-p_4s_1^xs_4^x+p_1s_1^xs_4^x-p_2(s_2+1)^x(s_4+1)^x+p_4(s_2+1)^x(s_4+1)^x+p_2(s_2+1)^xs_4^x-p_4(s_2+1)^xs_4^x<p_3s_3^x(s_4+1)^x-p_3(s_3+1)^xs_4^x-p_4s_3^x(s_4+1)^x+p_4(s_3+1)^xs_4^x$$

Factoring gives us

$$((p_3-p_1)s_1^x-(p_2-p_3)(s_2+1)^x)((s_3+1)^x-s_3^x)+((p_4-p_1)s_1^x-(p_2-p_4)(s_2+1)^x)((s_4+1)^x-s_4^x)<(p_3-p_4)(s_3^x(s_4+1)^x-(s_3+1)^xs_4^x)$$

Case 3: $p_2>p_4>p_1>p_3$. Strategy A recommends the vote $[0,1,0,1]$ but strategy T recommends $[0,1,0,0]$. To find when strategy A leads to a worse expectation of the election result than strategy T, we set $V_{[0,1,0,1]}<V_{[0,1,0,0]}$:

$$\frac{p_1s_1^x+p_2(s_2+1)^x+p_3s_3^x+p_4(s_4+1)^x}{s_1^x+(s_2+1)^x+s_3^x+(s_4+1)^x}<\frac{p_1s_1^x+p_2(s_2+1)^x+p_3s_3^x+p_4s_4^x}{s_1^x+(s_2+1)^x+s_3^x+s_4^x}$$

The same derivation as in case 1, but with alternatives 3 and 4 switched, results in

$$(p_4-p_1)s_1^x+(p_4-p_3)s_3^x<(p_2-p_4)(s_2+1)^x$$

Case 4: $p_2 > p_4 > p_3 > p_1$. Strategy A recommends the vote $[0,1,1,1]$ but strategy T recommends $[0,1,0,0]$, just as in case 2, so the result found there:

$$((p_3 - p_1)s_1^x - (p_2 - p_3)(s_2 + 1)^x)((s_3 + 1)^x - s_3^x) + ((p_4 - p_1)s_1^x - (p_2 - p_4)(s_2 + 1)^x)((s_4 + 1)^x - s_4^x) < (p_3 - p_4)(s_3^x(s_4 + 1)^x - (s_3 + 1)^x s_4^x)$$

holds here. For convenience, call that inequality Ψ. Note that

$$(p_2 > p_3 > p_4 > p_1 \quad \text{and} \quad \Psi) \quad \text{or} \quad (p_2 > p_4 > p_3 > p_1 \quad \text{and} \quad \Psi)$$

is equivalent to

$$p_2 > p_3 > p_1 \quad \text{and} \quad p_2 > p_4 > p_1 \quad \text{and} \quad \Psi$$

Case 5: $p_3 > p_2 > p_4 > p_1$. Strategy A recommends the vote $[0,1,1,1]$ but strategy T recommends $[0,1,1,0]$. To find when strategy A leads to a worse expectation of the election result than strategy T, we set $V_{[0,1,1,1]} < V_{[0,1,1,0]}$:

$$\frac{p_1 s_1^x + p_2(s_2+1)^x + p_3(s_3+1)^x + p_4(s_4+1)^x}{s_1^x + (s_2+1)^x + (s_3+1)^x + (s_4+1)^x} < \frac{p_1 s_1^x + p_2(s_2+1)^x + p_3(s_3+1)^x + p_4 s_4^x}{s_1^x + (s_2+1)^x + (s_3+1)^x + s_4^x}$$

To simplify the derivation, we now define $A = p_1 s_1^x + p_2(s_2+1)^x + p_3(s_3+1)^x$ and $B = s_1^x + (s_2+1)^x + (s_3+1)^x$, so the previous inequality becomes

$$\frac{A + p_4(s_4+1)^x}{B + (s_4+1)^x} < \frac{A + p_4 s_4^x}{B + s_4^x}$$

Since $s_i \geq 0$ for all i, the denominators are positive, and so

$$\left(A + p_4(s_4+1)^x\right)\left(B + s_4^x\right) < \left(A + p_4 s_4^x\right)\left(B + (s_4+1)^x\right)$$

Expanding factors,

$$AB + As_4^x + p_4(s_4+1)^x B + p_4(s_4+1)^x s_4^x < AB + A(s_4+1)^x + p_4 s_4^x B + p_4 s_4^x (s_4+1)^x$$

Canceling the ABs and substituting for A and B gives

$$(p_1s_1^x+p_2(s_2+1)^x+p_3(s_3+1)^x)s_4^x+p_4(s_4+1)^x(s_1^x+(s_2+1)^x+(s_3+1)^x)+p_4(s_4+1)^x s_4^x <$$
$$(p_1s_1^x+p_2(s_2+1)^x+p_3(s_3+1)^x)(s_4+1)^x+p_4s_4^x(s_1^x+(s_2+1)^x+(s_3+1)^x)+p_4s_4^x(s_4+1)^x$$

Expanding factors and then canceling terms once again,

$$p_1s_1^xs_4^x+p_2(s_2+1)^xs_4^x+p_3(s_3+1)^xs_4^x+p_4s_1^x(s_4+1)^x+p_4(s_2+1)^x(s_4+1)^x+p_4(s_3+1)^x(s_4+1)^x < p_1s_1^x(s_4+1)^x+p_2(s_2+1)^x(s_4+1)^x+p_3(s_3+1)^x(s_4+1)^x+p_4s_1^xs_4^x+p_4(s_2+1)^xs_4^x+p_4(s_3+1)^xs_4^x$$

Grouping s_1^x terms on the left side and $(s_2+1)^x$ terms and $(s_3+1)^x$ terms on the right side gives

$$p_4s_1^x(s_4+1)^x-p_4s_1^xs_4^x-p_1s_1^x(s_4+1)^x+p_1s_1^xs_4^x <$$
$$p_2(s_2+1)^x(s_4+1)^x-p_2(s_2+1)^xs_4^x-p_4(s_2+1)^x(s_4+1)^x+p_4(s_2+1)^xs_4^x+p_3(s_3+1)^x(s_4+1)^x-p_3(s_3+1)^xs_4^x-p_4(s_3+1)^x(s_4+1)^x+p_4(s_3+1)^xs_4^x$$

Factoring gives us

$$(p_4-p_1)s_1^x((s_4+1)^x-s_4^x) < (p_2-p_4)(s_2+1)^x((s_4+1)^x-s_4^x)+(p_3-p_4)(s_3+1)^x((s_4+1)^x-s_4^x)$$

Since $x>1$, it is always true that $(s_4+1)^x-s_4^x>0$, and so

$$(p_4-p_1)s_1^x < (p_2-p_4)(s_2+1)^x+(p_3-p_4)(s_3+1)^x$$

Case 6: $p_4>p_2>p_3>p_1$. Strategy A recommends the vote $[0,1,1,1]$ but strategy T recommends $[0,1,0,1]$. To find when strategy A leads to a worse expectation of the election result than strategy T, we set $V_{[0,1,1,1]} < V_{[0,1,0,1]}$:

$$\frac{p_1s_1^x+p_2(s_2+1)^x+p_3(s_3+1)^x+p_4(s_4+1)^x}{s_1^x+(s_2+1)^x+(s_3+1)^x+(s_4+1)^x} < \frac{p_1s_1^x+p_2(s_2+1)^x+p_3s_3^x+p_4(s_4+1)^x}{s_1^x+(s_2+1)^x+s_3^x+(s_4+1)^x}$$

The same derivation as in case 5, but with alternatives 3 and 4 switched, results in

$$(p_3-p_1)s_1^x < (p_2-p_3)(s_2+1)^x+(p_4-p_3)(s_4+1)^x$$

$\square$

4.4.6 A general result for strategy A using the Merrill metric

Perhaps surprisingly, a stronger result regarding strategy A can be found even if the number of alternatives k is unrestricted. If we assume that the cardinal preferences are tie-free ($p_i \neq p_j$ when $i \neq j$) and that the election state is free of ties and near-ties
($s_1 > s_2 + 1 > s_3 + 2 > \ldots > s_k + k - 1$ without loss of generality), then strategy A always recommends approving alternatives i, where $p_i \geq p_1$ if $p_1 > p_2$ and $p_i > p_1$ if $p_1 < p_2$.

Theorem 4.4.8. *If the current election state $\vec{s}$ is tie- and near-tie-free, the focal voter's cardinal preferences $\vec{p}$ are tie-free and the Merrill metric is used to evaluate election states with the exponent x taken to approach infinity, then strategy A dominates any other strategy; i.e., it leads to a next election state judged to be superior to any other election state reachable by one approval ballot.*

Proof. We consider two cases, $p_1 > p_2$ and $p_2 > p_1$.

In the first case, where $p_1 > p_2$, the expected value of the current election state[5] according to the Merrill metric is

$$V_{[0,0,\ldots 0]} = \frac{\sum_{i=1}^{k} p_i s_i^x}{\sum_{i=1}^{k} s_i^x} = [p_1, p_2, \ldots p_k] \cdot \frac{[s_1^x, s_2^x, \ldots s_k^x]}{s_1^x + s_2^x + \cdots + s_k^x}$$

which is essentially nothing more than a weighted average of the p_i values. As $x \to \infty$, $\frac{s_i^x}{s_j^x} \to 0$ for all $i > j$. Therefore $\frac{s_1^x}{s_1^x + s_2^x + \cdots + s_k^x} \to 1$ and $\frac{s_i^x}{s_1^x + s_2^x + \cdots + s_k^x} \to 0$ for $i > 1$, so $V_{[0,0,\ldots 0]} \to p_1$.
Furthermore, $V_{[0,0,\ldots 0]} < p_1$ as $x \to \infty$, since (taking out the p_1 component by reducing s_1 to zero)

$$[p_2, p_3, \ldots p_k] \cdot \frac{[s_2^x, s_3^x, \ldots s_k^x]}{s_2^x + s_3^x + \cdots + s_k^x} \to p_2$$

and $p_2 < p_1$. So $V_{[0,0,\ldots 0]} \to p_1$ from below as $x \to \infty$, and, since no tie can be immediately broken or created, maximizing the expected value of the next election state is done by maximizing the "weights" s_i^x where $p_i \geq p_1$ and minimizing the others, which means approving only those alternatives i where $p_i \geq p_1$. p_1 is included since $V_{[0,0,\ldots 0]} < p_1$ for all large enough x.

[5] Note that the $V_{[0,0,\ldots 0]}$ notation simply refers to the expected value of the election state found by voting the ballot $[0, 0, \ldots 0]$, which is the same as the current election state.

The second case, where $p_1 < p_2$, is substantially similar to the first. Again, as $x \to \infty$, $\frac{s_1^x}{s_1^x+s_2^x+\cdots+s_k^x} \to 1$ and $\frac{s_i^x}{s_1^x+s_2^x+\cdots+s_k^x} \to 0$ for $i > 1$, so $V_{[0,0,\ldots 0]} \to p_1$. But this time $V_{[0,0,\ldots 0]} > p_1$ as $x \to \infty$, since

$$[p_2, p_3, \ldots p_k] \cdot \frac{[s_2^x, s_3^x, \ldots s_k^x]}{s_2^x + s_3^x + \cdots + s_k^x} \to p_2$$

and $p_2 > p_1$. So $V_{[0,0,\ldots 0]} \to p_1$ from *above* as $x \to \infty$, and, since no tie can be immediately broken or created, maximizing the expected value of the next election state is done by maximizing the weights s_i^x where $p_i > p_1$ and minimizing the others, which means approving only those alternatives i where $p_i > p_1$. p_1 is *not* included since $V_{[0,0,\ldots 0]} > p_1$ for all large enough x.

Therefore, to maximize the Merrill metric for the next election state, a voter's ballot should approve only alternatives i where $p_i \geq p_1$ if $p_1 > p_2$ and $p_i > p_1$ if $p_1 < p_2$. Strategy A does exactly this when the election state and voter's cardinal preferences are tie-free. $\square$

4.5 General results using the branching-probabilities election-state metric

The branching-probabilities metric described in section 4.3.2 estimates the expected value to a focal voter of an election state by using the set of already-voted ballots as a representative sample of the ballots to come. This metric thus makes the assumption that each of the ballots not yet seen will approve each alternative with probability equal to the proportion of already-voted ballots that approve that alternative, giving each alternative's final vote total a binomial distribution.

To evaluate various approval strategies according to the branching-probabilities metric, we must define precisely how to calculate a voter's expected value of an election state. It turns out that the definition

$$\binom{a}{b} = \begin{cases} \dfrac{a!}{(a-b)!b!} & \text{if} \quad 0 \leq b \leq a \\ 0 & \text{otherwise} \end{cases}$$

will simplify the derivations below.

If the election state $\vec{s}$ is based on y already-voted ballots, we can define $\vec{\pi} = \frac{\vec{s}}{y}$, the proportion of ballots that approves each alternative. Since the branching-probabilities metric assumes that each of the x ballots not yet seen will approve alternative i with probability π_i, the probability that alternative i finishes with exactly a votes is

$$\binom{x}{a - s_i} \pi_i^{a-s_i} (1 - \pi_i)^{x-a+s_i}$$

Similarly, the probability that alternative j finishes with exactly b votes is

$$\binom{x}{b - s_j} \pi_j^{b-s_j} (1 - \pi_j)^{x-b+s_j}$$

The probability that alternative j finishes with fewer than a votes is

$$\sum_{b=s_j}^{a-1} \binom{x}{b - s_j} \pi_j^{b-s_j} (1 - \pi_j)^{x-b+s_j}$$

The probability that *all* alternatives other than i finish with fewer than a votes is

$$\prod_{j \neq i} \left(\sum_{b=s_j}^{a-1} \binom{x}{b - s_j} \pi_j^{b-s_j} (1 - \pi_j)^{x-b+s_j} \right)$$

The probability that alternative i finishes with exactly a votes *and* all other alternatives finish with fewer is

$$\binom{x}{a - s_i} \pi_i^{a-s_i} (1 - \pi_i)^{x-a+s_i} \cdot \prod_{j \neq i} \left(\sum_{b=s_j}^{a-1} \binom{x}{b - s_j} \pi_j^{b-s_j} (1 - \pi_j)^{x-b+s_j} \right)$$

Finally, since the eventual winning vote total must be at least s_1 (the current total of the leading alternative) and at most $x + y$ (the total number of ballots), the probability that alternative i finishes ahead of all other alternatives is

$$W_i = \sum_{a=s_1}^{x+y} \left(\binom{x}{a - s_i} \pi_i^{a-s_i} (1 - \pi_i)^{x-a+s_i} \cdot \prod_{j \neq i} \left(\sum_{b=s_j}^{a-1} \binom{x}{b - s_j} \pi_j^{b-s_j} (1 - \pi_j)^{x-b+s_j} \right) \right)$$

In section 4.3.2, we allowed the possibility of tied winners and assumed that ties would be broken in favor of each tied winner with equal probability. If we ignore the possibility of tied winners, which are less and less likely as x gets large, the focal voter's estimated value of the final outcome from election state $\vec{s}$ is simply

$$\sum_{i=1}^{k} p_i W_i$$

Note that if x were small enough to make ties sufficiently likely, so that $\sum_{i=1}^{k} W_i$ might be substantially less than 1, then

$$\frac{\sum_{i=1}^{k} p_i W_i}{\sum_{i=1}^{k} W_i}$$

would be a better estimate. An exact expected value would have to take account of the possibility of each subset of the alternatives tying for the win. In the interest of relative simplicity, for the rest of this chapter we will assume that any final election state will have one clear winner.

As x goes to infinity, this binomial distribution approaches a normal distribution, with alternative i's mean final vote total at $s_i + x\pi_i$ (and thus mean approval proportion at π_i) and the variance approaching zero. For $i < j$ (and thus $s_i > s_j$), it always holds that $s_i + x\pi_i > s_j + x\pi_j$; as x gets large it becomes vanishingly unlikely that alternative i will finish behind alternative j. In particular, the probability that alternative 1, the leader in election state $\vec{s}$, finishes ahead of all the others approaches 1.

It will also prove valuable to notice that, if the rest of the vector $\vec{s}$ remains constant, increasing s_i (and thus π_i) necessarily increases W_i and decreasing s_i necessarily decreases W_i.

4.5.1 A general result for strategy A using the branching-probabilities metric

A similar result to that found in section 4.4.6 can be found using the branching-probabilities metric when x, the number of future ballots, is assumed to approach infinity.

The focal voter may choose any subset of the k alternatives for which to vote and so may reach 2^k possible election states. If the focal voter is the yth voter and we call the election state reached by his or her ballot $\vec{s}$, then W_i, defined above, is the probability that alternative i eventually wins.

Recall that, if we assume that the cardinal preferences are tie-free ($p_i \neq p_j$ when $i \neq j$) and that the election state is free of ties and near-ties ($s_1 > s_2 + 1 > s_3 + 2 > \ldots > s_k + k - 1$), then strategy A always recommends approving alternatives i, where $p_i \geq p_1$ if $p_1 > p_2$ and $p_i > p_1$ if $p_1 < p_2$.

Theorem 4.5.1. *If the current election state $\vec{s}$ is tie- and near-tie-free, the focal voter's cardinal preferences $\vec{p}$ are tie-free and the branching-probabilities metric is used to evaluate election states with x, the number of future ballots, taken to approach infinity, then strategy A dominates any other strategy; i.e., it leads to a next election state judged to be superior to any other election state reachable by one approval ballot.*

Proof. We consider two cases, $p_1 > p_2$ and $p_2 > p_1$.

In the first case, where $p_1 > p_2$, the expected value of the current election state according to the branching-probabilities metric is

$$V_{[0,0,\ldots 0]} = [p_1, p_2, \ldots p_k] \cdot [W_1, W_2, \ldots W_k] = \sum_{i=1}^{k} p_i W_i$$

where

$$W_i = \sum_{a=s_1}^{x+y} \left(\binom{x}{a-s_i} \pi_i^{a-s_i} (1-\pi_i)^{x-a+s_i} \cdot \prod_{j \neq i} \left(\sum_{b=s_j}^{a-1} \binom{x}{b-s_j} \pi_j^{b-s_j} (1-\pi_j)^{x-b+s_j} \right) \right)$$

which is essentially nothing more than a weighted average of the p_i values. As $x \to \infty$, $\frac{W_j}{W_i} \to 0$ for all $i > j$. In fact, $W_1 \to 1$ and $W_i \to 0$ for $i > 1$, so $V_{[0,0,\ldots 0]} \to p_1$. Furthermore, $V_{[0,0,\ldots 0]} < p_1$ as $x \to \infty$, since (taking out the p_1 component by reducing s_1 to zero[6])

$$[p_2, p_3, \ldots p_k] \cdot [W_2, W_3, \ldots W_k] \to p_2$$

and $p_2 < p_1$. So $V_{[0,0,\ldots 0]} \to p_1$ from below as $x \to \infty$, and, since no tie can be immediately broken or created, maximizing the expected value of the next election state is done by maximizing the winning probabilities ("weights") W_i where $p_i \geq p_1$ and minimizing the others—done by maximizing s_i where $p_i \geq p_1$ and minimizing s_i where $p_i < p_1$—which means approving only those alternatives i where $p_i \geq p_1$. p_1 is included since $V_{[0,0,\ldots 0]} < p_1$ for all large enough x.

[6]Note that reducing s_1 to zero also reduces π_1 and W_1 to zero and changes the rest of $\vec{\pi}$ and $\vec{W}$ so that they each still sum to one.

The second case, where $p_1 < p_2$, is substantially similar to the first. Again, as $x \to \infty$, $W_1 \to 1$ and $W_i \to 0$ for $i > 1$, so $V_{[0,0,\ldots 0]} \to p_1$. But this time $V_{[0,0,\ldots 0]} > p_1$ as $x \to \infty$, since

$$[p_2, p_3, \ldots p_k] \cdot [W_2, W_3, \ldots W_k] \to p_2$$

and $p_2 > p_1$. So $V_{[0,0,\ldots 0]} \to p_1$ from below as $x \to \infty$, and, since no tie can be immediately broken or created, maximizing the expected value of the next election state is done by maximizing the winning probabilities W_i where $p_i > p_1$ and minimizing the others—done by maximizing s_i where $p_i > p_1$ and minimizing s_i where $p_i \leq p_1$—which means approving only those alternatives i where $p_i > p_1$. p_1 is *not* included since $V_{[0,0,\ldots 0]} > p_1$ for all large enough x.

Therefore, to maximize the Merrill metric for the next election state, a voter's ballot should approve only alternatives i where $p_i \geq p_1$ if $p_1 > p_2$ and $p_i > p_1$ if $p_1 < p_2$. Strategy A does exactly this when the election state and voter's cardinal preferences are tie-free. □

4.6 Summary of contributions

In this research, we have presented approaches to evaluating arbitrary strategies for approval voting in a DSV context. Specifically, we have accomplished the following.

1. Presented two compelling metrics for use in evaluating the effectiveness of approval strategies that respond to poll data (an election state): one based on Merrill [38] and one based on branching approval probabilities.

2. Used the Merrill metric to compare strategy pairs A *vs.* T, A *vs.* J, T *vs.* J and A *vs.* Z for three-alternative elections.

3. Used the Merrill metric to compare strategy pairs A *vs.* T and A *vs.* J for four-alternative elections.

4. Proved that, in the tie- and near-tie-free case, strategy A dominates all other approval strategies using the Merrill metric as the exponent approaches infinity.

5. Proved that, in the tie- and near-tie-free case, strategy A dominates all other approval strategies using the branching-probabilities metric as the number of future ballots approaches infinity.

Chapter 5

Fixed-size Minimax

This chapter represents joint work with Vangelis Markakis and Aranyak Mehta.

5.1 Introduction

Voting has been a very popular method for preference aggregation in multi-agent environments. It is often the case that a set of agents with different preferences need to make a choice among a set of alternatives, where the alternatives could be various entities such as potential committee members, or joint plans of action. A standard methodology for this scenario is to have each agent express his/her/its preferences and then select an alternative (or more than one alternative in multiwinner elections) according to some voting protocol. Several decision-making applications in artificial intelligence have followed this approach including problems in collaborative filtering [46] and planning [26, 27].

In this work we focus on solution concepts for approval voting, which is a voting scheme for committee elections (multiwinner elections). In such a protocol, the voters are allowed to vote for, or approve of, as many alternatives as they like. In the last three decades, many scientific societies and organizations have adopted approval voting, including, among others, the American

Mathematical Society (AMS), the Institute of Electrical and Electronics Engineers (IEEE), the Game Theory Society (GTS) and the European Association for Logic, Language and Information.

A ballot in an approval voting protocol can be seen as a binary vector that indicates the alternatives approved of by the voter. Given the ballots, the obvious question is: what should the outcome of the election be? The solution concept that has been used in almost all such elections is the minisum solution, *i.e.*, output the committee which, when seen as a 0/1-vector, minimizes the sum of the Hamming distances to the ballots. If there is no restriction on the size of the elected committee this is equivalent to a majority vote on each alternative. If there is a restriction, *e.g.*, if the elected committee should be of size exactly k, then the minisum solution consists of the k alternatives with the highest number of approvals [16].

Recently, a new solution concept, the minimax solution, was proposed by Brams, Kilgour and Sanver [15]. The minimax solution chooses a committee which, when seen as a 0/1-vector, minimizes the *maximum* Hamming distance to all ballots. When there is a restriction that the size of the committee should be exactly k, then the minimax solution picks, among all committees of size k, the one that minimizes the maximum Hamming distance to the ballots.

The main motivation behind the minimax solution is to address the issues of fairness and compromise. Since minimax minimizes the disagreement with the least satisfied voter, it tends to result in outcomes that are more widely acceptable than the minisum solution. Also, majority tyranny is avoided: a majority of voters cannot guarantee a specific outcome, unlike under minisum. On the other hand, advantages of the minisum approach include simplicity, ease of computation and nonmanipulability. Further discussions on the properties and the pros and cons of the minisum and the minimax solutions are provided by Brams, Kilgour and Sanver [15, 16].

In this work we address computational aspects of the minimax solution, with a focus on elections for committees of fixed size. In contrast to the minisum solution, which is easy to compute in polynomial time, we show that finding a minimax solution is NP-hard. We therefore resort to polynomial-time heuristics and approximation algorithms.

We first exhibit a simple algorithm that achieves an approximation factor of 3. We then propose a variety of local search heuristics, some of which use the solution of our approximation algorithm as an initial point. All our heuristics run relatively fast and we evaluated the quality of their output both on randomly generated data as well as on the 2003 Game Theory Society election. Our simulations show that the heuristics perform very well, finding a solution very close to optimal on average. In fact for some heuristics the average error in the approximation can be as low as 0.05%.

Finally, in section 5.5, we focus on the question of manipulating the minimax solution. We show that any algorithm that computes an optimal minimax solution is manipulable. However, the same may not be true for approximation algorithms. As an example, we show that our 3-approximation algorithm is nonmanipulable.

5.1.1 Related work

The minimax solution concept that we study here was introduced by Brams, Kilgour and Sanver [15]. In subsequent work by the same authors [16, 32, 12], a weighted version of the minimax solution is studied, which takes into account the number of voters who voted for each distinct ballot and the proximity of each ballot to the other voters' ballots. The algorithms that are proposed by those authors [15, 16, 32, 12] all run in worst-case exponential time, and this is not surprising since the problem is NP-hard, as we exhibit in section 5.3. Approximation algorithms have previously been established only for the version in which there is no restriction on the size of the committee (which includes as a possibility that no alternative is elected). This variant is referred to as the endogenous minimax solution and it also arises in coding theory under the name of the Minimum Radius Problem or the Hamming Center Problem and in computational biology, where it is known as the Closest String Problem. In the context of computational biology, it was shown by Li, Ma and Wang [37] that the endogenous version admits a Polynomial-Time Approximation Scheme (PTAS), *i.e.*, a $(1 + \epsilon)$-approximation for any constant ϵ (with the running time depending exponentially in $1/\epsilon$). Other constant-factor approximations for the endogenous version had been obtained before [29, 34]. We are not aware of any polynomial-time approximation algorithms or any heuristic approaches for the non-endogenous versions, *i.e.*, in the presence of any upper or lower bounds on the size of the committee. Complexity considerations for winner determination in multiwinner elections have also been addressed recently [49] but not for the minimax solution.

5.2 Definitions and notation

We now formally define our problem. We have an election with m ballots and n alternatives. Each ballot is a binary vector $v \in \{0,1\}^n$, with the meaning that the ith coordinate of v is 1 if the voter approves of alternative i. For two binary vectors v_i, v_j of the same length, let $H(v_i, v_j)$ denote their Hamming distance, which is the number of coordinates in which they differ. For a vector $v \in \{0,1\}^n$, we will denote by $\mathrm{wt}(v)$ the number of coordinates that are set to 1 in v. The maxscore of a binary vector is defined as the Hamming distance between it and the ballot farthest from it: $\mathrm{maxscore}(v) \equiv \max_i H(v, v_i)$ where v_i is the ith ballot. We first define the problem in its generality.

> **Bounded-size Minimax (BSM(k_1, k_2))**
>
> INSTANCE: m ballots $v_1, \ldots, v_m \in \{0,1\}^n$ and integers k_1 and k_2 where $0 \leq k_1, k_2 \leq n$
>
> PROBLEM: Find a vector v^* such that $k_1 \leq \mathrm{wt}(v^*) \leq k_2$ so as to minimize $\mathrm{maxscore}(v^*)$.

BSM includes as a special case the endogenous version, BSM$(0, n)$, *i.e.*, no restrictions on the size of the committee. Also, since in some committee elections, the size of the committee to be elected is fixed (*e.g.*, the Game Theory Society elections), we are interested in the variant of BSM with $k_1 = k_2 = k$, which we call Fixed-size Minimax (FSM(k)).

> **Fixed-size Minimax (FSM(k))**
>
> INSTANCE: m ballots $v_1, \ldots, v_m \in \{0,1\}^n$ and integer k where $0 \leq k \leq n$
>
> PROBLEM: Find a vector v^* such that $\mathrm{wt}(v^*) = k$ so as to minimize $\mathrm{maxscore}(v^*)$.

In this research, we focus on elections with committees of fixed size and report our findings for FSM. We briefly mention in the relevant sections throughout the chapter as well as in section 5.6 which of our results extend to the general BSM problem.

As we show in the next section, BSM and FSM are NP-hard. Therefore, a natural approach is to focus on polynomial-time approximation algorithms. We use the standard notion of approximation algorithms: An algorithm for a minimization problem achieves an approximation ratio (or factor) of α ($\alpha \geq 1$), if for every instance of the problem the algorithm outputs a solution with cost at most α times the cost of an optimal solution.

5.3 NP-hardness and approximation algorithms

We first show that it is unlikely to have a polynomial-time algorithm for the minimax solution. In fact for the endogenous version of BSM, BSM$(0, n)$, NP-hardness has already been established by Frances and Litman [28], where the problem is stated in the context of coding theory. It follows that BSM in general is NP-hard. We show now that FSM is also NP-hard.

Theorem 5.3.1. *FSM is NP-hard.*

Proof. Suppose we had a polynomial-time algorithm for FSM. Then we could run such an algorithm first with $k = 0$, then with $k = 1$ and so on up to $k = n$ and output the best solution. That would give an optimal solution for BSM$(0, n)$. Hence FSM is also NP-hard. An alternative proof for the NP-hardness of FSM (and consequently of BSM as well) via a reduction from Vertex Cover was also obtained by LeGrand [36]. □

FSM(k) can be solved in polynomial time if k is an absolute constant, since then we can just go through all the $\binom{n}{k}$ different committees and output the best one. Also, if m is an absolute constant then we can express the problem as an integer program with a constant number of constraints, which by a result of Papadimitriou [45] can be solved in polynomial time.

The standard approach in dealing with NP-hard problems is to search for approximation algorithms. We will say that an algorithm for a minimization problem achieves an approximation ratio of α if for every instance of the problem the algorithm outputs a solution with cost at most α times the cost of an optimal solution. We will show that a very simple and fast algorithm achieves an approximation ratio of 3 for FSM(k), for every k. In fact, we will see that the algorithm has a factor of 3 for approval voting problems with much more general constraints.

But before stating the algorithm we need to introduce some more notation. Given a vector v, we will say that u is a *k-completion* of v, if $\mathrm{wt}(u) = k$, and $H(u, v)$ is the minimum possible Hamming distance between v and any vector of weight k. It is very easy to obtain a k-completion for any vector v: if $\mathrm{wt}(v) < k$, then pick any $k - \mathrm{wt}(v)$ coordinates in v that are 0 and set them to 1; if $\mathrm{wt}(v) > k$ then pick any $\mathrm{wt}(v) - k$ coordinates that are set to 1 and set them to 0.

The algorithm is now very simple to state: Pick arbitrarily one of the m ballots, say v_j. Output a k-completion of v_j, say u. Obviously this algorithm runs in time $O(n)$, independent of the number of voters.

Theorem 5.3.2. *The above algorithm achieves an approximation ratio of* 3.

Proof. Let v^* be an optimal solution ($\mathrm{wt}(v^*) = k$) and let $\mathrm{OPT} = \mathrm{maxscore}(v^*) = \max_i H(v^*, v_i)$ be the maximum distance of a ballot from the optimal solution. Let v_j be the ballot picked by the algorithm and let u be the k-completion of v_j that is output by the algorithm. We need to show that for every i, $H(u, v_i) \leq 3\,\mathrm{OPT}$. By the triangle inequality, we know that for every $1 \leq i \leq m$, $H(u, v_i) \leq H(u, v_j) + H(v_j, v_i)$. By applying the triangle inequality again we have:

$$H(u, v_i) \leq H(u, v_j) + H(v_j, v^*) + H(v^*, v_i)$$

Since v^* is an optimal solution, we have that $H(v^*, v_i) \leq \mathrm{OPT}$ and $H(v^*, v_j) \leq \mathrm{OPT}$. Also since u is a k-completion of v_j, by definition $H(u, v_j) \leq H(v^*, v_j) \leq \mathrm{OPT}$. Hence in total we obtain that $H(u, v_i) \leq 3\,\mathrm{OPT}$, as desired. □

We can also show that if at least one voter has weight k, then the algorithm achieves a ratio of 2. The algorithm can be easily adapted to give a ratio of 3 for the BSM version too; we only need to modify the notion of a k-completion accordingly. In fact, for $\mathrm{BSM}(0, n)$, the ratio is 2.

Note also that the analysis shows that there can be many different solutions that constitute a 3-approximation, since a ballot can potentially have many different k-completions.

Remark 5.3.3. **Generalized Constraints:** One may define an approval voting problem with constraints that are more general than simply those on the size of the committee (as in BSM). For example, one may have constraints on the number of members elected from a particular subgroup of alternatives (quotas), or constraints which require exactly one out of two particular alternatives to be in the committee (XOR constraints). Suppose, for any vote vector v, we can compute in polynomial time a *feasible-completion* of v, which is a committee that satisfies the constraints, and is closest to v in Hamming distance. Then, we can extend our algorithm to this setting in a natural manner, and prove that it provides a factor 3 approximation.

We are not aware of any better approximation algorithm for FSM. The endogenous version, BSM$(0, n)$, admits a PTAS, *i.e.*, for every constant ϵ, there exists a $(1 + \epsilon)$-approximation, which is polynomial in n and m and exponential in $1/\epsilon$. The PTAS was obtained in [37], in the context of computational biology. Before that, constant-factor approximations for BSM$(0, n)$ had been obtained in [29] and [34]. We believe that algorithms with such better factors may also be obtainable for FSM(k).

5.4 Local search heuristics for fixed-size minimax

Although the algorithm of section 5.3 gives a theoretical worst-case guarantee (we may even have a better performance in practice), a factor 3-algorithm may still be far away from acceptably good outcomes. Thus we focus on polynomial-time heuristics, which turn out to perform well in practice, if not optimally, even though we cannot obtain an improved worst-case guarantee. The heuristics that we investigate are based on local search; some of them use the 3-approximation as a starting point and retain its ratio.

5.4.1 A framework for FSM heuristics

Our overall heuristic approach is as follows. We start from a binary vector (picked according to some rule) and then we investigate if neighboring solutions to the current one improve the current maxscore. The local moves that we allow are removing some alternatives from the current committee and adding the same number of alternatives in, from the set of alternatives who do not belong to the current committee. We keep making local moves until no improvement in maxscore is seen for n consecutive moves.

1. Start with some $c \in \{0,1\}^n$.

2. Repeat until maxscore(c) does not change for n loop iterations:

   (a) Let A be the set of all binary vectors reachable from c by flipping up to p number of 0-bits of c to 1 and p 1-bits to 0, where p is an integer constant. (Note that c will necessarily be a member of A.)

   (b) Let $A^\star$ be the set that includes all members of A with smallest maxscore.

   (c) Choose at random one member of $A^\star$ and make it the new c.

3. Take c as the solution.

It is obviously important that the heuristic find a solution in time polynomial in the size of the input. In the worst case, the loop in the heuristic could run for n iterations for each step down in maxscore, so even if the maxscore of the initial c is the largest possible, n, no more than $O(n^2)$ iterations of the loop will be made. Each loop iteration runs in $O(mn^{2p+1})$ time, since the number of swaps to be considered is $O(n^{2p})$ and calculating the maxscore of each takes $O(mn)$ time, so the worst-case running time for the heuristic is $O(mn^{2p+3})$, which is of course polynomial in m and n as long as p is constant.

This heuristic framework has two parameters: the starting point for the binary vector c and the constant number p of alternatives to replace in one local move. While many combinations are possible, we will investigate using four different approaches to determining the c starting point and two values of p—1 and 2—resulting in eight specific heuristics. The four c starting points we use are

1. A fixed-size-minisum solution: the set of the k alternatives most approved on the ballots.

2. The FSM 3-approximation presented above: a k-completion of a ballot.

3. A random set of k alternatives.

4. A k-completion of a ballot with highest maxscore.

For approach 2, the ballot and k-completion are not chosen randomly: Of the ballots with Hamming weight nearest to k, the v^* minimizing sumscore$(v) \equiv \sum_i H(v^*, v_i)$ is chosen, and bits flipped are chosen to minimize resulting sumscore. The endogenous minimax equivalent of each of these approaches was investigated by LeGrand [36].

We will use the notation $h_{i,j}$ to refer to the heuristic with starting point i and $p = j$. For example, $h_{3,1}$ is the heuristic that starts with a random set of k alternatives and swaps at most one 0-bit with one 1-bit at a time.

5.4.2 Evaluating the heuristics

We show that the heuristics find good, if not optimal, winner sets on average. The approach is as follows. Given n, m and k, some large number of simulated elections are run. For each election, m ballots of n alternatives are generated according to some distribution. The maxscores of the optimal minimax set and the winner sets found using each of the heuristics and our 3-approximation (with ballot and flipped bits chosen at random) are then calculated.

We use two ballot-generating distributions: "unbiased" and "biased". The unbiased distribution simply sets each bit on each ballot to 0 or 1 with equal probability, like flipping an unbiased coin. The biased distribution generates for each alternative two approval probabilities, π_1 and π_2, between 0 and 1 with uniform randomness. The ballots are then divided into three groups. 40% of the ballots are generated according to the π_1 values; that is, each ballot approves each alternative with probability equal to its π_1 value. Another 40% of the ballots are generated according to the π_2 values, and the remaining 20% are generated as in the unbiased distribution.

We ran 5000 simulated elections in each of seven different configurations, varying n, m, k and the ballot-generating distribution. We also ran the heuristics 5000 times each on the ballots from the 2003 Game Theory Society council election.

Table 5.1 gives the highest realized approximation ratio (maxscore found divided by optimal maxscore) found over all 5000 elections for each heuristic, our 3-approximation (with ballot and flipped bits chosen at random), the minisum set (for comparison), and a maximax set. A maximax set is a set of size k that has the highest possible maxscore; it can be found by choosing a ballot

Table 5.1: Largest approximation ratios found for local search heuristics

n	20	20	20	24	20	20	24
k	10	10	10	12	10	10	12
m	50	200	800	50	50	200	161
ballots	unbiased	unbiased	unbiased	unbiased	biased	biased	GTS 2003
minimax set	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
$h_{1,1}$	1.1818	1.0769	1.0714	1.1538	1.2000	1.0909	1.0714
$h_{2,1}$	1.1818	1.0769	1.0714	1.1538	1.2000	1.1818	1.0714
$h_{3,1}$	1.1818	1.0769	1.0714	1.1538	1.2000	1.1818	1.0714
$h_{4,1}$	1.1818	1.0769	1.0714	1.1538	1.2000	1.1818	1.0714
$h_{1,2}$	1.0909	1.0769	1.0714	1.0769	1.1000	1.0833	1.0000
$h_{2,2}$	1.0909	1.0769	1.0714	1.0769	1.1000	1.0833	1.0000
$h_{3,2}$	1.0909	1.0769	1.0714	1.0769	1.1000	1.0833	1.0000
$h_{4,2}$	1.0909	1.0769	1.0714	1.0769	1.1000	1.0833	1.0000
3-approx.	1.6667	1.4615	1.3571	1.6154	1.8182	1.5833	1.3571
minisum set	1.5455	1.4615	1.3571	1.6923	1.6364	1.5833	1.2143
maximax set	1.8182	1.5385	1.4286	1.8462	2.2222	1.8182	1.7143

with Hamming weight nearest to $n - k$, flipping all of its bits and then performing a k-completion on it.

It can be seen that our 3-approximation in practice performs appreciably better than its guarantee—its ratio was less than 2 for every simulated election. (We were able to find instances of ratio-3 performance for smaller values of n, *e.g.*, 6.) As Table 5.1 shows, the heuristics reliably find solutions with ratios well below 2, but the average ratios found, given in Table 5.2, show that the average performance of the heuristics is more impressive still.

Finally, we compared the maxscores found by the heuristics with the worst possible maxscore of a winner set, and scaled them so that the maxscore of the exact minimax set becomes 100% and that of a maximax set becomes 0%, giving a more intuitive performance metric for heuristics. For example, if the minimax set has a maxscore of 12, a maximax set has a maxscore of 20 and a heuristic finds a solution with maxscore 13, the heuristic's scaled performance for that election will be $(20 - 13)/(20 - 12) = 87.5\%$. The averages of these scaled performances can be found in Table 5.3.

We draw the following conclusions from our experiments.

Table 5.2: Average approximation ratios found for local search heuristics

n	20	20	20	24	20	20	24
k	10	10	10	12	10	10	12
m	50	200	800	50	50	200	161
ballots	unbiased	unbiased	unbiased	unbiased	biased	biased	GTS 2003
minimax set	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
$h_{1,1}$	1.0058	1.0320	1.0007	1.0093	1.0083	1.0210	1.0012
$h_{2,1}$	1.0118	1.0365	1.0007	1.0147	1.0112	1.0251	1.0017
$h_{3,1}$	1.0122	1.0370	1.0007	1.0151	1.0122	1.0262	1.0057
$h_{4,1}$	1.0117	1.0364	1.0007	1.0149	1.0116	1.0262	1.0059
$h_{1,2}$	1.0004	1.0129	1.0005	1.0011	1.0004	1.0025	1.0000
$h_{2,2}$	1.0004	1.0164	1.0005	1.0014	1.0005	1.0029	1.0000
$h_{3,2}$	1.0004	1.0164	1.0005	1.0018	1.0005	1.0031	1.0000
$h_{4,2}$	1.0003	1.0167	1.0005	1.0014	1.0006	1.0029	1.0000
3-approx.	1.2477	1.1871	1.1204	1.2567	1.3121	1.2424	1.3571
minisum set	1.1650	1.1521	1.1060	1.1665	1.2119	1.1932	1.2143
maximax set	1.6746	1.4895	1.3343	1.7320	1.8509	1.6302	1.7143

- The heuristics perform well. Given the ballot distributions we used, very rarely would a heuristic find a solution that is unacceptably poorer than the optimal minimax solution. In particular, $h_{2,1}$ and $h_{2,2}$ vastly outperform the plain 3-approximation (while retaining its ratio-3 guarantee) with only a modest increase in running time.

- The heuristics perform significantly better on average when $p = 2$ than when $p = 1$. Increasing p further can be expected to improve performance further, at the expense of increased running time.

- Comparing the performance of the heuristics with equal p, all four perform similarly overall, but the best c-starting-point approach on average seems to be the first (a fixed-size-minisum solution); it significantly outperforms the other three sometimes (*e.g.*, when $p = 1$ in the unbiased-coin cases with 50 ballots) and is never outperformed by them with any statistical significance.

5.5 Manipulation

Recall that the Gibbard–Satterthwaite theorem [30, 53] says that any election system that chooses exactly one winner from a slate of more than two alternatives and satisfies a few obviously

Table 5.3: Average scaled performance of local search heuristics

n	20	20	24	20	20	24
k	10	10	12	10	10	12
m	50	200	50	50	200	161
ballots	unbiased	unbiased	unbiased	biased	biased	GTS '03
$h_{1,1}$	99.18%	94.05%	98.83%	99.07%	96.86%	99.82%
$h_{2,1}$	98.33%	93.23%	98.11%	98.74%	96.24%	99.77%
$h_{3,1}$	98.27%	93.13%	98.06%	98.62%	96.06%	99.20%
$h_{4,1}$	98.33%	93.24%	98.08%	98.68%	96.08%	99.18%
$h_{1,2}$	99.95%	97.60%	99.87%	99.95%	99.63%	100.00%
$h_{2,2}$	99.95%	96.96%	99.83%	99.94%	99.57%	100.00%
$h_{3,2}$	99.95%	96.95%	99.79%	99.94%	99.54%	100.00%
$h_{4,2}$	99.96%	96.89%	99.83%	99.94%	99.57%	100.00%
3-approx.	63.36%	62.31%	65.04%	63.36%	61.73%	50.00%
minisum	75.57%	69.40%	77.29%	75.04%	69.49%	70.00%

desirable assumptions (such as an absence of bias for some alternatives over others) is sometimes manipulable by insincere voters. In other words, there exist situations under any reasonable single-winner system in which some voters can gain better outcomes for themselves by voting insincerely.

Happily, the Gibbard–Satterthwaite theorem does not apply to the minimax and minisum solutions since they are free to choose winner sets of any size. In fact, the minisum procedure is completely nonmanipulable when any set of winners is allowed, as shown by Brams *et al.* [16]. This is true because a minisum election with n alternatives is exactly equivalent to n elections of two "alternatives" each: approve or disapprove that alternative. Since a voter's decision to approve or disapprove one alternative has absolutely no effect on whether other alternatives are chosen as winners, there is no more effective strategy than voting sincerely. Consequently, it is reasonable to expect a set of minisum ballots to have been sincerely voted.

Unfortunately, in addition to being possibly hard to compute exactly, the minimax solution is easily shown to be manipulable for the FSM version.

Definition 5.5.1. Fix an approval voting algorithm A and a set of ballots $\mathbf{v} = (v_1, v_2, \ldots v_m)$. Fix a voter i, and let $\mathbf{v^{-i}}$ denote the ballots of the rest of the voters. The loss $L_A^i(\mathbf{v})$ of voter i is defined as $H(v_i, A(\mathbf{v}))$. Algorithm A is said to be manipulable if there exist ballots $\mathbf{v}$, a voter i, and a ballot $v' \neq v_i$, such that $L_A^i(v_i, \mathbf{v^{-i}}) > L_A^i(v', \mathbf{v^{-i}})$.

We prove FSM's manipulability by giving an example of a voter gaining a superior outcome by voting insincerely (an analogous example for the endogenous version was provided by LeGrand [36]):

Theorem 5.5.2. *Any algorithm that computes an optimal solution for FSM is manipulable.*

Proof. Consider the following set of *sincere* ballots:

00110, **00011**, **00111**, **00001**, **10111**, **01111**

The minimax winner sets of size 2 are **00011** and **00101** with a maxscore of 2. The first voter, however, could manipulate the result by voting the insincere ballot **11110**. In that case, it can be checked that the optimal solution of size 2 is **00110**, which is exactly the most preferred outcome of the first voter. □

Such examples illustrate a general guideline to manipulating a minimax election: If there are alternatives of which the majority disapproves, a voter may be able to vote safely in favor of those alternatives to force more agreement with his relatively controversial choices. Put another way, if the minimax set can be seen as a kind of average of all ballots, a voter can move his ballot farther away from the current consensus to drag it closer to his ideal outcome. The minimax solution is extremely sensitive to "outliers" compared to the minisum solution, in much the same way that the average of a sample of data is more sensitive to outliers than the median.

If all voters use the above strategy, each alternative will tend toward having about as many approvals as disapprovals, making the result extremely unstable. Even electorates with widespread agreement will see their disagreements dramatically magnified by insincere strategy.

Although algorithms that always compute an optimal minimax solution are manipulable, the same may not be true if we allow approximation algorithms. The following theorem shows that we can have nonmanipulable algorithms if we are willing to settle for approximate solutions.

Theorem 5.5.3. *The voting procedure that results from using the* 3*-approximation algorithm described in section 5.3 is nonmanipulable by insincere voters.*

Proof. The algorithm picks a ballot v_j at random and outputs a k-completion of v_j. For a voter i, if the algorithm did not pick v_i, then the voter cannot change the output of the algorithm by lying. Furthermore, if the algorithm did pick v_i, then the best outcomes of size k for v_i are precisely all the k-completions of v_i. Therefore, by lying, the voter cannot possibly alter the outcome to his benefit. □

We conjecture that the heuristics of section 5.4 are also hard to manipulate. Although we do not have a proof for this, our intuition is the following. The heuristics use a lot of randomization—in all of them, either the starting point or the local move is based on a random choice. It therefore seems unlikely for a voter to be able to change his vote in such a way that the random choices of the algorithms will (even in expectation) work towards his benefit.

The above theorems give rise to the following question: What is the smallest value of α for which there exists a nonmanipulable polynomial-time approximation algorithm with ratio α?

Another interesting question is whether there exist algorithms (either exact or approximate) which are NP-hard to manipulate (i.e., although they are manipulable, the voter would have to solve an NP-hard problem in order to cheat). See Bartholdi *et al.* [7, 9] as well as more recent work [19, 20, 21, 25] along this line of research. In another recent work [48], average-case complexity is introduced as a complexity measure for manipulation instead of worst-case complexity (NP-hardness).

5.6 Future work

We have initiated a study of the computational issues involved in committee elections. Our results, along with the analysis of the endogenous version by LeGrand [36], show that local search heuristics perform very well in approximating the minimax solution in polynomial time.

There are still many interesting directions for future research. First, we are planning to adjust our heuristics for the weighted version of the minimax solution [16]. This version takes into account both the number of voters that vote each distinct ballot and the proximity of each ballot to the other voters' ballots. We are also investigating variations of local search that may improve the performance even more, *e.g.*, can there be a better starting point or can we enrich the set of local moves? Another interesting topic would be to compare local search with other heuristic approaches that could be adapted for our problem, like simulated annealing or genetic algorithms, or to reduce a minimax problem to a SAT (SATISFIABILITY) problem and use a SAT solver [39].

In terms of theoretical results, the most compelling question is to determine the best approximation ratio that can be achieved in polynomial time for the minimax solution. The questions stated in section 5.5 regarding manipulation would also be interesting to pursue.

5.7 Acknowledgements

We would like to thank Eric van Damme, secretary-treasurer of the Game Theory Society, for letting us use the ballot data of the 2003 Game Theory Society council election in our experiments. We would also like to thank Steven Brams for introducing us to the problem and for his valuable comments and pointers to the literature.

5.8 Summary of contributions

In this research, we have accomplished the following.

1. Established that calculating the outcome of a fixed-size minimax election is NP-hard.
2. Specified an approximation algorithm of ratio 3 for FSM.
3. Suggested a large class of heuristics for solving FSM and evaluated their performance on randomly generated and real-world ballot input.
4. Established that fixed-size minimax is vulnerable to manipulation by insincere voters.
5. Proved that our 3-approximation algorithm for fixed-size minimax is immune to manipulation by insincere voters.

References

[1] Metacritic. `http://www.metacritic.com/`.

[2] Rotten Tomatoes. `http://www.rottentomatoes.com/`.

[3] Andrew W. Appel. Effective audit policy for voter-verified paper ballots. In *Presented at the annual meeting of the American Political Science Association*, Chicago, Illinois, August 2007.

[4] Kenneth J. Arrow. A difficulty in the concept of social welfare. *Journal of Political Economy*, 58(4):328–346, August 1950.

[5] Chandrajit L. Bajaj. The algebraic degree of geometric optimization problems. *Discrete & Computational Geometry*, 3:177–191, 1988.

[6] Michel Balinski and Rida Laraki. A theory of measuring, electing, and ranking. *Proceedings of the National Academy of Sciences of the United States of America*, 104(21):8720–8725, May 2007.

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

[8] John J. Bartholdi III and James B. Orlin. Single transferable vote resists strategic voting. *Social Choice and Welfare*, 8:341–354, 1991.

[9] John J. Bartholdi III, Craig A. Tovey, and Michael A. Trick. How hard is it to control an election? *Mathematical Computational Modeling*, 16(8/9):27–40, 1992.

[10] Duncan Black. On the rationale of group decision-making. *Journal of Political Economy*, 56(1):23–34, February 1948.

[11] Prosenjit Bose, Anil Maheshwari, and Pat Morin. Fast approximations for sums of distances, clustering and the Fermat-Weber problem. *Computational Geometry: Theory and Applications*, 24(3):135–146, April 2003.

[12] Steven J. Brams. *Mathematics and democracy: designing better voting and fair-division procedures*. Princeton University Press, Princeton, New Jersey, 2008.

[13] Steven J. Brams and Peter C. Fishburn. Approval voting. *American Political Science Review*, 72(3):831–847, September 1978.

[14] Steven J. Brams and Peter C. Fishburn. *Approval Voting*. Birkhäuser, Boston, Massachusetts, 1983.

[15] Steven J. Brams, D. Marc Kilgour, and M. Remzi Sanver. A minimax procedure for negotiating multilateral treaties. Technical report, Department of Politics, New York University, New York, New York, 2004.

[16] Steven J. Brams, D. Marc Kilgour, and M. Remzi Sanver. A minimax procedure for electing committees. Technical report, Department of Politics, New York University, New York, New York, 2006.

[17] Jack Brimberg. The Fermat-Weber problem revisited. *Mathematical Programming*, 71:71–76, 1995.

[18] Alonzo Church. An unsolvable problem of elementary number theory. *American Journal of Mathematics*, 58:345–363, 1936.

[19] Vincent Conitzer, Jérôme Lang, and Tuomas Sandholm. How many candidates are needed to make elections hard to manipulate? In *Proceedings of the 9th Conference on Theoretical Aspects of Rationality and Knowledge (TARK-03)*, pages 201–214, Bloomington, Indiana, 2003.

[20] Vincent Conitzer and Tuomas Sandholm. Complexity of manipulating elections with few candidates. In *Proceedings of the Eighteenth National Conference on Artificial Intelligence (AAAI '02)*, pages 314–319, Edmonton, Canada, 2002.

[21] Vincent Conitzer and Tuomas Sandholm. Universal voting protocol tweaks to make manipulation hard. In *Proceedings of the 18th International Joint Conference on Artificial Intelligence (IJCAI)*, pages 781–788, Acapulco, Mexico, 2003.

[22] Lorrie Faith Cranor. *Declared-Strategy Voting: An Instrument for Group Decision-Making*. PhD thesis, Department of Computer Science, Washington University, St. Louis, Missouri, December 1996.

[23] Lorrie Faith Cranor and Ron K. Cytron. Towards an information-neutral voting scheme that does not leave too much to chance. In *Midwest Political Science Association Annual Meeting*, April 1996.

[24] Anthony Downs. *An Economic Theory of Democracy*. Harper, New York, New York, 1957.

[25] Edith Elkind and Helger Lipmaa. Hybrid voting protocols and hardness of manipulation. In *The 16th Annual International Symposium on Algorithms and Computation (ISAAC 2005)*, pages 206–215, Sanya, Hainan, China, December 2005.

[26] Eithan Ephrati and Jeffrey S. Rosenschein. The Clarke tax as a consensus mechanism among automated agents. In *Proceedings of the Ninth National Conference on Artificial Intelligence (AAAI '91)*, pages 173–178, Anaheim, California, 1991.

[27] Eithan Ephrati and Jeffrey S. Rosenschein. Multi-agent planning as a dynamic search for social consensus. In *Proceedings of the Thirteenth International Joint Conference on Artificial Intelligence (IJCAI '93)*, pages 423–429, Chambéry, France, 1993.

[28] Moti Frances and Ami Litman. On covering problems of codes. *Theory of Computing Systems*, 30(2):113–119, March 1997.

[29] Leszek Gasieniec, Jesper Jansson, and Andrzej Lingas. Efficient approximation algorithms for the Hamming center problem. In *Proceedings of the Tenth Annual ACM-SIAM Symposium on Discrete Algorithms (SODA '99)*, pages 905–906, Baltimore, Maryland, 1999.

[30] Allan Gibbard. Manipulation of voting schemes: A general result. *Econometrica*, 41(3):587–601, May 1973.

[31] Bev Harris. *Black box voting: ballot tampering in the 21st century*. Plan Nine Publishing, High Point, North Carolina, 2003.

[32] D. Marc Kilgour, Steven J. Brams, and M. Remzi Sanver. How to elect a representative committee using approval balloting. In Bruno Simeone and Friedrich Pukelsheim, editors, *Mathematics and Democracy: Recent Advances in Voting Systems and Collective Choice*. Springer, forthcoming, 2007.

[33] Stephen C. Kleene. *Mathematical Logic*. Wiley, New York, New York, 1967.

[34] J. Kevin Lanctot, Ming Li, Bin Ma, Shaojiu Wang, and Louxin Zhang. Distinguishing string selection problems. *Information and Computation*, 185:41–55, August 2003.

[35] Rob LeGrand. Approval strategy A. `http://groups.yahoo.com/group/election-methods-list/message/9713`, May 2002.

[36] Rob LeGrand. Analysis of the minimax procedure. Technical Report WUCSE-2004-67, Department of Computer Science and Engineering, Washington University, St. Louis, Missouri, November 2004.

[37] Ming Li, Bin Ma, and Shaojiu Wang. Finding similar regions in many strings. In *Proceedings of the Thirty-First Annual ACM Symposium on Theory of Computing (STOC '99)*, volume Atlanta, Georgia, pages 473–482, 1999.

[38] Samuel Merrill III. *Making Multicandidate Elections More Democratic*. Princeton University Press, Princeton, New Jersey, 1988.

[39] David G. Mitchell. A sat solver primer. *EATCS Bulletin (The Logic in Computer Science Column)*, 85:112–133, February 2005.

[40] Peter Neumann, Rebecca Mercuri, and Lauren Weinstein. Internet and electronic voting. *The Risks Digest, ACM Committee on Computers and Public Policy*, 21(14):559–566, December 2000.

[41] Lawrence Norden. The machinery of democracy: Voting system security, accessibility, usability, and cost. Technical report, Brennan Center for Justice, New York University School of Law, New York, New York, October 2006.

[42] Hannu Nurmi. *Voting Paradoxes and How to Deal with Them*. Springer-Verlag, Berlin, Germany, 1999.

[43] Mike Ossipoff. Strategy for approval vote. `http://www.barnsdle.demon.co.uk/vote/apstr2.html`, September 2002.

[44] Guy Ottewell. The arithmetic of voting. *In Defense of Variety*, 4:42–44, 1977.

[45] Christos H. Papadimitriou. On the complexity of integer programming. *Journal of the ACM*, 28(4):765–768, October 1981.

[46] David M. Pennock, Eric Horvitz, and C. Lee Giles. Social choice theory and recommender systems: Analysis of the axiomatic foundations of collaborative filtering. In *Proceedings of the Seventeenth National Conference on Artificial Intelligence (AAAI '00)*, pages 729–734, Austin, Texas, 2000.

[47] William Poundstone. *Gaming the vote: why elections aren't fair (and what we can do about it)*. Hill and Wang, New York, New York, 2008.

[48] Ariel D. Procaccia and Jeffrey S. Rosenschein. Junta distributions and the average-case complexity of manipulating elections. *Journal of Artificial Intelligence Research*, 27, December 2006.

[49] Ariel D. Procaccia, Jeffrey S. Rosenschein, and Aviv Zohar. Multi-winner elections: Complexity of manipulation, control and winner-determination. In *Proceedings of the Twentieth International Joint Conference on Artificial Intelligence (IJCAI '07)*, pages 1476–1481, Hyderabad, India, January 2007.

[50] Marc Renault. Four proofs of the ballot theorem. *Mathematics Magazine*, 80(5):345–352, December 2007.

[51] Donald G. Saari. *Basic Geometry of Voting*. Springer-Verlag, Berlin, Germany, 1995.

[52] Donald G. Saari. *Choatic Elections!: A Mathematician Looks at Voting*. American Mathematical Society, Providence, Rhode Island, 2001.

[53] Mark Allen Satterthwaite. Strategy-proofness and Arrow's conditions: Existence and correspondence theorems for voting procedures and social welfare functions. *Journal of Economic Theory*, 10(2):187–217, April 1975.

[54] Christopher G. Small. A survey of multidimensional medians. *International Statistical Review*, 58(3):263–277, December 1990.

[55] Alan M. Turing. Intelligent machinery. Technical report, National Physical Laboratory, Teddington, Middlesex, England, 1948.

[56] Y. Vardi and Cun-Hui Zhang. A modified Weiszfeld algorithm for the Fermat-Weber location problem. *Mathematical Programming*, 90(3):559–566, 2001.

[57] John von Neumann and Oskar Morgenstern. *Theory of Games and Economic Behavior*. Princeton University Press, Princeton, New Jersey, 1953.

[58] Alfred Weber. *Alfred Weber's Theory of the Location of Industries*. University of Chicago Press, Chicago, Illinois, 1929. Translated from the German by C. J. Freidrich.

[59] Robert J. Weber. Comparison of voting systems, 1977.

Vita

Robert Hampton LeGrand III

Date of birth	9 September 1976
Place of birth	San Angelo, Texas
Degrees	B.S. *Magna cum laude*, Computer Science, August 1998, Texas A&M University, College Station M.C.S., Computer Science, December 1999, Texas A&M University, College Station Ph.D., Computer Science, May 2008, Washington University, St. Louis
Professional societies	Association for Computing Machinery
Publications	Rob LeGrand, Evangelos Markakis and Aranyak Mehta. Some results on approximating the minimax solution in approval voting. In *Proceedings of the Sixth International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS '07)*, pages 1185–7. International Foundation for Autonomous Agents and Multiagent Systems (IFAAMAS), 2007. Delvin Defoe, Rob LeGrand and Ron K. Cytron. On the connection between functional programming languages and real-time Java scoped memory. In *Proceedings of the 5th International Workshop on Java Technologies for Real-time and Embedded Systems*, pages 73–82. ACM Press, 2007. Delvin Defoe, Rob LeGrand and Ron K. Cytron. Cost analysis for real-time Java scoped-memory areas. *Journal of Systemics, Cybernetics and Informatics*, Volume 5, No. 4, pages 70–77. International Institute of Informatics and Systemics, 2007. Rob LeGrand, Evangelos Markakis and Aranyak Mehta. Approval voting: Local search heuristics and approximation algorithms for the minimax solution. Presented at the 1st International Workshop on Computational Social Choice, Amsterdam, Netherlands, December 2006. Delvin Defoe, Rob LeGrand and Ron K. Cytron. Asymptotic analysis for real-time Java scoped-memory areas. In *Proceedings of the 4th International Conference on Computing, Communications and Control Technologies (CCCT 2006)*, Volume II, pages 131–138. International Institute of Informatics and Systemics, 2006.

Tobias Mann, Morgan Deters, Rob LeGrand and Ron K. Cytron. Static determination of allocation rates to support real-time garbage collection. In *Proceedings of the 2005 ACM SIGPLAN/SIGBED Conference on Languages, Compilers, and Tools for Embedded Systems (LCTES'05)*, pages 193–202. ACM Press, 2005.

Rob LeGrand. Analysis of the minimax procedure. Technical Report WUCSE-2004-67, Department of Computer Science and Engineering, Washington University, St. Louis, Missouri, November 2004.

May 2008

Approval & Declared-Strategy Voting, LeGrand, Ph.D., 2008