UNIVERSITY OF
LIVERPOOL

Manipulative Voting Dynamics

Thesis submitted in accordance with the requirements of
the University of Liverpool for the degree of Doctor in Philosophy

by

Neelam Gohar

Supervisor: Professor Paul W. Goldberg

February 2012

Abstract

In AI, multi-agent decision problems are of central importance, in which independent agents aggregate their heterogeneous preference orders among all alternatives and the result of this aggregation can be a single alternative, corresponding to the groups collective decision, or a complete aggregate ranking of all the alternatives. Voting is a general method for aggregating the preferences of multiple agents. An important technical issue that arises is manipulation of voting schemes: a voter may be to make the outcome most favorable to itself (with respect to his own preferences) by reporting his preferences incorrectly. Unfortunately, the Gibbard-Satterthwaites theorem shows that no reasonable voting rule is completely immune to manipulation, recent literature focussed on making the voting schemes computationally hard to manipulate. In contrast to most prior work Meir et al. [40] have studied this phenomenon as a dynamic process in which voters may repeatedly alter their reported preferences until either no further manipulations are available, or else the system goes into a cycle. We develop this line of enquiry further, showing how potential functions are useful for showing convergence in a more general setting. We focus on dynamics of weighted plurality voting under sequences made up various types of manipulation by the voters. Cases where we have exponential bounds on the length of sequences, we identify conditions under which upper bounds can be improved. In convergence to Nash equilibrium for plurality voting rule, we use lexicographic tie-breaking rule that selects the winner according to a fixed priority ordering on the candidates. We study convergence to pure Nash equilibria in plurality voting games under unweighted setting too. We mainly concerned with polynomial bounds on the length of manipulation sequences, that depends on which types of manipulation are allowed. We also consider other positional scoring rules like Borda, Veto, k-approval voting and non positional scoring rules like Copeland and Bucklin voting system.

This thesis is dedicated to my family specially my parents, my grandfather and my uncle who have always stood by me and supported me throughout my life. They have been a constant source of love, concern, support and strength all these years. I warmly appreciate their generosity and understanding.

Acknowledgements

My thanks and appreciation to Professor Paul W. Goldberg whose encouragement and guidance from the initial to the final level enabled me to develop an understanding of the subject. His insightful comments and constructive criticisms at different stages of my research were thought-provoking and they helped me focus my ideas.

I am also thankful to my advisor Piotr Krysta for his encouragment.

I owe my deepest gratitude to Frontier Women University Peshawar and Higher Education Commission of Pakistan for the financial support.

I am also thankful to the system staff, admin staff and teaching staff of Department of Computer Science, University of Liverpool who have made available their support in a number of ways.

I must acknowledge as well the many friends and colleagues for encouragement and emotional support.

Most importantly, none of this would have been possible without the love and patience of my family. I would like to express my heart-felt gratitude to my family.

Lastly, I offer my regards and blessings to all of those who supported me in any respect during the completion of this thesis.

Neelam Gohar

Contents

List of Figures

1

Introduction

This introductory chapter contains the following sections: Section 1.1 presents the background knowledge, some relevant recent work and a short overview of results. In Section 1.2 we summarize the related work, Section 1.3 gives a brief problem statement which also describes the contributions and significance of the problem. Section 1.4 gives a structure of every chapter within this thesis.

1.1 Background

One of the newer areas explored in artificial intelligence is multi-agent systems, which analyzes interactions between multiple agents, each of which with its own personal objectives. For example, each router in the Internet might be an agent, and when a packet forwarded from source and destination, each router prefers to do as little work as possible and another example is dividing processes between processors.

One of the actively growing subareas explored in multi-agent systems is computational social choice theory that provides theoretical foundation for preference aggregation and collective decision-making in multi-agent domains. Computational social choice is concerned with the application of techniques developed in computer science, such as complexity analysis or algorithm design, to the study of social choice mechanisms, such as voting . It seeks to import concepts from social choice theory into AI and computing. For example, social welfare orderings developed to analyze the quality of resource allocations in human society are equally well applicable to problems in multi-agent systems or network design.

People often have to reach a joint decision even though they have conflicting preferences over the alternatives. The joint decision can be reached by an informal negotiating process or by a carefully specified protocol. Over the course of the past decade or so, computer scientists have also become deeply involved in this study. Social choice theory investigates many kinds of multiperson decision-making problems. Multiperson decision-making problems are important, frequently encountered processes and many real world problems involve multiple decision makers.

Within computer science, there is a number of settings where a decision must be made based on the conflicting preferences of multiple parties. For example determining whose job gets to run first on a machine, whose network traffic is routed along a particular link, or what advertisement is shown next to a page of search results. The paradigms of computer science give a different and useful perspective on some of the classic problems in economics and related disciplines. For example, various results in economics prove the existence of an equilibrium, but do not provide an efficient method for reaching such an equilibrium. Also greater computing power and better algorithms, have made it possible to run computationally demanding protocols that lead to much better outcomes. Preference aggregation has been extensively studied in social choice theory and voting is the most general preference aggregation scheme.

A natural and very general approach for deciding among multiple alternatives is to vote over them. Voting is one of the most popular way of reaching common decisions. The study of elections is a showcase area where interests come from computer science specialists as theory, systems, and AI and such other fields as economics, business, operations research, and political science. Social choice theory deals with voting scenarios, in which a set of individuals must select an outcome from a set of alternatives. In the general theory of voting, agents can do more than vote for a single alternative, usually each individual ranks the possible alternatives and a voting rule selects the winning alternative based on the voters' preferences. A voting rule takes as input a collection of votes, and as output returns the winning alternative. For example, a simple rule known as the Plurality rule chooses the alternative that is ranked first the most often. In this case, the agents do not really need to give a full ranking, it suffices to indicate one's most-preferred alternative, so each voter is in fact just voting for a single alternative.

Voting is a well-studied method of preference aggregation, in terms of its theoretical properties, as well as its computational aspects [11, 54]; various practical, implemented applications that use voting exist [18, 32, 35]. Voting is an essential element of mechanism design (how privately known preferences of many people can be aggregated towards a social choice) for multi-agent systems, and applications built on such systems, which includes ad hoc networks, virtual organizations, and a crucial aspect of decision support tools implementing online deliberative assemblies.

[32] present the architecture and implementation status of an agent-based movie recommender system. In particular, how the agent stores and uses user preferences to find recommendations that are likely to be useful to the user. They have adapted methods developed in the voting theory literature to find compromises between possibly disparate preference as voting is a well understood mechanism for reaching consensus. [35] highlighted the usage of user preferences in automated meeting scheduling system (a software that automate and share information processing tasks of associated human users). In this modern world of processes and agents, it is not just people whose preferences must be aggregated but the preferences of computational agents must also be aggregated. In both artificial intelligence and system communities a great array of issues have been proposed as appropriate to approach via voting systems. These issues range from spam detection to web search engines to planning in multi-agent systems and much more (e.g, [19, 20, 23, 51]).

Recent work in the AI literature has studied the properties of voting schemes for performing preference aggregation [11, 23, 54]. A social choice function is a function that takes lists of people's ranked preferences and outputs a single alternative (the "winner" of the election). A good social choice function represents the "will of the people". Rather than just choosing a winning alternative, most of the voting rules can also be used to find an aggregate ranking of all the alternatives. For example, we can sort the alternatives by their Borda score, thereby deciding not only on the "best" alternative but also on the second-best, and so on. There are numerous applications of this that are relevant to computer scientists, for example one can pose the same query to multiple search engines, and combine the resulting rankings of pages into an aggregate ranking.

Researchers in social choice theory have studied extensively the properties of various families of voting rules, but have typically neglected computational issues. Sincere voting assumes that voters always choose their most preferred candidates and/or parties. It has been argued in both the formal and empirical literature, however, that voters may not always vote for their most preferred candidates. Sincere voting is voting in accordance with one's true preferences over alternatives. While strategic voting is voting over assumed outcomes, in which a voter uses skills to determine an action that secures a best possible outcome in his view. This is the trade-off a rational voter faces in an election. She must balance her relative preference for the different candidates against the relative likelihood of influencing the outcome of the election [7]. However, in voting one of the major technical issues is manipulation of voting schemes. Elections are endangered not only by the organizers but also by the voters (manipulation), who might be tempted to vote strategically (that is, not according to their true preferences) to obtain their preferred outcome.

Manipulation in voting is considered to be any scenario in which a voter reveals false preferences in order to improve (with respect to his own preferences) the outcome of the election. A manipulative vote leads to successful manipulation if it changes the election outcome to one preferred by that particular voter. Since voters are considered rational agents, who want to maximize their own utility, their best strategy may be to manipulate an election if this will gain them a higher utility. This has various negative consequences; not only do voters spend valuable computational resources determining which lie to employ, but worse, the outcome may not be one that reflects the social good. The Gibbard-Satterthwaite result [33, 58] states that any non-dictatorial voting scheme is vulnerable to manipulation, that is, there will always be a preference profile in which at least one of the individuals has an incentive not to elicit her true preferences. Gibbard-Satterthwaite, Gardenfors, other such theorems open doors to strategic voting, which makes voting a richer phenomenon. In order to achieve some standard of non-manipulability in voting schemes, in all the previous work the complexity of the manipulation is considered where one could try to avoid manipulation by using protocols where determining a beneficial manipulation is hard; for a survey, see [25].

Complexity offers a powerful tool to frustrate manipulators who seek to manipulate or control election outcomes. The motivation for studying complexity issues comes from the Gibbard-Satterthwaite Theorem showing that every reasonable election system can be manipulated [33, 58]. So better design of election systems cannot prevent manipulation. Computational complexity can serve as a barrier to dishonest behavior by the voters, and Bartholdi et al. [4] proposed classifying voting rules according to how difficult it is to manipulate them. They argued that well-known voting rules such as Plurality, Borda, Copeland and Maximin are easy to manipulate. Since then, the computational complexity of manipulation under various voting rules received considerable attention in the literature, both from the theoretical and from the experimental perspective (see, [61, 63]) and the recent surveys [13, 24, 60]. The complexity of the manipulation problem for a single voter is quite well understood and this problem is efficiently solvable for most common voting rules with notable exception of single transferable vote (STV) [4, 5], the more recent work has focussed on coalitional manipulation, i.e., manipulation by multiple, possibly weighted voters.

We have not dealt with computational complexity issues here, we are considering bounds on the length of sequences of manipulations that voters can perform. Despite the basic manipulability of reasonable voting systems, it would still be desirable to find ways to reach a stable result, which no agent will be able to change. One possibility is the convergence of myopic improvement dynamics, where strategic voters change their votes step by step in order to get a better outcome. A voting profile is in equilibrium, when no voter can make his more preferable candidate to get elected. This iterative voting is used, in the real world, in various situations, such as elimination decisions in various "reality shows". The study of dynamics in strategic voting is very interesting and highly relevent to the multi-agent systems, as it helps to tackle the multi-agent decision making problems, where autonomous agents (that may be distant, self-interested and/or unknown to each other) have to choose a joint plan of action or allocate resources or goods. We work with different types of moves that leads to successful manipulation.

1.1.1 Manipulative dynamics

Meir et al. [40] have studied this phenomenon as a dynamic process in which voters may repeatedly alter their reported preferences until either no further manipulations are available, or else the system goes into a cycle. Here we develop this line of enquiry further, showing how potential functions are useful for showing convergence in a more general setting. We focus on Plurality voting with weighted voters, and obtain bounds on the lengths of sequences of manipulations, that depend on which types of manipulation are allowed. We analyze the sequences of changes of votes that may result from various voters performing manipulations and we bound the length of sequences of votes with the help of potential functions. Potential functions are valuable for proving the existence of pure Nash equilibria and the convergence of best response dynamics. Even-Dar et al. [21] introduced the idea of using a potential function to measure closeness to a balanced allocation, and used it to show convergence for sequences of randomly-selected "best response" moves in a load-balancing setting in which tasks may have variable weights and resources may have variable capacities. We study convergence to pure Nash equilibria in Plurality voting games. In such a game, the voters strategically choose a candidate to vote for, and the winner is determined by the Plurality rule. A voting profile is in equilibrium, when no voter can change his vote so that a more preferable candidate gets elected. In our model, we assume the elementary stepwise system (ESS), i.e, at every state only one voter is allowed to move. Thus, a voter switches his support to another candidate in response to the moves of other voters so that a sequence of moves occurs. This sequence may stop at a steady state where no voter wishes to switch, or may continue indefinitely. This steady state is called the Nash equilibrium. The concept of Nash equilibria has become an important mathematical tool in analyzing the behavior of selfish users in non-cooperative systems [50] i.e., games where players act in an independent and selfish way. Such iterative games reach an equilibrium point from either an arbitrary or a truthful initial state. We focus more on weighted voting setting, where voters may have different weights in elections. The topic of convergence to stable outcomes in strategic voting settings is interesting to Artificial Intelligence. We are mainly concerned with polynomial bounds on the length of manipulation sequences.

For our model, we consider an election with m alternatives, and with n voters each of whom has a total ordering of the alternatives. A system comprised of finite number of states and transitions occur from state to state when voters change their mind and support an alternative candidate. Every state is mapped into a real value by the potential function and transitions cause the potential to increase or decrease. States can be defined as the profiles of "declared preferences" of voters. A transition is a manipulation move by a single voter. We focus on the Plurality voting rule because Plurality has been shown to be particularly susceptible to manipulation, both in practice and theory [29, 57]. We consider other voting rules as well. We assume that voters have knowledge regarding the currently supported candidates of the other voters in case of Plurality voting. For other positional scoring rules, voters have knowledge regarding the total scores of all candidate at a state. Complete information is not needed in such a set up. Voters manipulate according to their true preferences. Voters change their vote (make manipulation) after observing the current state and outcome. If voters have their true preferences then a manipulator changes his preference list in favour of a less preferred candidate and make him a new winner if he does not like the current winner and it results in a better winner (for that voter) than the current winner. In case if voters declared preferences that are different than that of his true preferences and the outcome is not favourable for him, then he changes his preference list in favour of his most-preferred candidate that can win. If a voter cannot affect the outcome at some state, he simply keeps his current preference list. This process of manipulation proceeds in turns, where a single voter changes his preference list at each step/turn. Voters take turns modifying their votes; these manipulations are according to the way in which they affect the outcome of the election. The process ends when no voter has objections and the outcome is set by the last state.

In manipulation dynamics, voters change their mind to make a "manipulative vote" that changes the outcome of the election. We are considering bounds on the length of sequences of manipulations that can take place. We also consider voting rules with lexicographic tie-breaking rule that depends strictly on linear preference orders to choose a winner in case of ties. In most of our results we use a weighted voting system. A weighted voting system is the one in which the preferences of some voters carry more weight than the preferences of other voters. Some of our results have dependence on the voters' weights. We have results for different weight settings. We used weighted votes as the introduction of weights generalizes the usability of voting schemes.

If voters are allowed to vote simultaneously, then this iterative process may never converge to an equilibrium [40], that's the reason in our model at every state only one voter is allowed to move.. The system is modeled as a sequence of steps and in each step one voter switches from one candidate to another. We establish bounds on the length of sequences of manipulations that voters can perform. We consider these with respect to the different types of moves that leads to successful manipulation. We do not concern ourselves with the impact of manipulation on social welfare; we treat manipulation as an "occupational hazard" and ask the question: in a system where manipulation may occur, when can we guarantee that the voters will end up satisfied with their (possibly manipulative) votes, in the context of the votes offered by the others? Put another way, we posit that in various real-world situations, it may be better to reach a poor decision than no decision at all. We can regard the voting system as a game in which each voter has, as pure strategies, the set of all votes he may make. (In Plurality voting, a vote is just the choice of a single "preferred" candidate.) Each voter has a type, consisting of a ranking of the candidates that represents his real preferences. We ask whether pure Nash equilibria exist for any set of voter types, and more importantly whether such an equilibrium can be reached via a sequence of myopic changes of vote, by the players. This can be regarded as a very simplistic model of a negotiation process amongst the voters, and we would like to ensure that it does not end in deadlock.

Our main issue is the proof of termination, and the bounds on the length of sequences of manipulations that can take place. We are interested in bounds on the number of possible steps that are purely in terms of number of candidates m and number of voters n (and independent of the total size of the weight which can be quite large). An important property of the voting rules discussed in [4] is that they may produce multiple winners. In real-life settings, when an election ends in a tie, it is not uncommon to choose the winner using a tie-breaking rule that is non-lexicographic in nature. When an election under a particular voting rule ends in a tie, we use lexicographic tie-breaking rule that uses a fixed linear order on candidates to break ties.

1.1.2 Tactical voting dynamics

Sincere voting is voting in accordance with one's true preferences over alternatives. While, Strategic voting is voting over assumed outcomes, in which a voter uses skills to determine an action that secures a best possible outcome in his view. Strategic voting under Plurality rule refers to a voter deserting a more preferred candidate with a poor chance of winning for a less preferred candidate with a better chance of winning [26]. The logic of tactical/strategic voting, of course, is that of Duverger's law, which states that the supporters of a small party would not "waste" their votes by voting for their most preferred party (candidate) because it does not have a chance to win under a Plurality system with single member districts. Instead, they vote for the major party that is most acceptable to them and that has a chance of winning.

Let us suppose a voter believes that her most preferred candidate has little chance of competing for the lead in the election. Voting for such a candidate may be a "waste". The voter may decide to switch her vote to the expected leading candidate she most prefers in order to make her vote "pivotal" in determining a more preferable outcome. This is the trade-off a rational voter faces in an election. Strategic voting is an important component of Duverger's Law, if voters are rational, they end up voting for one of the two leading candidates [6].

Another voting dynamics we consider is that of tactical voting dynamics in which a voter changes his mind to make a tactical vote according to the *m*ind changing rule as defined later. The purpose of making a tactical vote is to increase the score of a preferred candidate which may or may not lead to changing an election outcome. The set of all voters' declared preferences is summarized in the concept of a state. A transition occurs from current state to a new state when a voter changes his mind and chooses a different candidate to support (under Plurality). In a state of a system, each voter determines whether it can improve (w.r.t his own true preferences) the outcome by altering its own vote while assuming that all other votes remain the same. A mind changing rule is as follows: a voter considers all alternative candidates that he ranked higher than the current winner of the state. He can then change his support to that alternative who has currently most votes, breaking ties in favour of his own preference. With this mind changing, a transition occurs and the system enters into a new state from the current state. At each iteration, state of the system associates each voter with a candidate currently supported by that voter under Plurality rule. Tactical voting is different than the manipulation dynamics because it simply raises the votes of an expected leading candidate he most prefers. In this kind of voting a voter instead of wasting vote by voting for his most preferred candidate who does not have a chance to win, its better to vote for a candidate and raise the score of that alternative who is more acceptable to him and has a chance of winning. We analyze the sequences of votes that may result from various voters performing tactical vote in both weighted and unweighted settings.

Our results. We focus on Plurality voting with weighted voters, in which each voter reports a single preferred candidate. A voter's weight is fixed throughout. The *score* of a candidate is the total weight of voters who support that candidate, hence the winning candidate is the one with highest score, and we assume a standard lexicographic tie-breaking rule in which a candidates have a given total order on them that determines the winner if two or more of them have maximal score. [40] also considers this tie-breaking rule, and compares it with a randomized one.

We investigate rate of convergence, i.e. the number of steps of manipulation that may be needed to reach a pure Nash equilibrium. We focus on types of manipulations where there are no cycles in the state/transition graph, where convergence is guaranteed, and we analyze bounds on the number of steps required. The rate of convergence will be expressed as a function of the number of voters n, the number of candidates m, the ratio $w_{\max}$ of maximum to minimum weights, and the number of distinct weights K. Guaranteed convergence may also depend on types of manipulation available; a classification is given in Chapter 2.

We identify combinations of types of moves that are able to lead to cycles of manipulation moves. We consider combinations of move types where convergence is guaranteed, and exhibits various potential functions to obtain upper bounds on the number of manipulation steps possible. Alternative types of moves seem to require alternative potential functions, and we give upper bounds as expressions in terms of the parameters n, m and K.

1.2 Related work

Meir et al. [40] study the convergence of pure strategy Nash equilibria in Plurality games. They showed that myopic best response dynamics may cycle, even when start from a truthful voting profile, for both deterministic and randomized tie-breaking schemes. Our work extends that of Meir et al. [40] in that we consider weighted voters as well. The notion of voting dynamics as well as convergence of voting games particularly Plurality exist in previous research. For deterministic tie-breaking scheme, we demonstrate that if one excludes certain deviations than an improvement path is guaranteed. There is also a number of very recent papers apart from Meir et al. [40] that analyze strategic behaviour in voting using tools of non-cooperative game theory [14, 62]. [14] consider the setting where all voters are strategic, where an election can be viewed as a game, and the election outcomes correspond to Nash equilibria of this game. They analyze two variants of Plurality voting, namely, simultaneous voting, where all voters submit their votes at the same time, and sequential voting, where the voters express their preferences one by one. Sequential voting always has an equilibrium in pure strategies. They take the approach suggested by Farquharson [26] and view manipulation as an unavoidable attribute of an electoral system with rational voters.

The model in [62] consider a voting process in which voters vote one after another as an extensive-form game. They study equilibria of sequential voting for a number of voting rules (including Plurality), however, they use a deterministic tie-breaking rule. Feddersen et al. [27] study a Plurality voting game in which voters are strategically rational and search for different equilibria choices. However, in order to reach an equilibrium, they limit the possible preference choices to single-peaked preferences. Also the model assume that both voters and candidates possess complete information and voters use only pure strategies. There have been several studies applying Nash equilibrium to dynamics process, particularly in allocation of public goods. Much of the work is summarized in [38]. They characterize all Nash equilibria, using different approaches under the restriction that preferences are single-peaked preferences like Feddersen et al. [27]. Hinich et al [37] change single-peaked preferences to a specific probabilistic model of voters over an Euclidean space of candidates, where individuals vote randomly according to probability functions based on their preferences, and where the candidates maximize expected votes. Another relevant work is by [41], they focus on the existence and uniqueness of strong equilibria in Plurality games. Strong equilibrium is a weaker concept, still stronger than Nash equilibrium. No coalitional manipulation can get an incentive by making a coordinated diversion in case of strong equilibrium. Same approach is used in [15] by considering dominant strategies in Plurality voting. To reach an equilibrium [59] use a specific voting rule and Euclidean preferences. They proved that under the Euclidean preferences the majority rule converges and that there is a unique equilibrium. All above papers assume that voters have some knowledge of the other voters' preferences.

Another model was suggested in [44] found Nash equilibrium for positional scoring rule like approval, Borda and Plurality. They assume voters have some knowledge about preferences of other voters but not every election converges. Iterative voting with Plurality was examined by [9] limiting voters' information about others voters' preferences e.g. when voters are myopic and also assuming that voters have sufficient information about all voters. The focus of this study is on the role played by the state of knowledge of the agents. A related dynamical model was considered by [2], they examined the conditions to achieve an equilibrium in iterative games using Plurality voting rule, in which a group of agents make a sequence of collective decisions on whether to remain in the current state of the system or switch to an alternative state. At each step, a voter is selected at random and may propose a single alternative to the one currently winning the election; a pairwise vote take place between the current winner and the new alternative. As in [40] cyles may arise; the ability of the chosen voter to select a single alternative for a pairwise election indeed makes it possible to exhibit cycles which cannot be escaped. An iterative procedure for reaching solution was also used by [17] but they use money like value among voters/agents. They consider how agents can come to a consensus without needing to reveal full information about their preferences, and without needing to generate alternatives prior to the voting process.

The study of manipulability of various voting rules, i.e., understanding the algorithmic complexity of individual or coalitional manipulation, is an active research area. Much of this work views manipulation as a type of adversarial behavior that needs to be prevented, either by imposing restrictions on voters preferences, or by identifying a voting rule for which manipulation is computationally hard, preferably in the average case rather than in the worst case. Making manipulation difficult to compute is a way followed recently by several authors [4, 5, 10, 11, 12] who address to the computational complexity of manipulation for Plurality and other voting rules.

[4] showed how computational complexity protect the integrity of social choice. While many standard voting schemes can be manipulated with only polynomial computational effort, they exhibit a voting rule that efficiently computes winners but is computationally resistant to manipulation. [5] showed that Single Transferable Vote (STV) is apparently unique among voting schemes in actual use today in that it is computationally resistant to manipulation. Under STV each voter submits a total order of the candidates. STV tallies votes by reallocating support from weaker candidates to stronger candidates and excess support from elected candidates to remaining contenders. [10] asked the question: how many candidates are needed to make elections hard to manipulate? They answer this question for the voting protocols: Plurality, Borda, STV, Copeland, Maximin, regular Cup, and randomized Cup.

The main manipulation question studied in [11] is that of coalitional manipulation by weighted voters. They characterize the exact number of candidates for which manipulation becomes hard for the Plurality, Borda, STV, Copeland, Maximin, Veto, Plurality with runoff etc. They show that for simpler manipulation problems, manipulation cannot be hard with few candidates. Some earlier work show that high complexity of manipulation rely on both the number of candidates and the number of voters being unbounded. [12] derived hardness results for the more common setting where the number of candidates is small but the number of voters can be large. They show that with complete information about the others' votes, individual manipulation is easy, and coalitional manipulation is easy with unweighted voters.

1.3 Problem statement

We study the convergence to pure startegy Nash equilibria in Plurality voting games. We also study other positional scoring rules and some non positional scoring rules as well. We consider election with m alternatives and with n voters each of whom has a total ordering of the alternatives. In such a game, the voters srategically choose a candidate to vote for, and the winner is determined by the Plurality/other voting rules. Voters take turns modifying their votes; these manipulations are classified according to the way in which they affect the outcome of the election. We focus on achieving a stable outcome taking strategic behaviour into account. A voting profile is in equilibrium, when no voter can change his vote so that his more preferable candidate gets elected. We investigate bounds on the number of iterations that can be made for different voting rules. We focus on the weighted voting settings, where voters may have different weights in elections. We consider equi-weighted votes too. An important property of the voting rules is that they may produce multiple winners, i.e., they are, in fact, voting correspondences. When an election ends in a tie, we choose the winner using a tie-breaking rule that is lexicographic in nature.

1.3.1 Contribution and comparison with previous work

Most of the previous work about manipulation dealt with computational complexity issues, where one could try to avoid manipulation by using protocols where determining a beneficial manipulation is hard [11, 25]. The well-known Gibbard-Satterthwaite theorem [33, 58] states that a reasonable voting rule is completely immune to strategic manipulation. This makes the analysis of election a complicated and challenging task. One approach to understanding voting is the analysis of solution concepts such as Nash equilibria (NE). Several studies exist in prior research that apply game theoretic solution concepts to the voting games. But the most recent and relevent work is that of Meir et al. [40]. Meir et al. [40] suggested the framework of voting as a dynamic process in which voters repeatedly change their reported preferences one at a time (if voters are allowed to change their preferences simultaneously, the process will never converge). This iterative process continues until either no further manipulations are available or else the system goes into a cycle. In the paper they study different versions of iterative voting, varying tie-breaking rules, weights and policies of voters, and the initial profile. Their results show that in order to guarantee convergence, it is necessary and sufficient that voters restrict their actions to natural best responses. They also showed that with weighted voters or when better replies are used, convergence is not guaranteed. Hence, myopic better response dynamics may cycle, even when start from a truthful voting profile, for both deterministic and randomized tie-breaking schemes. This topic of convergence to stable outcomes in strategic voting setting is interesting to artificial intelligence. It tackles the fundamental problem of decision making where agents are considered to be autonomous entities and they have to choose a joint plan of action or allocation of resources. A related dynamical model was considered by Airiau and Endriss [2], in which at each step, a voter is selected at random and may propose a single alternative to the one currently winning the election; a pairwise vote takes place between the current winner and the new alternative. As in [40] cycles may arise; the ability of the chosen voter to select a single alternative for a pairwise election indeed makes it possible to exhibit cycles which cannot be escaped. We expand this framework further, concerning the dynamics of weighted Plurality voting under sequences made up by various types of manipulations by the voters. We also consider other voting rules apart from Plurality. We use the idea of using potential function for studying the rate of convergence to equilibria in more general setting. For lexicographical tie-breaking scheme under different weight settings, we demonstarte that convergence to equilibria can be guaranteed considering different types of moves that leads to successful manipulation. Polynomial bounds are obtained and proofs are based on constructing a potential function with guaranteed value of increase/decrease at each step. Our results suggests different choices of potential functions can handle different versions of the problem. We also show that a cycle exist if we allow all types of moves, that's the reason we obtain bounds for different subsets of moves where the voting dynamics converges. Our results and the results obtained in [40] provide quite a complete knowledge of what combinations of types of manipulation move can result in cycles. We have observations regarding the compatibility of different types of manipulation moves and our convergence results are based on such observations. We identify combinations of types of moves that are able to lead to cycles of manipulation moves. We consider combinations of move types where convergence is guaranteed. We show that if one exclude certain deviations (if moves are incompatible), than an improvement path is guaranteed to terminate. Our results hold for an arbitrary initial point and in our settings, voters don't need to have complete information about other voters' preferences unlike some previous work. Our work helps to develop the analytical tools to charaterize situations in which one can expect to see a convergence. This study is a necessary first step to help to develop methods that could help design dynamics that would converge to equilibria.

1.3.2 Significance and importance of the problem

Manipulation of voting schemes has various negative consequences; not only do voters spend valuable computational resources, but worse, the outcome is less likely to be one that reflects the social good. However, we do not concern ourselves with the impact of manipulation on social welfare; despite the basic mainulability of all reasonable voting systems, it would still be desirable to find ways to reach a stable result, which no voter will be able to change. Considering manipulation a serious issue, we ask the question: in a system where manipulation may occur, when can we guarantee that the voters will end up satisfied with their (possibly manipulative) votes, in the context of the votes offered by the others? Meir et al. [40] have studied the dynamic process of making manipulations. Our work builds on the existing work, namely on the work of Meir et al. [40]. We try to shed light on the quantitative aspects of manipulative move sequences where convergence is guaranteed. We use potential functions and show how potential functions are useful for showing convergence in voting schemes. Our results, in conjuction with [40] provide quite a complete knowledge of what combinations of types of manipulation move can result in cycles and what combination of moves see the convergence. For our results, voters don't need to have complete information about the preferences of other voters and voters start from an arbitrary initial point.

The study of dynamics in strategic voting is interesting and very relevent to multi-agent systems, as it helps to understand, control and design multi-agent decision making processes. Our work helps to develop the analytical tools that are needed for this topic. Excluding certain deviations does not imply convergence to stable outcome but such results help to develop the tools and methods that could help desiging such processes.

1.3.3 Specific research questions

We work on the rate of convergence of different voting systems, specifically Plurality voting under myopic moves by voters. We ask does pure Nash equilibria exist for any set of voter types? and whether such an equilibrium can be reached via a sequence of myopic changes of vote, by the voters? We are interested in finding the number of steps of manipulation that may be needed to reach a pure Nash equilibrium.

The distinct questions we ask are: what type of manipulation moves lead to cycle?, what types of moves are compatible and converges to equilibria? and what is the rate of comvergence under different weight settings and under lexicographic tie-breaking rule? We focus on types of manipulations where there are no cycles and where convergence is guaranteed, and we analyze bounds on the number of steps required under different weight settings (given in chapter 2).

1.4 Structure of thesis

The rest of the thesis is organised as follow. The notations, assumaptions and some basic definitions are given in Chapter 2. Definitions of different types of manipulative moves are given along with different weight settings. Chapter 3 is about tactical voting dynamics, some results of convergence for Plurality and other positional scoring rules are described. Also different potential functions and a general definition of potential fucntion is also stated. We study manipulative voting dynamics in Chapter 4 with examples of different moves. Also results for different weight settings under different set of moves. In Chapter 5, we give results for mixture of different moves and also a result when all types of moves are allowed. Chapter 6 shows the cycles for positional scoring rules, non positional socring rule like Copeland, Bucklin and also Plurality with runoff under lexicographical tie-breaking rule. Finally, the conclusions and suggestions for future research are discussed in chapter 7.

1. INTRODUCTION

2

Preliminaries

In this chapter we give a general description of the model. The goal is to introduce and discuss preliminaries, notations and definitions. Section 2.1 introduces notations and some basic assumptions along with generally accepted symbols. We cover the preliminaries and introduce the necessary notation in this section. Certain key definitions of the terminologies used throughout the thesis are discussed in Section 2.2. We also introduced potential function with examples.

2.1 Notation and Assumptions

Let us denote the set of alternatives $\mathcal{A}$, where $|\mathcal{A}| = m$, a set of n voters $\mathcal{V} = \{1, 2, \ldots, n\}$ and a social choice rule f. Let $\mathcal{L}$ be the set of all linear orders on $\mathcal{A}$. Suppose candidates are competing under the *Plurality rule.* Plurality is the voting rule most often used in real-world elections, the important point is, it completely disregards all the information provided by the voter preferences except for the top ranking. We assume that voters have strong preference orderings over these candidates.

In an election, n voters express their preferences over a set of m alternatives. To be precise, each voter is assumed to reveal linear preferences: a ranking of the alternatives. The outcome of the election is determined according to a voting rule. Preferences of each voter i are represented by a linear order R_i on $\mathcal{A}$ and the sequence $R = (R_1, \ldots, R_n) \in \mathcal{L}^n$ is called a preference profile. Thus, a profile associates with each voter a preference ordering of the candidates, each of length m (number of candidates). Voters' preferences over alternatives ($\mathcal{A}$) are the important primitives. Two voters i and j are of the *same type* if they have identical preferences, i.e. $R_i = R_j$. The type of voter i is denoted as $\langle i \rangle$. It is identified with R_i. Voters of the same type are also called *like minded.* Like minded voters form a bloc. A preference profile is a distribution of voters over all possible preference rankings, so voters can be classified into $m!$ mutually exclusive voting blocs, B_1 to $B_{m!}$, according to their preference rankings over m candidates. The number of voters in bloc B_i is denoted as N_i.

Let L denote a single linear order such that $L \in \mathcal{L}$. If $V \subseteq \mathcal{V}$, then $R_{-V}(L)$ is the profile obtained from R when all voters from V vote L and all other voters retain their original linear orders. For $V \subseteq \mathcal{V}$ we will write $a \succ_V b$ if all voters from V strictly prefer a to b. Under the social choice rule, the notation $a \succ_i^S b$ is used to denote that voter i prefers candidate a to b in state S. A system has "true preferences" (fixed) for each voter $i \in \mathcal{V}$. The true preference of voter i over candidates $\mathcal{A}$ is denoted as $\succ_i$.

Voters also have "declared preferences" (can change) associated with a state of the system. A state allocates a declared preference profile to each voter. States can change over a sequence of time steps. For each state S voter $i \in \mathcal{V}$ has declared preferences denoted as $\succ_i^S$. Another notation $a \succ^S b$ represents that candidate a gets more votes than b, w.r.t. declared preferences in S. There is a possibility that voters announce different preferences than that of their true preferences (strategic voting).

Let S be a typical state and $\mathcal{S}$ is the domain of all allowable states. A social choice function determines for each possible profile (set of preference lists) of the voters the winner or set of winners of an election, where a social welfare function determines a social preference list, a single list that ranks the alternatives from first to last. A social choice function maps preference profile to a non empty set s of $\mathcal{L}$ and can be defined as $f : \mathcal{L}^{\mathcal{V}} \rightarrow s$ where $s \subseteq \mathcal{A}$ and a social welfare function is a mapping $f : \mathcal{L}^{\mathcal{V}} \rightarrow L$ where L denotes a single linear order. At state S, each voter $i \in \mathcal{V}$ is assumed to have a strict preference relation $\succ_i^S$ over the set $\mathcal{A}$. A state is the specification of declared preferences of each voter, $A \succ_S B$ means A ranks above B in state S w.r.t aggregated ranking (for a given Social welfare function). We denote by $\succ^S = (\succ_1^S, \succ_2^S, \ldots, \succ_n^S)$ the profile of individual preferences of voters at state S, for all $i \in \mathcal{V}$ and for all $S \in \mathcal{S}$ [56]. A voting rule is a function $f : \mathcal{L}^{\mathcal{V}} \rightarrow \mathcal{A}$, that maps preference profiles to winning alternatives, as state S is the specification of declared preferences of each voter. From the declared preferences of voters at a state S, we obtain winning and losing candidates according to the voting scheme used. We represent each state in the form of lexicographical order of numbers or we can say state S has an associated vector $N_1(S), \ldots, N_m(S)$. Let under Plurality rule

$$N_1(S), \ldots, N_m(S)$$

be the support of candidates at state S, sorted in decreasing order such that $N_1(S)$ denotes the number of votes for the candidate that receives highest support at state S. Under weighted votes setting, we define a state as a lexicographical order of numbers in descending order $N_j(S) = \sum_{i \in \mathcal{V}_j} w_i$ where voters in $\mathcal{V}_j$ support candidate j. $N_j(S)$ is the number that represents the total weight associated with candidate j at state S and w_i represents the weight of voter i. We use different potential functions for different versions of the problem and describe their notations where needed. An important property of the voting rules is that they may produce multiple winners i.e, they map a preference profile R to a non empty set s of $\mathcal{A}$. Tie-breaking rules are used to find a unique winner of an election from a subset of winners. A simple tie-breaking rule T does not depends on R and the value of $T(R, s)$ is uniquely determined by s, where $s \subseteq \mathcal{A}$. In such rules ties are broken according to an arbitrary fixed order over the candidates, so a manipulator cannot change the set of tied candidates, although he can change the election outcome by making different moves.

2.2 Definitions

Some important definitions are:

Definition 1 *(Election) Let $\mathcal{A}$ be a set of m alternatives and let $\mathcal{L}$ be the set of all linear orders on $\mathcal{A}$. Let $\mathcal{V}$ be a set of n voters and each voter $i \in \mathcal{V}$ has a fixed true preference list (which we denote as $\succ_i$), and a declared preference list which he announces and can change it which we denote as $\succ_i^S$.*

Definition 2 *(Voting Rule) A voting rule is a function $f : \mathcal{L}^{\mathcal{V}} \to \mathcal{A}$, that maps preference profiles to winning alternatives.*

Definition 3 *(Plurality) Also known as the simple majority rule. Each voter casts a vote for one candidate. The social choice is the candidate with the most votes. More* *generally, for weighted votes C, where we have $w : \mathcal{V} \to \mathbb{R}^+$, the winner is the candidate with greatest total weight of voters putting C first.*

Definition 4 *(Preference order) Each voter $i \in \mathcal{V}$ is associated with a linear order R_i over the candidates $\mathcal{A}$; this ordering is called voter's i preference order.*

Definition 5 *(Declared preference) A declared preference is the vote that a voter submits to the social choice function in use.*

Definition 6 *(State under any scoring social choice rule other than* Plurality*) State S of a system associates with each voter i, a declared rank ordering of candidates by which voter i is presumed to rate candidates and can be defined as $f : \mathcal{V} \to \mathcal{L}$.*

Definition 7 *(State under* Plurality *Rule) In the case of* Plurality *rule where the declared Preference list of a voter is just a single candidate, State S of the system is a function $f : \mathcal{V} \to \mathcal{A}$.*

Definition 8 *Assume we are using* Plurality. *Fix a state of the system. A* bloc *is a (maximum sized) set of voters who all support the same candidate w.r.t. declared preferences. However, voters belonging to the same bloc may or may not be* like minded.

Definition 9 *Fix a state S of the system. A* winner *$w(S)$ of a state is a candidate* or *a set of candidates over $\mathcal{A}$ who is chosen by the SCR, applied to the declared preferences of voters.*

Definition 10 Termination *of the process occurs, when no further* transition *is possible.*

Definition 11 *(Transition or change of state in case of individual voter migration under Plurality rule for tactical voting). Fix a state S of the system in which voter $i \in \mathcal{V}$ currently supports candidate $j \in \mathcal{A}$. The system can make a transition from current state S to a new state S', if voter i can switch to another candidate $j' \in \mathcal{A}$, and $w(S') \succ_i w(S)$ (that is, voter i prefers the winner in S' to the winner in S).*

Definition 12 *(Transition in case of Group migration under Plurality rule for tactical voting). Fix a state S of the system in which a set of like-minded voters $V \in \mathcal{V}$ currently support candidate $j \in \mathcal{A}$. The system can make a transition from current state S to S', if for set of voters V there is a candidate j' such that $j' \succ_V^S j$ and $\mathcal{A}'$ is the subset of $\mathcal{A}$ such that $\mathcal{A}' \succ_V w(S)$, $j' \succ_S \mathcal{A}'$.*

Definition 13 *(Transition for manipulation dynamics) A transition is a manipulation move (change of declared preference) by a single voter that changes the election's outcome to one he prefers. Voters make transitions according to their true preferences.*

Definition 14 *(Potential Function) Given a process involving a finite set $\mathcal{S}$ of states, a potential function*

$$\Phi : \mathcal{S} \to \mathbb{R}$$

should have the property that any allowable transition from state S to new state S' should always increase the value of Φ. (One could alternatively require the value of Φ to always decrease.) If it's possible for Φ to only take a finite number of distinct values, this will show that the process of making transitions must terminate.

We describe the voting rules considered in this thesis. All these rules assign scores to canddiates; the winners are the candidates with the highest scores.

Definition 15 *A positional scoring rule let $\vec{a} = \langle \alpha_1, \ldots, \alpha_m \rangle$ is a vector of integers such that $\alpha_1 \geq \alpha_2 \ldots \geq \alpha_m$. For each voter, a candidate receives α_1 points if it ranked first by the voter, α_2 if it is ranked second etc. The score of the candidate is the total number of points the candidate receives.*

Definition 16 *(Borda rule) Under the voting procedure proposed by Jean-Charles de Borda, each voter submits a complete ranking of all m candidates. For each voter that places a candidate first, that candidate receives $m - 1$ points, for each voter that places her second she receives $m - 2$ points, and so forth. The Borda count is the sum of all the points. The candidates with the highest Borda count win.*

Definition 17 *(Veto rule) Also known as anti-Plurality rule. A point is given to everyone except the least preferred candidate. The scoring vector for Veto rule is* $\langle 1, \ldots, 1, 0 \rangle$.

Definition 18 *(k-Approval voting rule) In k-approval voting rule a point is given to the most preferred k candidates (or points are given to all except the least preferred k candidates). The scoring vector for k-approval voting rule is* $\langle 1^k, 0^{m-k} \rangle$.

Scoring rules are a broad and concisely-representable classes of voting rules; scoring rules award points to alternatives according to their position in the preferences of the voters. Under this unified framework, we can express certain specific rules as:

- Plurality: $\vec{a} = \langle 1, 0, \ldots, 0 \rangle$.
- Borda: $\vec{a} = \langle m-1, m-2, \ldots, 0 \rangle$.
- Veto: $\vec{a} = \langle 1, \ldots, 1, 0 \rangle$.

where $\vec{a}$ is a sequence of scores allocated by a voter to the candidates in descending order of preference. A good indication of the importance of scoring rules is given by the fact that they are exactly the family of voting rules that are anonymous (indifferent to the identities of the voters), neutral (indifferent to the identities of the alternatives), and consistent (an alternative that is elected by two separate sets of voters is elected overall) [52].

There are also voting systems that are not scoring rules like given below.

Definition 19 *(Copeland rule) Simulate a pairwise election for each pair of candidates in turn (in a pairwise election, a candidate wins if it is preferred over the other candidate by more than half of the voters). A candidate gets 1 point if it defeats an opponent, 0 points if it draws, and -1 points if it loses.*

Definition 20 *(Bucklin scheme) Bucklin is a ranked voting method that proceeds in rounds, one rank at a time, until a majority is reached. Initially, votes are counted for all candidates ranked in first place; if no candidate has a majority, votes are recounted with candidates in both first and second place. This continues until one candidate has a total number of votes that is more than half the number of voters.*

Definition 21 *(Plurality with Runoff) The Plurality with runoff voting rule selects a winner in two rounds. A first round eliminates all candidates except the two candidates who receive the highest scores using the Plurality rule. The second round determines the winner between these two where they compete in a pairwise election.*

Definition 22 *(Pairwise election) Candidate A beats candidate B in a pairwise election if a majority of the voters prefer A to B.*

Definition 23 *(Tactical vote in case of Individual voter migration under Ranked based rules) Fix a state S of the system in which voter $i \in \mathcal{V}$ has declared preference $\succ_S^i$. System can make a transition from current state S to a new state S', if voter i can switch to another candidate $j' \in \mathcal{A}$, if and only if $\mathcal{A}'$ is the subset of $\mathcal{A}$ such that $\mathcal{A}' \succ_i^S w(S)$, $j' \succ_S \mathcal{A}'$ then j' moved to the top in the declared ranking of voter i while all candidates other than $\mathcal{A}'$ move one position down.*

Definition 24 *(Tactical vote in case of Group migration under Ranked based rules) Fix a state S of the system in which a set of like-minded voters $V \subseteq \mathcal{V}$ has declared preference $\succ_S^V$. System can make a transition from current state S to S', if for set of voters V there is a candidate j', if and only if for all $\mathcal{A}' \subseteq \mathcal{A}$ such that $\mathcal{A}' \succ_V w(S)$, $j' \succ_S \mathcal{A}'$, then j' moved to the top in the declared ranking of V voters while all candidates apart from $\mathcal{A}'$ move one position down.*

Definition 25 *(Best response) A best response is the change of voters declared preference in favour of his most preferred candidate capable of winning.*

Definition 26 *(Tie-breaking rule) A tie-breaking rule T for an election $(\mathcal{A}, \mathcal{V})$ is a mapping $T = T(R, s)$ such that for any $s \subseteq \mathcal{A}$, $s \neq \emptyset$, outputs a candidate $c \in s$.*

Definition 27 *(Lexicographic tie-breaking) Ties are broken using a priority ordering on the candidates, if there is a set of tied alternatives, it selects a candidate who is first in the sequence as a winner according to a fixed priority ordering.*

2.2.1 Manipulations

The typical form of manipulation is, in which voters misrepresent their preference orderings over the alternatives and she may benefit from misrepresenting her preferences. One can consider a manipulation successful if it causes some candidate to win that is preferred by each one of the manipulators to the candidate who would win if the manipulators voted truthfully. There is no reason to prefer one preference list over another if outcomes of elections are the same. Essentially all voting rules are manipulable, i.e., a voter may benefit from misrepresenting her preferences over the alternatives [33, 58]. We are concerned with the convergence to stable outcomes in strategic voting settings in plurailty voting games. We restrict our attention to the Plurality rule, unless explicitly stated otherwise.

2.2.1.1 Types of moves

A move is the switching of a voter from one candidate to another in order to make a manipulative vote. We consider bounds on the length of sequences of manipulation under Plurality where each manipulation leads to a new winner and each voter has a weight which is a positive number and is fixed throughout. The score of a candidate i is the sum of weights of voters that voted for candidate i. If voters are unweighted then the score of a candidate is the number of votes of that candidate. Using Plurality rule a voter's declared preference may be expressed as a single candidate, and it is not necessary to identify a ranking (but a voter's true preference is still a ranking). There are various different types of moves that a voter can perform to make a manipulation. The following classification of moves is defined for Plurality rule.

1. Loser to new winner: A move from candidate C to C', where neither was winner beforehand, and C' is winner after the move.

2. Loser to existing winner: A move from candidate C to the existing winner C' to improve the score of C'.

3. Winner to loser: A move from a winning candidate C to C' to make C'' a new winner where C'' is different from C and C'.

4. Winner to winner: A move from a winning candidate C to a new winning candidate C' because the manipulator prefers C' over C.

(a) Winner to larger winner: A move from a winning candidate C to another candidate C' such that C' is winner after move with total score more than previous score of C.

(b) Winner to smaller winner: A move from a winning candidate C to another winning candidate C' such that the total score of C' is less than C.

(c) Winner to new winner of the same size: A move from a winning candidate C to another winning candidate C' such that the total score of C' is equal to the score of C but according to tie-breaking rule $C' \succ C$.

[40] consider the possible steps of type 1, 3 and 4 moves under the Plurality rule for un-weighted voters. Moves of type 2 do not change the winning candidate. So, type 2 moves arguably need not be considered in a game-theoretic setting, although ideally we would obtain bounds that allow type 2 moves to take place. Type 3 is arguably unnatural since (for Plurality), a type 1 move $C \longrightarrow C''$ would have the same effect, and be more natural.

2.2.1.2 Types of manipulations

In manipulation dynamics, voters change their mind to make a "manipulative vote" that changes the current result of the election. We assume that some tie-breaking rule applies if 2 candidates receive the same level of support. We assume that tie-breaking rule is lexicographic i.e., given a set of tied alternatives, it selects one that is first in order with respect to a fixed ordering.

The first type of manipulation, a voter migrates to a new winner with increased support than the previous winner. Type 1 and 4a moves can take place in this type of manipulative dynamics. We have potential functions that work for this type of manipulation dynamics. It is known from [21, 30] that given a potential function the process of repeatedly making self-improving moves must terminate at a Nash equilibrium. In the weighted-voter setting with manipulative dynamics, there can be a second type, where a voter migrates to a new winner with decreased support than the previous winner (if, in the previous state, the voter supports the winner, but then changes to a new candidate who becomes the winner), and a move is only allowed when the winner changes in the next state. Only 4b type of moves are possible in this type of dynamics. A third type of manipulation in the weighted-voter setting is when a voter make a manipulative vote that increases the support of the winning candidate but may not always change the winner. 1, 2 and 4a types of moves can take place in this type of manipulative dynamics. Fourth type of manipulation is, a voter migrates to a new winning candidate with either increased support than the previous winner or decreased support. The only restriction is that a winner changes. There are various different types of moves that a voter can perform to make a manipulative vote of this type like type 1 (Loser to new winner), type 4a (Winner to larger winner) and type 4b move (i.e, Winner to smaller winner).

If moves like that are allowed, two important questions are *does the process of making such manipulations terminate?* and *how long may this sequence of manipulations be?*. It looks like the process terminates (it would be interesting to prove that it does terminate and the maximum number of steps required to terminate this process). We are asking this question in the context of elections and also the question that how long this sequence must be. Bounds on the possible number of steps required to terminate the process in terms of weights for first, second and third type of manipulation is $\sum_{i \in V} w(i) = W$ where W is the total weight and weights are integers. However, we are interested in bounds on the number of steps that are purely in terms of m (number of candidates) and n (number of voters) and independent of the total size of the weight or values of weights which can be quite large. An initial observation is the number of states (using Plurality) is at most m^n. It is interesting to find a bound that is polynomial in terms of m and n (and independent of the total size of the weight which can be quite large). More specifically, we are interested in the number of steps to be made by the system to achieve the Nash equilibrium.

Observation 1 *Third type of manipulation where moves of type 1, 2 and 4a are allowed, all these moves increase the score of the winner. Hence, the score of the winning candidate may be used as a potential function to show termination for these types of manipulation move.*

Most of our results are for sequences of moves of types 1, 2 and 4a, because convergence to an equilibrium can be guaranteed for these moves. This is an easy observation, as in this case the score of the winner can be viewed as a natural potential function which monotonically increases along the improvement path. While in general the rate of convergence is exponential, polynomial bounds are obtained for the case of bounded weights, either integer or real. The proofs are based on constructing a potential function with a guaranteed value of increase at each step. We shall see however that in some situations one can design "smarter" potential functions that are more useful for showing a faster convergence rate. Since there are n weighted voters, all possible ways in which n weights can combine is 2^n so we can say there are 2^n possible values for a voter and there are m different candidates so we have an initial observation that the number of transitions (using Plurality) in general weight setting are at most 2^{mn} if there are no cycles. Since the bound is exponential in both m and n, we are trying to obtain a bound that is a slower-growing function than 2^{mn}.

The following example illustrate how voters can change their votes in response to each other.

Example 1 *The Chairman's paradox: Suppose there are voters $\mathcal{V} = \{1,2,3\}$, alternatives $\mathcal{A} = \{A,B,C\}$. Suppose that voter 1 has preferences $A \succ_1 B \succ_1 C$, voter 2 has preferences $B \succ_2 C \succ_2 A$ and voter 3 has preferences $C \succ_3 A \succ_3 B$ (a Condorcet cycle). Suppose further that in the event that the voters vote for distinct candidates, then the choice of voter 1 (the "chairman") is the winner. This rule of breaking ties in favor of voter 1 can be implemented with voter weights: let voter 1 have weight $\frac{3}{2}$ while voters 2 and 3 have weight 1. If initially the voters support their favorite candidates, then voter 2 has an incentive to deviate, and he migrates to voting for C. Afterwards, no further migrations are possible. The chairman's least favorite candidate is chosen.* [1]

Suppose instead that initially voter 1 votes for B, and voters 2 and 3 vote for C. Then voter 2 can migrate to B (type 4a move), after that, voter 1 migrates to A (type 4b move), at which point the voters are supporting their preferred candidates. So, voter 2 returns to C (suggesting that voter 1's myopic move to A was a tactical blunder).

[1] The paradox [26] is the stronger result than under the *impartial culture* assumption, where preference lists are chosen at random, the chairman actually does worse *on average* than the other voters!

2.2.1.3 Weights settings

For both tactical and manipulative voting dynamics, we not only consider equi-weighted but also weighted voting system. A weighted voting system is one in which the preferences of some voters carry more weight than the preferences of other voters. A voter's weight may represent a group of voters coordinates their actions in order to affect the election outcome. Manipulation by a single voter presents a grave concern from a theoretical perspective, in real-life elections this issue does not usually play a significant role, typically the outcome of a popular vote is not close enough to be influenced by a single voter. Indeed, a more significant problem is that of coalitional manipulation, where a group of voters coordinates their actions in order to affect the election outcome. While many human elections are unweighted, the introduction of weights generalizes the useability of voting schemes, and can be particularly important in multiagent systems settings with very heterogenous agents [11]. Each voter has an associated weight in form a positive number and it is fixed throughout. Our results have dependence on the voters' weights. It is interesting to consider manipulation dynamics with weighted voters because even a single weighted voter can make a "manipulative vote" and can change an election's outcome while an unweighted vote can hardly change an election's outcome. Weighted votes raise new questions. It requires us to carefully design potential function. That is the reason manipulative voting is less interesting when votes are unweighted. Weighted votes will also help in tie breaking. In weighted voters setting, we assign a weight w_i (integer or real value) to each voter $i \in \mathcal{V}$, so not all voters are equally important unlike when voters are unweighted i.e, $w = (1, \dots, 1)$. We assume each voter $i \in \mathcal{V}$ has a fixed weight. To compute the winner on a profile $(R_1, \dots, R_n)$ under a voting rule f given voters' weights $w = (w_1, \dots, w_n)$, we apply f on a modified profile such that for each $i = 1, \dots, n$ contains w_i copies of R_i. We have results for 3 different weight settings.

1. General weight setting:
   A weight function is a mapping $w : \mathcal{V} \rightarrow \mathbb{R}^+$. For this type of setting we have bounds in terms of m and n.

2. Bounded real weight setting:
   Weights are positive real numbers. All n voters have weights in the range $[1, w_{max}]$. For this setting we seek bounds in terms of w_{max} as well as m and n.

3. Bounded integer weight setting:
   Voters' weights are positive integers and lie in the range $\{1, 2, \ldots, w_{max}\}$. In this setting, weight function is a mapping $w : \mathcal{V} \to \mathbb{N}$. We seek bounds in terms of w_{max}, m and n.

An additional parameter K can also be added to all 3 settings of weights where $K < n$ is the number of distinct weights. The total weight of voters are:

$$\sum_{i \in \mathcal{V}} w(i) = W$$

where W is the total weight. We can say that $|\mathcal{V}| \leq W$.
Since here we are considering Plurality rule so the declared preference list of a voter is single candidate as Plurality rule is the positional scoring rule with scoring vector $\vec{a} = \langle 1, 0, \ldots, 0 \rangle$. The total weight of voters who favoured a specific candidate at a particular state S can be obtained as:

$$N_j(S) = \sum_{i \in \mathcal{V}} w_i$$

Here, $N_j(S)$ is the number that represents the total weight of voters who selected candidate j at state S and w_i represents the weight of voter i. For positional scoring rules (apart from Plurality), values of candidates are derived from the declared preferences of the weighted votes at a given state (say S) as given below:

$$N_j(S) = \sum_{i \in \mathcal{V}} s_i . w_i$$

$N_j(S)$ is the number that represents the total value associated with candidate j at state S, where s_i denotes the score of a candidate j in the declared preference list of voter i at state S according to the scoring rule used and w_i represents the weight of voter i.

2.2.2 Existence of Potential functions and Pure Nash Equilibria

The potential function method has emerged as a general and key technique in understanding the convergence to equilibria. The potential function method is used to find the existence of pure Nash equilibria, convergence of best response dynamics and the price of stability. The notion of potential function was first introduced for general game classes by [43]. Rosenthal [56] use a potential function to prove the existence of pure strategy Nash equilibria in congestion games. Potential functions are valuable for proving the existence of pure Nash equilibria, so we can say that potential functions are clearly relevant to equilibria. Even-Dar et al. [21] use a potential function to measure closeness to a balanced allocation, and use it to show convergence for sequences of randomly-selected "best response" moves in a more general setting in which tasks may have variable weights and resources may have variable capacities. We are interested in the rate of convergence and in principle the idea of using potential functions for studying the rate of convergence to equilibria is a natural one. The goal is to determine the number of steps required to reach Nash equilibrium.

Given a process involving a finite number of states, a potential can be defined as

$$\Phi : \mathcal{S} \rightarrow \mathbb{R}^+$$

where $\mathcal{S}$ is a set of states. Transitions are self-improving moves $S \longrightarrow S'$ where S and S' are states; $\Phi(S) < \Phi(S')$ for all such moves means Φ is potential function. If transitions always cause Φ to increase. Then the process must terminate, and a simple bound on the number of steps is the number of alternative values Φ can take. Or you could require $\Phi(S) > \Phi(S')$ always.

Examples below show the potential functions used for the rate of convergence to equilibria. In a Bin packing problem, objects of different volumes must be packed into a finite number of bins of capacity C in a way that minimizes the number of bins used. In the [7] (where a classical Minimum Bin packing problem is discussed with the constraint that the items to be packed are handled by selfish agents, and all the bins have the same fixed cost and the cost of a bin is shared among all the items it contains according to the normalized fraction of the bin they use) a suitable potential function is used for the convergence of the Bin packing game to a pure Nash equilibrium and which proves to be useful in the case in which all the "heights" of items are rational numbers. In [7] height is used to refer to the size/weight of an item and the sum of the heights of the items packed in to a particular bin such as the j-th bin (say B_j) is denoted as H_j. In order to bound the convergence time the potential function defined is:

$$\Phi(t) = 2^{\sum_{i=1}^{k(t)} H_i^2}$$

As the item perform an improving step while migrating from one bin to another bin, the value of potential function increases by a multiplicative factor and will reach its maximum at some point when the potential function reaches its upper bound.

Another useful potential function can be

$$\Phi(t) = \sum_{i=1}^{k(t)} H_i^2$$

Just like the potential function of Bin Packing game the above potential function (the sum of the squares of heights) is also a valid one as it is the non exponential version of the previous potential function. This potential function also increases at each step by a constant factor of at least $2a$ (a denotes the height of item) when the item migrates (in order to minimize its cost) to a bin in which it fits better with respect to the unused space. The potential function helps approximate the sequence of steps.

The concept of Nash equilibria has become an important mathematical tool in analyzing the behavior of selfish users in non-cooperative systems [50]. One way to nashify an assignment is to perform a sequence of greedy selfish steps. A greedy selfish step is a user's change of its current pure strategy to its best pure strategy with respect to the current strategies of all other users. Any sequence of greedy selfish steps leads to a pure Nash equilibrium. However, the length of such a sequence may be exponential in n [28]. It has already been proved that a sequence of self-improvement moves converges to a Nash Equilibrium [21, 30] but is recently studied in the context of voting by [40]. Since voters are considered rational agents, who want to maximize their own utility, their best strategy may be to manipulate an election if this will gain them a higher utility.

2.3 Summary

We gave a classification of different types of moves a voter can make, some basic terms used were defined and also a description of the different weight settings we consider for our results was given. We introduced the potential function with examples from previous work.

3

Tactical voting dynamics

The chapter is about the introduction of Nash equilibria and potential functions. We also analyzed in this chapter the sequences of votes that may result from various voters performing tactical vote in unweighted setting and also weighted setting. We conclude that the process of making tactical vote terminates and we find the length of sequence of making tactical vote for positional scoring rules. In Section 3.1, we described the tactical voting and results for the termination of making tactical vote in case of positional scoring rules. Tactical voting is also analyzed under real weight setting in Section 3.2. Section 3.3 concludes the chapter.

3.1 Tactical voting

The model is a system of states and transitions. Voters have "true preferences" (fixed), and "declared preferences" which can change. Each voter's individual preference is summarized in the concept of a state. A transition occurs from current state to a new state when a voter changes his mind and chooses a different candidate to support (under Plurality). In a state of a system, each voter determine whether it can improve the outcome by altering its own vote while assuming that all other votes remain the same. This model is different than that of manipulation dynamics because it simply raises the votes of an expected leading candidate she most prefers. A voter can change his mind (choose a different candidate to support) according to the following mind changing rule: Voters consider current state, a state is being a description of how all blocs vote and the outcome implied by that voting. Now each bloc/a single voter determine whether it can improve the outcome by altering its own vote while assuming that all other votes remain the same. In a state of a system, consider all alternative candidates that a voter ranked higher than the current winner of the state. Voter/bloc of voters can then change his support to that alternative candidate who has currently most votes, breaking ties in favor of his own preference. With this mind changing, transition occurs and system enters into a new state from the current state. If no bloc can improve the outcome, the current situation is a Nash equilibrium. It turns out that no more than $m-2$ blocs can improve the outcomes. At each iteration, "state of the system" associates each voter with a candidate currently supported by that voter.

In three candidate elections under *Plurality rule*, each voter has two strategies only: voting for either her first or second preferred candidate. Under *Plurality rule* voting for one least prefered alternative is dominated by the strategy of voting for one's most prefered alternative, so no voter will ever vote for his least preferred alternative.

We consider two kinds of tactical voting dynamics

- Individual voter migration (Definition 11 from Chapter 2)
- Group migration or Coalitional migration (Definition 12 from Chapter 2) A coalition is a set of self-interested agents that agree to cooperate to execute a task or achieve a goal. Such coalitions were thoroughly investigated within game theory. In our model Coalitional migration means, a group of voters can change their support to another candidate simultaneously, according to the rules of tactical voting. Coalition members may coordinate their votes. A winning coalition can force the outcome of the social choice function.

An example of Group migration:
Suppose there are 3 candidates a, b and c such that $a \succ b \succ c$ at state S, where $w(S)$ denotes the winner of state S.

$$w(S) = a$$

A set of like-minded voters $V \in \mathcal{V}$ currently support candidate c. A subset $V_1 \subseteq V$ is such that $b \succ_{V_1}^{S} c$, so V_1 voters switch their support from candidate c to b and a transition occurs from state S to S', where now $w(S') = b$ or a or $\{a, b\}$. The bloc of voters supporting candidate b increased in new state S' and that of c decreased by the same number with which the bloc of b has increased.

3.1.1 Process termination for Plurality rule

For our tactical model, we consider a system comprised of a finite number of states and transitions occurs from state to state when voters change their mind and support an alternative candidate. In the case of *Plurality* rule where a declared preference list of a voter is just a single candidate, state S is a function $f : \mathcal{V} \rightarrow \mathcal{A}$ and a *bloc* is a (maximum sized) set of voters who all support the same candidate.

Let us fix the set of alternatives $\mathcal{A}$, a set of voters $\mathcal{V} = \{1, 2, \ldots, n\}$ and voters have strong preference ordering over these candidates. The system has true preferences (fixed) for each voter $i \in \mathcal{V}$ denoted as $\succ_i$ and declared preferences of voter i that are represented as $\succ_i^S$. From the declared preferences of voters at a state S, we obtain scores of candidates according to the voting scheme used. A state is represented in the form of lexicographical order of numbers. Let

$$N_1(S), \ldots, N_m(S)$$

be the bloc sizes of candidates at state S, sorted in decreasing order such that $N_1(S)$ denotes the number of votes for the candidate that receives highest support at state S. At state S of the system, when a voter or a coalition of like-minded voters make a tactical vote and switch to another candidate then the system make a transition from state S to S' and according to the mind changing rule, votes from lower supported candidate are shifted to higher supported candidate. The potential function that we use to prove the termination of mind changes at state S is

$$\Phi(S) = N_2(S) + 2N_3(S) + \ldots + (m-1)N_m(S) \tag{3.1}$$

equivalently

$$\Phi(S) = \sum_{i=1}^{m-1} i(N_{i+1}(S))$$

where Φ denotes the potential of state S, $N_i(S)$ denotes the bloc sizes of candidates in lexicographical order, where i represents the lexicograhical position of candidates at state S and $i = 1, 2, \ldots, m$.

Lemma 1 *Φ as defined in Equation 3.1 is a potential function under restricted kind of tatical votes.*

Proof. If we consider that there are finitely many states in the system that allows transitions between states, then a potential function Φ is a function that maps every state of the system to a real value and satisfies the following condition: If the current state of the system is S, and voters $V \in \mathcal{V}$ (where V may be a single voter or a set of like minded voters) switch from candidate i to candidate j as $j \succ_V^S i$ and system migrates from current state S to a new state S', then the number of voters V (who change their mind at state S) is the least number with which the value of the potential function decreases as proved below.

Case 1: Bloc sizes preserve the same lexicographical order Here we consider the case where voters change their mind and switch to another candidate, as a result of this migration of votes, the system make a transition from the current state to a new state and candidates remain in the same sorted order in the new state as they were in the previous state. We can say that mind changing does not make a candidate more popular.

Let S be the current state of system and the bloc sizes at state S are

$$N_1(S), \dots, N_x(S), \dots, N_y(S), \dots, N_m(S) \tag{3.2}$$

and the potential at state S is

$$\Phi(S) = N_2(S) + 2N_3(S) + \dots (x-1)N_x(S) + \dots + (y-1)N_y(S) + \dots + (m-1)N_m(S) \tag{3.3}$$

Let $N_x(S)$ and $N_y(S)$ represent the bloc sizes supporting candidate j and i respectively at state S. Let $V \subseteq B_i(S)$ (where $B_i(S)$ is the bloc of voters supporting candidate i at state S) be the set of voters who change their support from candidate i to candidate j as $j \succ^S i$ and $N_x(S) > N_y(S)$, such that $N_{x-1}(S') > N_x(S') > N_{x+1}(S') > \dots > N_y(S') > N_{y+1}(S')$, in other words, bloc sizes correspond to the same ordering of the candidates, as a result transition occurs from state S to S'. So the new state S' of system is

$$N_1(S'), \ldots, N_x(S'), \ldots, N_y(S'), \ldots, N_m(S') \tag{3.4}$$

where

$$N_x(S') = N_x(S) + |V|$$

and

$$N_y(S') = N_y(S) - |V|$$

The potential function at S' is

$$\Phi(S') = N_2(S') + 2N_3(S') + \ldots + (x-1)N_x(S') + \ldots + (y-1)N_y(S') + \ldots + (m-1)N_m(S')$$

Representing $\Phi(S')$ in form of $\Phi(S)$

$$\Phi(S') = N_2(S) + 2N_3(S) + \ldots + (x-1)(N_x(S) + |V|) + \ldots + (y-1)(N_y(S) - |V|) + \ldots + (m-1)N_m(S)$$

which shows that transition from S to S' affects only N_x and N_y, and the decrease in potential function is

$$\Phi(S) - \Phi(S') = ((x-1)N_x(S) + (y-1)N_y(S)) - ((x-1)N_x(S') + (y-1)N_y(S')) \tag{3.5}$$

By putting the values of $N_x(S') = N_x(S) + |V|$ and $N_y(S') = N_y(S) - |V|$ in Equation 3.5

$$= ((x-1)N_x(S) + (y-1)N_y(S)) - ((x-1)(N_x(S) + |V|) + (y-1)(N_y(S) - |V|))$$

$$= (x-1)N_x(S) + (y-1)N_y(S) - (x-1)(N_x(S) + |V|) - (y-1)(N_y(S) - |V|)$$
$$= (x-1)(N_x(S) - (N_x(S) + |V|)) + (y-1)(N_y(S) - (N_y(S) - |V|))$$
$$= |V|((y-1) - (x-1))$$
$$= |V|(y-x)$$

Since $y > x$, we have $|V|(y-x) > 0$

$$\Rightarrow \Phi(S) > \Phi(S')$$

Case 2: Mind changing of voters results in changing the popularity of the candidates: Suppose mind changing of a voter always increase the popularity of some candidate and let the current state of system is S and the bloc sizes at this state S are

$$N_1(S), \dots, N_x(S), \dots, N_y(S), \dots, N_m(S)$$

where x is the lexicographical position of candidate j at state S and y is that of candidate i at state S and $x < y$ and the potential at state S is

$$\Phi(S) = N_2(S) + 2N_3(S) + \dots (x-1)N_x(S) + \dots + (y-1)N_y(S) + \dots + (m-1)N_m(S)$$

where $N_x(S)$ and $N_y(S)$ represent the bloc sizes supporting candidate j and i respectively at state S.

Let $V \subseteq B_i(S)$ be the set of like minded voters who change their support from candidate i to candidate j as $j \succ_V^S i$ and $N_x(S) > N_y(S)$, such that the popularity of candidate j increases. Then $N_x(S)$ may shift towards left side and the $N_y(S)$ towards right side. New state S' is

$$N_1(S'), \dots, N_{x'}(S'), \dots, N_{y'}(S'), \dots, N_m(S')$$

where x' is the lexicographical position of candidate j at state S' as after migration of votes from candidate i to candidate j, the bloc size of candidate j may shift towards left, so after sorting bloc sizes at state S', x' is the new position of candidate j and y' is that of candidate i at state S' and $x' < y'$, $x' \leq x$, $y' \geq y$. However,

$$N_{x'}(S') = N_x(S) + |V| \tag{3.6}$$

$$N_{y'}(S') = N_y(S) - |V| \tag{3.7}$$

and, we have

$$N_{i+1}(S') = N_i(S)$$

where $x' \leq x$ and $x' < i < x$, and

$$N_{j-1}(S') = N_j(S)$$

where $y' \geq y$ and $y < j < y'$

The potential at S' is

$$\Phi(S') = N_2(S')+2N_3(S')+\ldots+(x'-1)N_{x'}(S')+\ldots+(y'-1)N_{y'}(S')+\ldots+(m-1)N_m(S')$$

Representing $\Phi(S')$ in form of $\Phi(S)$

$$\Phi(S') = N_2(S)+2N_3(S)+\ldots+(x'-1)(N_x(S)+|V|)+\ldots+(y'-1)(N_y(S)-|V|)+\ldots+(m-1)N_m(S)$$

which shows that transition from S to S' affects the bloc sizes between $N_{x'}$ and $N_{y'}$ both at state S and S' including $N_{x'}$ and $N_{y'}$. To compare the potential of both states first we consider that in the new state S' bloc size $N_x(S)+|V| = N_{x'}(S')$ move one position towards left i.e. $x'(S') = (x-1)(S')$ and as a result $N_{x'}(S) = N_x(S')$. In the same way $N_y(S) - |V| = N_{y'}(S')$ move one position towards right i.e. $y'(S') = (y+1)(S')$ and $N_{y'}(S) = N_y(S')$. Now we have state S as

$$N_1(S), \ldots, N_{x'}(S), N_x(S), \ldots, N_{y'}(S), N_y(S), \ldots, N_m(S)$$

and potential at state S is

$$\Phi(S) = N_2(S) + 2N_3(S) + \ldots + (x'-1)N_{x'}(S) + (x-1)N_x(S) + \ldots + (y-1)N_y(S) + (y'-1)N_{y'}(S) + \ldots + (m-1)N_m(S))$$

and after transition the potential of new state S' is: $\Phi(S') = N_2(S') + 2N_3(S') + \ldots + (x'-1)N_{x'}(S')+(x-1)N_x(S')+\ldots+(y-1)N_y(S')+(y'-1)N_{y'}(S')+\ldots+(m-1)N_m(S')$

Representing $\Phi(S')$ in form of $\Phi(S)$

$$\Phi(S') = N_2(S) + 2N_3(S) + \ldots + (x'-1)(N_x(S) + |V|) + (x-1)N_x(S') + \ldots + (y-1)N_y(S') + (y'-1)(N_y(S) - |V|) + \ldots + (m-1)N_m(S)$$

Let δ denote the decrease/change in potential in every two successive states.

$\delta = [N_2(S) + 2N_3(S) + \ldots + (x'-1)N_{x'}(S) + (x-1)N_x(S) + \ldots + (y-1)N_y(S) + (y'-1)N_{y'}(S) + \ldots + (m-1)N_m(S))] - [N_2(S) + 2N_3(S) + \ldots + (x'-1)(N_x(S) + |V|) + (x-1)N_x(S') + \ldots + (y-1)N_y(S') + (y'-1)(N_y(S) - |V|) + \ldots + (m-1)N_m(S)]$

Discarding the factors that remain uneffected:

$= (x'-x)(N_{x'}(S) - N_x(S)) + (y-y')(N_y(S) - N_y'(S)) + |V|(y'-x')$

Clearly, we have $x' < x < y < y'$

So, $(x'-x)(N_{x'}(S) - N_x(S)) < 0$ as $x' < x$ and here, we have $(x'-x) = -1$

and $(N_{x'}(S) - N_x(S)) < |V|$ So, $(x'-x)(N_{x'}(S) - N_x(S)) < -|V|$

Similarly,

$(y-y')(N_y(S) - N_y'(S)) < 0$ as $y < y'$ and $(y-y') = -1$

and $(N_y(S) - N_y'(S)) < |V|$ So, $(y-y')(N_y(S) - N_y'(S)) < -|V|$

Hence, $(x'-x)(N_{x'}(S) - N_x(S)) + (y-y')(N_y(S) < -2|V|$

or even if we suppose,

$(x'-x)(N_{x'}(S) - N_x(S)) + (y-y')(N_y(S) = -2|V|$

Now $|V|(y'-x') > -2|V|$ as we have $x' < x < y < y'$ or $y' > y > x > x'$, which shows that,

$$(y' - x') > 2$$

This proves, that

$$(x' - x)(N_{x'}(S) - N_x(S)) + (y - y')(N_y(S) - N'_y(S)) + |V|(y' - x') > 0$$

$$\Rightarrow \Phi(S) > \Phi(S')$$

Now considering a more general case when a candidate gains popularality as a result of migration of votes and a candidate who gains votes shifts towards left without any restrictions. The change in potential function to find the difference between $\Phi(S)$ and $\Phi(S')$ is,

$\delta = [N_2(S) + 2N_3(S) + \ldots (x-1)N_x(S) + \ldots + (y-1)N_y(S) + \ldots + (m-1)N_m(S)] - [N_2(S') + 2N_3(S') + \ldots + (x'-1)N_{x'}(S') + \ldots + (y'-1)N_{y'}(S') + \ldots + (m-1)N_m(S')]$

$= [(x' - 1)N_{x'}(S) + x'N_{x'+1}(S) + \ldots + (x-1)N_x(S) + (y-1)N_y(S) + \ldots + (y' - 1)N_{y'}(S)] - [(x'-1)N_{x'}(S') + x'N_{x'+1}(S') + \ldots + (y'-1)N_{y'}(S')]$

$= (-1)[N_{x'+1}(S') + (-1)N_{x'+2}(S') + \ldots + (-1)N_x(S')] + [N_y(S') + \ldots + N_{y'-1}(S')] + [(x-1)N_x(S) - (x'-1)N_{x'}(S')] + [(y-1)N_y(S) - (y'-1)N_{y'}(S')]$

From both previous cases we have seen that $\Phi(S) - \Phi(S') > 0$, So

$= (-1)[N_{x'+1}(S') + \ldots + N_x(S')] + [N_y(S') + \ldots + N_{y'-1}(S')] + |V|(y' - x') + (x - x')N_x(S) + (y - y')N_y(S) > 0$

So like the previous case

$$\Phi(S) - \Phi(S') > 0$$

$$\Rightarrow \Phi(S) > \Phi(S')$$

The value of δ (the decrease in potential in every two successive states) is

$\delta = (-1)[N_{x'+1}(S') + \ldots + N_x(S')] + [N_y(S') + \ldots + N_{y'-1}(S')] + |V|(y' - x') + (x - x')N_x(S) + (y - y')N_y(S)$

Or in form of state S, above equation can be written as

$\delta = (-1)[N_{x'}(S) + \ldots + N_{x-1}(S)] + [N_{y+1}(S) + \ldots + N_{y'}(S)] + |V|(y' - x') + (x - x')N_x(S) + (y - y')N_y(S)$

where $|V|$ denotes the number of like minded voters who change their support from candidate i to j and x and y represent the lexicographical position of the bloc sizes supporting candidate j and i respectively at state S. While x' and y' denotes the new lexicographical position of the bloc sizes supporting candidate j and i respectively at state S', when $|V|$ voters change their mind and as a result the bloc size of candidate j shift towards left and that of candidate i towards right.

We can say that δ is the absolute constant by which the potential function is decreased in every iteration. Decreasing the potential value by at least δ in every iteration ensures the termination of the process. Hence if improving moves of voters at each new state decreases the value of the potential function, then a move by voter $V \in \mathcal{V}$ that results in a new state S', can leads to

$$\Phi(S) > \Phi(S')$$

where $S, S' \in \mathcal{S}$.

As votes migrate from lower bloc sizes towards higher bloc sizes, as a result transitions move from states to states having lower potential follows that the process will terminate. This ensures that every move of the dynamics decreases the potential function by a factor δ. The value of potential function Φ reduces by at least an absolute constant in every iteration. As the votes move from lower bloc sizes towards the higher bloc sizes, the size of the right side blocs reduces at each new state until the size becomes zero. Similarly the size of the left side blocs increases by the same factor until the size of the left most bloc becomes n. The value of δ in case 1 is

$$\delta = |V|(y - x) \tag{3.8}$$

If the potential function decreases in Case 1 (where the bloc sizes corresponds to the same ordering of the candidates), this ensures that the decrease in potential function is greater in Case 2 (where mind changing increase the popularity of some candidate) than in Case 1. The above establishes that Φ is a potential function. □

Theorem 1 *Under the tactical voting dynamics, the process always terminates in at most mn steps.*

Proof. Let n be the number of voters and m the number of candidates and S denotes the current state of the system represented in the form of lexicographical order of numbers as $N_1(S), \ldots, N_m(S)$. Now consider with transition from state S to S', the new state of the system is such that $N_1(S'), \ldots, N_m(S')$, which is obtained from state S by moving votes from the lower bloc sizes of the sequence towards the larger bloc sizes in such a way that candidate j receives votes from candidate i where $j \succ^S i$, and this migration of votes results in decrease of potential function as per Lemma 1. Also Lemma 1 shows that in every two successive states (for example, when a single voter changes her support) there is a loss of at least one potential unit. For every state S, $\Phi(S) > \Phi(S')$. □

3.1.2 Process termination for other positional scoring rules

Consider a set of m candidates (aka. alternatives $\mathcal{A}$, outcomes) and n voters; each voter ranks all the candidates, this submitted ranking is called a vote. A voting rule is a function mapping of the n voters' votes (i.e. preferences over candidates) to one of the m candidates (the winner) in the candidate set $\mathcal{A}$.

The rules we consider here are rank-based rules (particularly positional scoring rules), which means that a vote is defined as an ordering of the candidates (with the exception of the Plurality rule, for which a vote is a single candidate). Voters submit complete rank-ordering of all candidates, not just a single candidate. The preference of a voter i is a permutation L_i of $c_1, \ldots, c_m$ from best to worst. The aggregation rule is $L_1, \ldots, L_n \rightarrow w$ where $w \in \mathcal{A}$.

Positional score rule is a voting rule that computes a score (a number) for each candidate from each individual preference profile and the alternative with the greatest score is the winner. Each positional rule is characterized by a score vector which operates on any list of best-to-worst rankings of alternatives that might be submitted by the voters. Each vote generates a vector of k scores, and the outcome of the voting rule is based only on the sum of these vectors, more specifically, only on the order (in terms of score) of the sum of the components. A difficulty with the positional scoring rules, as well as with other reasonable selection procedures based on voters' ranking, is that a different candidate can arise when one of the original losers is removed. In other words, the winner can depend on the presence of another nonwinning alternative.

In previous section (where we consider only Plurality rule or a single candidate) when a switch by voter occurs, the bloc sizes (scores) of all other candidates remain the same except two candidates; increase in score of j' = decrease in score of j (recalling definition of transition). In such a case as we have seen, the potential function decreases with each transition by a minimum of 1. Now here we consider other rules.

3.1.2.1 Borda

For Borda, voters submit complete rank-ordering of all candidates. The tactical vote for Borda is as follow: a migration occurs when a voter changes his preference list in favour of another candidate by placing that candidate at the top of his preference list and moving all other candidates one position down in his preference list.

Theorem 2 *Under Borda, the process of making tactical vote always terminates in at most* $(\frac{nm(m-1)}{2})^2$ *steps.*

Proof. A state is represented in the form of lexicographical order of numbers i.e, $N_1(S), \ldots, N_m(S)$. The numbers are the sum of the Borda scores of candidates derived from the declared preference lists of all voters at a particular state S of system. The potential function at state S is the sum of the squares of total scores of candidates i.e, $\sum_{i=1}^{m}(N_i(S))^2$. Potential difference between two successive states S and S' is $\Phi(S') - \Phi(S) = 2 \cdot (N_y(S') - N_x(S))$, when voter i changes his declared preference list in favour of candidiate y. Potential increases with each migration and maximum potential is attained when all voters have the same candidate at the top of their preference list. □

3.1.2.2 Veto and k-approval voting rule

In case of Veto and k-approval voting rule, the Borda type of tactical vote does not make any change in the score of a candidate if the same candidate is Vetoed. So for Veto rule, a voter makes a tactical vote of type exchange. This type of tactical vote is used with Veto so that the score of a candidate raises with mind change by Vetoing a different candidate. With each mind change, voter "Vetoe" a different candidate in his rank ordering. When migration occurs, at the end several alternatives have the same maximal value. Suppose ties are broken in favor of the alternative that was ranked first by more voters; if several alternatives have maximal values and were ranked first by the same number of voters, the tie is broken in favor of the alternative that was ranked second by more voters; and so on [52].

Theorem 3 *Under Veto and k-approval voting rule, the process of making tactical vote always terminates in at most* $(n \cdot (m-1))^2$ *steps.*

Proof. A state is represented as $N_1(S), \ldots, N_m(S)$. The numbers are the sum of the Veto/k-approval scores of candidates at state S. We obtained the bound using the same potential function $\sum_{i=1}^{m}(N_i(S))^2$. With each migration, potential increases and the process of making tactical vote continue until maximum potential is attained. □

3.2 Weighted votes

In the previous version of tactical voting, we considered voters have equal weights, where it does not matter which agent submitted which vote. Here in this version voters are weighted. In a weighted voting system the preferences of some voters carry more weight than the preferences of other voters. Voters' weight corresponds to the size of the voters' group $\in \mathcal{V}$ (each group act as one voter). This is why we assume weights are integers. So a vote of weight k means k different voters. In that case the bound on the number of steps required to terminate the process in terms of weights is $\sum_{i \in \mathcal{V}} w(i) = W$

where W is the total weight for Plurality voting and weights are integers. However, we are interested in bounds on the number of steps that are in terms of m and n and also we consider real weight setting.

For positional scoring rules where a vote is the rank ordering of candidates submitted by the voter, values of candidates are derived from the declared preferences of the weighted votes at a given state (say S) is given below:

$$N_j(S) = \sum_{i \in V} s_i \cdot w_i$$

$N_j(S)$ is the number that represents the total value associated with candidate j at state S, where s_i denotes the score according to the voting rule used of a candidate j in the declared preference list of voter i at state S and w_i represents the weight of voter i. For Plurality rule, the declared preference list of a voter is single candidate. Hence, the equation is,

$$N_j(S) = \sum_{i \in V} w_i$$

Here, $N_j(S)$ is the number that represents the total weight of voters who voted for candidate j at state S and w_i represents the weight of voter i.

3.2.1 Plurality rule

Observation 2 *The support of the winner never decreases.*

Theorem 4 *Under real weight setting, the process of making tactical vote terminates in $2^n mn$ number of steps.*

Proof. In Observation 2, the support of the winner either increases or stays the same. In this kind of tactical voting, the winner support increases when a new candidate becomes a winner or in other words when a winner changes. Our potential function is the support of the winning candidate i.e, $\phi_1(S) = N_{win}(S)$ and winner can have 2^n distinct values. However, when the winner stays the same and tactical vote of a voter results in raising the score of a particular candidate without making him a winner then we use another potential function ϕ_2 to find the maximum possible number of steps when the winner remains the same.

$$\phi_2(S) = \sum_{i \in \mathcal{V}} |\{x \in \mathcal{A} : x \succ_i \text{vote}_S(i)\}|$$

where $\text{vote}_S(i)$ is the candidate supported by voter i in state S. Let's say at state S a voter with weight w_i moves from candidate x to y without making him a winner. Voter i moves because $y \succ_i^S x$. Now voter i can move back to x as long as winner stays the same. Potential ϕ_2 is at most nm when the support of the winner stays the same and hence the possible number of steps are $\leq 2^n mn$. □

3.2.2 Borda

Theorem 5 *Under real weight setting, the process of making tactical vote for Borda election terminates in $2^{nm}mn$ number of steps.*

Proof. In this kind of tactical voting, the winner's Borda score increases with each migration that cause a new winner. Potential ϕ_1 increase with each such step as $\phi_1(S) = N_{win}(S)$ and can have 2^{nm} distinct values. Where winner stays the same, we use ϕ_2. Potential ϕ_2 is at most nm. Hence, we obtained the bound using the potential functions ϕ_1 and ϕ_2. □

3.3 Conclusions

We have proved with the help of a potential function that the process of mind changing terminates at some point under the Plurality rule. We also have extended the same result to other positional scoring rules like Borda, Veto and k-approval voting rule. Process termination is analyzed for both unweighted and weighted voters.

4

Manipulative voting dynamics I

The chapter introduces manipulation dynamics. We analyze the sequences of votes that may result from various voters performing manipulative votes in different weighted settings. We conclude that the process of manipulation terminates and we find bounds on the length of sequences of manipulation under Plurality rule. In Section 4.1, we dicuss increased support manipulation dynamics with examples and obtained bounds for general as well as bounded real weight settings. In Section 4.2, the Copeland rule is discussed with examples and Section 4.3 is about decreased support manipulation dynamics. In Section 4.4, we conclude the chapter.

4.1 Increased support manipulative dynamics with weighted votes

In manipulation dynamics, voters change their mind to make a "manipulative vote" that changes the outcome of the election. One can consider a manipulation successful if it causes some candidate to win that is preferred by each one of the manipulators to the candidate who would win if the manipulators voted truthfully. Suppose we have a set of voters and candidates, each voter has a weight which is a positive number and it is fixed throughout. Voters can switch to another candidate to make a manipulative vote. Throughout the process of voting dynamics, true preferences are fixed and declared preferences of individual voter may change at each state. We consider the first type of manipulation where a voter makes a manipulative vote that changes the winner and the total weight of the new winner is higher than the previous winner's weight as in Observation 1. There are various different types of moves that a voter can perform to make this type of manipulation. For example type 1 (loser to new winner move), type 2 (loser to existing winner move) and type 4a (winner to larger winner move). Type 2 move does not change the winner but the size of the winner increases with this move. Moves of type 2 do not change the winning candidate. So, type 2 moves arguably need not be considered in game-theoretic setting, although ideally we would obtain bounds that allow type 2 moves to take place. Most of our results in this chapter are for sequences of moves of types 1, 2 and 4a for different weight settings. We shall see however that in some situations one can design "smarter" potential functions that are more useful for showing a faster convergence rate.

Examples 2, 3, 4, 5, 6, 7 and 8 to follow show this kind of dynamics.

4.1.1 A few examples of manipulative dynamics with increased support of the winning candidate at each state

Examples are for the first type of manipulative dynamics- where a voter may be able to make a manipulative vote where all moves result in increasing the overall support of the new winner and a move is only allowed when the winner changes. Let N_c denotes the sum of the weights of all the voters who voted for candidate c. A winner of the state is the candidate with the highest value of N_c. Migration of voters proceeds in rounds.

Let's say initially true and declared preferences are same. In context of the Plurality rule, the declared preferences really just need to identify a single preferred candidate. But voters' true preferences are still ranking of all candidates because voters manipulate according to their true preferences. The rule for ranking the remaining candidates is only relevant for other voting rules. Let m be the number of candidates and n be the number of voters. A, B, C, D, E are the candidates. In example 2, we have $m = 5$ and $n = 4$ where 3, 5, 8 and 10 are the weights of the voters. For $i = 1, 2, 3, 4, 5$, let candidate i refer to c_i. Suppose, initially a voter with weight 3 votes for candidate 1, another voter with weight 5 votes for candidate 2 and so on. When a voter makes a manipulative vote, she switches her support to that alternative (let c be that alternative) who was not a winner in the previous state and also the total weight (N_c) of that alternative is now greater than the previous state winner; which means that alternative is the current winner of the state. In Example 2, a voter with weight 3 has preference $ABCDE$, a voter with weight 5 has a preference list B$CEAD$, a voter with weight 8 has preference D$BAEC$ and a voter with weight 10 has preference E$CDAB$. When a voter makes a manipulative vote, she changes her declared preferences as follow: she moves her favourite candidate (candidate she want to switch to) to the top of her preference list and move all other candidates one position down in her preference list. So a voter can switch to any of his favourite candidate depending upon the current state to make a manipulative vote. With each move of a voter, a new candidate becomes a winner with increased value of N_c (more than the previous state winner). Bold weights in the table show the votes moved in a round.

Example 2

Voters' weights	*True preferences*
3	*ABCDE*
5	*BCEBD*
8	*DBAEC*
10	*ECDAB*

Rounds	*A*	*B*	*C*	*D*	*E*	*Winner*
0	*3*	*5*	-	*8*	*10*	E *with* $N_E = 10$
1	-	*5*	-	*3+8=11*	*10*	D *with* $N_D = 11$
2	-	-	-	*3+8=11*	*10+5=15*	E *with* $N_E = 15$

Changes in voters' declared preferences

Rounds	*Weight of manipulating voter*	*Declared preferences*
1	*3*	$ABCDE \to DABCE$
2	*5*	$BCEBD \to EBCBD$

In Example 2, the voter with lightest weight 3 makes a move. Initially, the voter with weight 3 has a preference list $ABCDE$ and supports candidate A (according to Plurality rule). With first move, she changes her support from A to D (D is the only candidate she can switch to, to make a manipulative vote) as she does not like E to be the winner, so her preference list is now $DABCE$ (D moved to the top of her list and all other candidates moved one position down). All moves are type 1 moves (loser to new winner moves).

In Example 3, $m = 5$ and $n = 5$, where 3, 5, 8, 10 and 14 are the weights of voters. All voters have their declared preferences e.g. a voter with weight 3 has a preference $ACDBE$, a voter with weight 5 has a preference $ABECD$, a voter with weight 8 has a preference $DBECA$, a voter with weight 10 has a preference list $BDAEC$, and a voter with weight 14 has a preference $CABED$. A voter's preference list changes when he makes a manipulative vote.

Example 3

Voters' weights	*True preferences*
3	$ACDBE$
5	$ABECD$
8	$DBECA$
10	$BDAEC$
14	$CABED$

Rounds	*A*	*B*	*C*	*D*	*E*	*Winner*
0	*3+5=8*	*10*	*14*	*8*	-	C *with* $N_C = 14$
1	*3*	*10+5=15*	*14*	*8*	-	B *with* $N_B = 15$
2	-	*10+5=15*	*14+3=17*	*8*	-	C *with* $N_C = 17$
3	-	*5*	*14+3=17*	*8+10=18*	-	D *with* $N_D = 18$
4	-	*5+14=19*	*3*	*8+10=18*	-	B *with* $N_B = 19$
5	-	*5+14=19*	-	*3+8+10=21*	-	D *with* $N_D = 21$
6	-	*10+5+14=29*	-	*3+8=11*	-	B *with* $N_B = 29$

Changes in voters' declared preferences

Rounds	*Weight of manipulating voter*	*Declared preferences*
1	*5*	$ABECD \rightarrow BAECD$
2	*3*	$ACDBE \rightarrow CADBE$
3	*10*	$BDAEC \rightarrow DBAEC$
4	*14*	$CABED \rightarrow BCAED$
5	*3*	$CADBE \rightarrow DCABE$
6	*10*	$DBAEC \rightarrow BDAEC$

A voter of weight 10 has initially a true and declared preference list B$DAEC$, when at one state C becomes winner, since C is her least favourite candidate, she makes a manipulative vote and switch to D by changing his declared preferences to D$BAEC$. Then at some later state, when D becomes winner she switched back to B which is her most favourite candidate. So the voter switched back to his true preferences.
Moves of voter with weight 10: B$DAEC \rightarrow$ D$BAEC \rightarrow$ B$DAEC$.
Example 3 also shows that the same winner (B, C and D) are repeated alternatively. All moves are type 1 moves (i.e, loser to new winner) except the last move (6th round) is a type 4a move (i.e, winner to larger winner).

In Example 4, $m = 5$ and $n = 6$, where 3, 6, 7, 9, 10 and 12 are the weights of voters.

Example 4

Voters' weights	*True preferences*
3	*BCEDA*
6	*ADCEB*
7	*BDACE*
9	*CABED*
10	*DCAEB*
12	*ECADB*

Rounds	*A*	*B*	*C*	*D*	*E*	*Winner*
0	*6*	*3+7=10*	*9*	*10*	*12*	E *with* $N_E = 12$
1	*6+7=13*	*3*	*9*	*10*	*12*	A *with* $N_A = 13$
2	*6+7=13*	-	*9*	*10*	*12+3=15*	E *with* $N_E = 15$
3	*7*	-	*9*	*10+6=16*	*12+3=15*	D *with* $N_D = 16$
4	*7*	-	*9+12=21*	*10+6=16*	*3*	C *with* $N_C = 21$
5	-	-	*9+12=21*	*10+6+7=23*	*3*	D *with* $N_D = 23$
6	-	-	*9+12+3=24*	*10+6+7=23*	-	C *with* $N_C = 24$

Changes in voters' declared preferences

Rounds	*Weight of manipulating voter*	*Declared preferences*
1	7	$BDACE \rightarrow ABDCE$
2	3	$BCEDA \rightarrow EBCDA$
3	6	$ADCEB \rightarrow DACEB$
4	12	$ECADB \rightarrow CEADB$
5	7	$ABDCE \rightarrow DABCE$
6	3	$EBCDA \rightarrow CEBDA$

In Example 5, $m = 6$ and $n = 9$, where 1, 2, 4, 5, 6, 8, 9, 10 and 12 are the weights of voters. all moves are of type 1 moves (loser to new winner).

Example 5

Voters' weights	*True preferences*
1	$BDEAFC$
2	$ABDFCE$
4	$BCDAFE$
5	$ACEDBF$
6	$BDFCEA$
8	$CBDEAF$
9	$DACFEB$
10	$EDBFEA$
12	$FCDAEB$

Rounds	*A*	*B*	*C*	*D*
0	*2+5=7*	*1+4+6=11*	*8*	*9*
1	*2*	*1+4+6=11*	*8+5=13*	*9*
2	-	*1+4+6=11*	*8+5=13*	*9*
3	-	*1+4+6=11*	*8*	*9*
4	-	*1+6=7*	*8*	*9*
5	-	*1+6=7*	*8*	*9+10=19*
6	-	*1+6=7*	*8+12=20*	*9+10=19*
7	-	*1+6=7*	*8+12=20*	*2+9+10=21*
8	-	*1+6=7*	*8+12+4=24*	*2+9+10=21*
9	-	*1*	*8+12+4=24*	*2+9+10+6=27*
10	-	*1*	*8+12+4+5=29*	*2+9+10+6=27*

Rounds	*E*	*F*	*Winner*
0	*10*	*12*	F *with* $N_F = 12$
1	*10*	*12*	F *with* $N_C = 13$
2	*10*	*12+2=14*	F *with* $N_F = 14$
3	*10+5=15*	*12+2=14*	E *with* $N_E = 15$
4	*10+5=15*	*12+2+4=18*	E *with* $N_F = 18$
5	*5*	*12+2+4=18*	D *with* $N_D = 19$
6	*5*	*2+4=6*	C *with* $N_C = 20$
7	*5*	*4*	D *with* $N_D = 21$
8	*5*	-	C *with* $N_C = 24$
9	*5*	-	D *with* $N_D = 27$
10	-	-	C *with* $N_C = 29$

Changes in voters' declared preferences

Rounds	*Weight of manipulating voter*	*Declared preferences*
1	5	$ACEDBF \to CAEDBF$
2	2	$ABDFCE \to FABDCE$
3	5	$CAEDBF \to ECADBF$
4	4	$BCDAFE \to FBCDAE$
5	10	$EDBFEA \to DEBFEA$
6	12	$FCDAEB \to CFDAEB$
7	2	$FABDCE \to DFABCE$
8	4	$FBCDAE \to CFBDAE$
9	6	$BDFCEA \to DBFCEA$
10	5	$ECADBF \to CEADBF$

Moves of a voter with weight 5: $ACEDBF \to CAEDBF \to ECADBF \to CEADBF$. All moves are loser to new winner move.

4.1.2 Upper bound for General weight setting

We consider the first type of manipulative dynamics where moves involved are type 1, 2, and 4a. We work on individual migration of votes for general weight setting. A 'move' is when a voter switches his support from one candidate to another in order to change the election outcome.

We have an initial observation that the number of states (using Plurality) in general weight setting is at most m^n, since states are not visited more than once, that is a bound on the number of steps. We try to obtain a bound that is a slower-growing function than m^n. While working with the manipulative dynamics, we allow a move when the winner changes or even the winner remains the same but the support of the winner increases with each move. A voter makes a move if it can improve the total support of the new winner. Bound on the maximum possible number of steps required to terminate the process in terms of weight is $\sum_{i\in V} w(i) = W$ where W is the total weight and weights are integers. We are interested in bounds on the number of steps that are purely in terms of m and n and independent of the size of the total weight and we also want results for real weight setting.

Theorem 6 *In the general weight setting, the process of making first type of manipulation (i.e, type 1, type 2 and type 4a) terminates in* $\min(2^n, n^K)$ *steps.*

Proof. We use the potential function $\Phi(S) = N_{\text{win}}(S)$, where $N_{\text{win}}(S)$ is the sum of the weights of all voters who voted for the winning candidate at state S. All type 1, type 2 and type 4a moves increase the total score of the winner at each state, so all these 3 moves strictly increase the support of the winning candidate. So we can say potential Φ increases with each such move as Φ is the support of the winning candidate. There are at most 2^n distinct possible values for $N_{\text{win}}(S)$ since the level of support of any candidate C is determined by, for each voter i, the binary choice of whether i supports C.

If K is the number of distinct weights in the system, the level of support for a candidate C is determined by K numbers in $\{1,\ldots,n\}$. For each weight, if we are given the number of voters having that weight who support C, then we have the score of C. Hence there are $\leq n^K$ values for this quantity. The bound is thus better for small K.

□

4.1.3 Bound for a small number of voters

Claim 1 *Once voter i leaves candidate j, it can only move back to j if a heavier weighted voter i' such that $w_{i'} > w_i$, migrates to j.*

Proof. Let the current state be S, when voter i having weight w_i switched from candidate j to j', the system migrates from state S to S'. At state S', candidate j' is the winner with highest total weight. Let $N_j(S)$ be the sum of weights of voters who voted for j at state S and $N'_j(S')$ be the sum of weights of voters who voted for j' at state

S'.

$$N_{j'}(S') = \sum_{i \in V} w_i$$

According to this type of manipulative voting,

$N_j(S) < N_{j'}(S')$, also $N_j(S') = N_j(S) - w_i$
$N_j(S) = N_j(S') + w_i$
So, $N_j(S) = N_j(S') + w_i < N_{j'}(S')$

This implies that $N_{j'}(S') - N_j(S') > w_i$.

Thus, the difference between $N_{j'}(S')$ and $N_j(S')$ is greater than the weight of voter i, so a voter heavier than voter i is required to move to candidate j first. This is because j should become winner, using the allowed moves (type 1, 2 and 4a moves) which strictly increase the support of the winning candidate.

□

Here is an example: In order to find a bound on all possible moves of voters, we first consider all possible number of moves of the heaviest voter, the second heaviest voter and so on.

Moves of the heaviest weighted voter:

Figure 4.1, 4.2 and 4.3 show the moves of the heaviest voter for type 1, type 2 and type 4a moves respectively, where there are 3 candidates and w_1 is the weight of the heaviest voter. Now let's analyze Figure 4.1 for type 1 moves of the heaviest voter.

Let the heaviest voter with weight w_1 in Figure 4.1 moves from candidate y to x because candidate z is the winner and he prefers x over z. So he switched his support from candidate y to x to make candidate x a winner of the new state S'. Let $N_x(S)$ and $N_y(S)$ are the sum of weights of voters who voted for x and y respectively at state S.

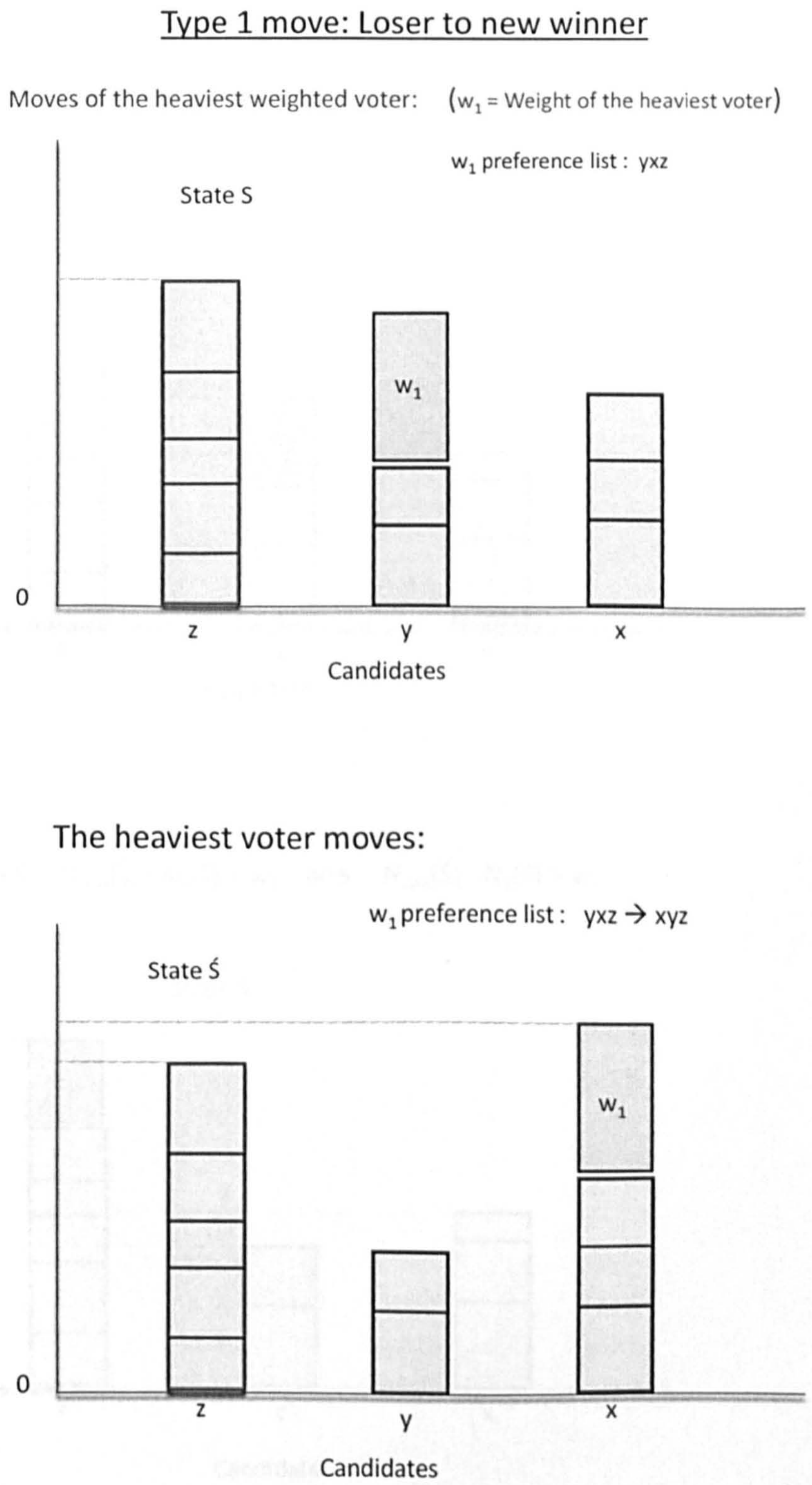

Figure 4.1: The heaviest voter moves.

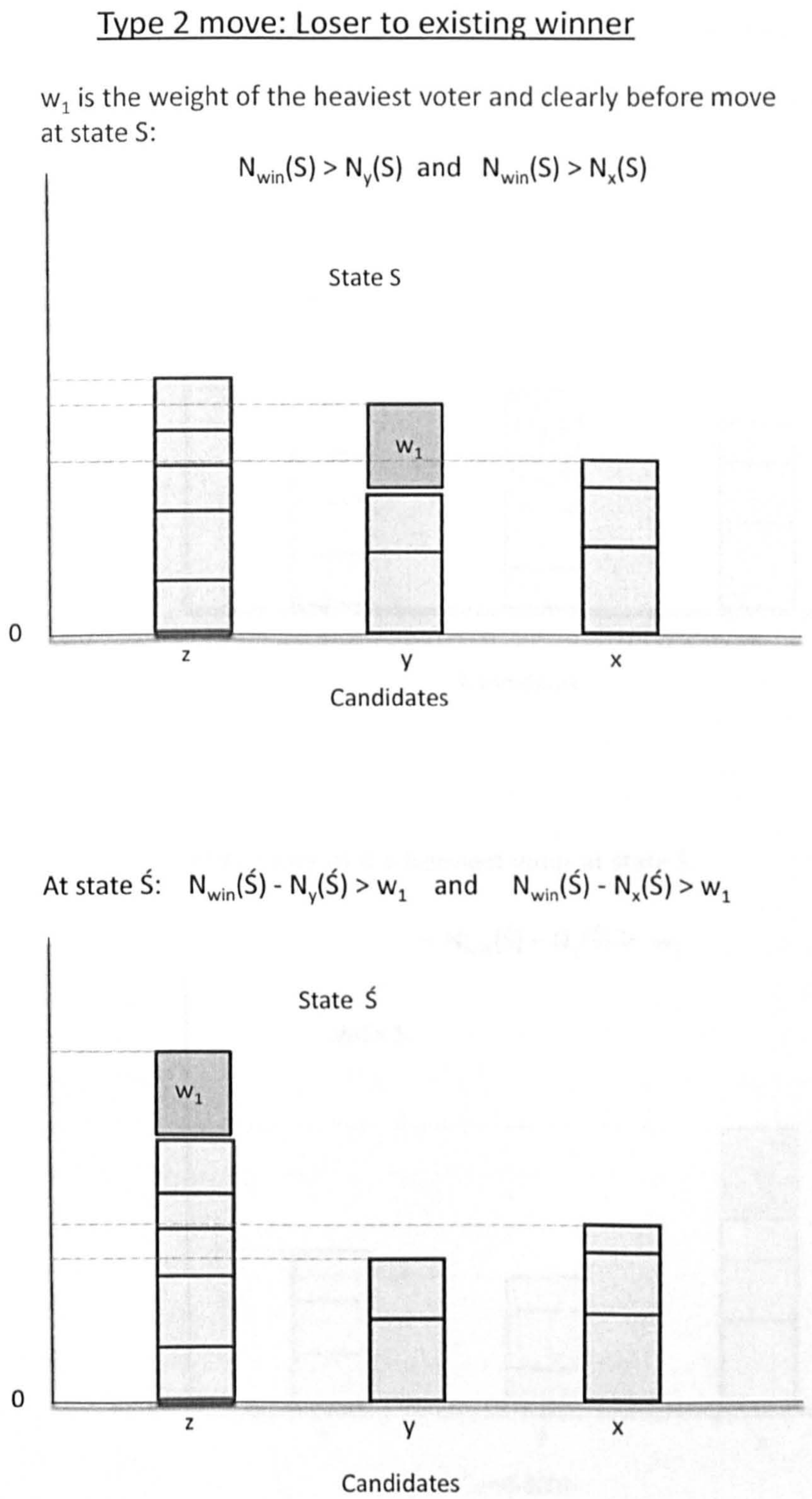

Figure 4.2: The heaviest voter moves.

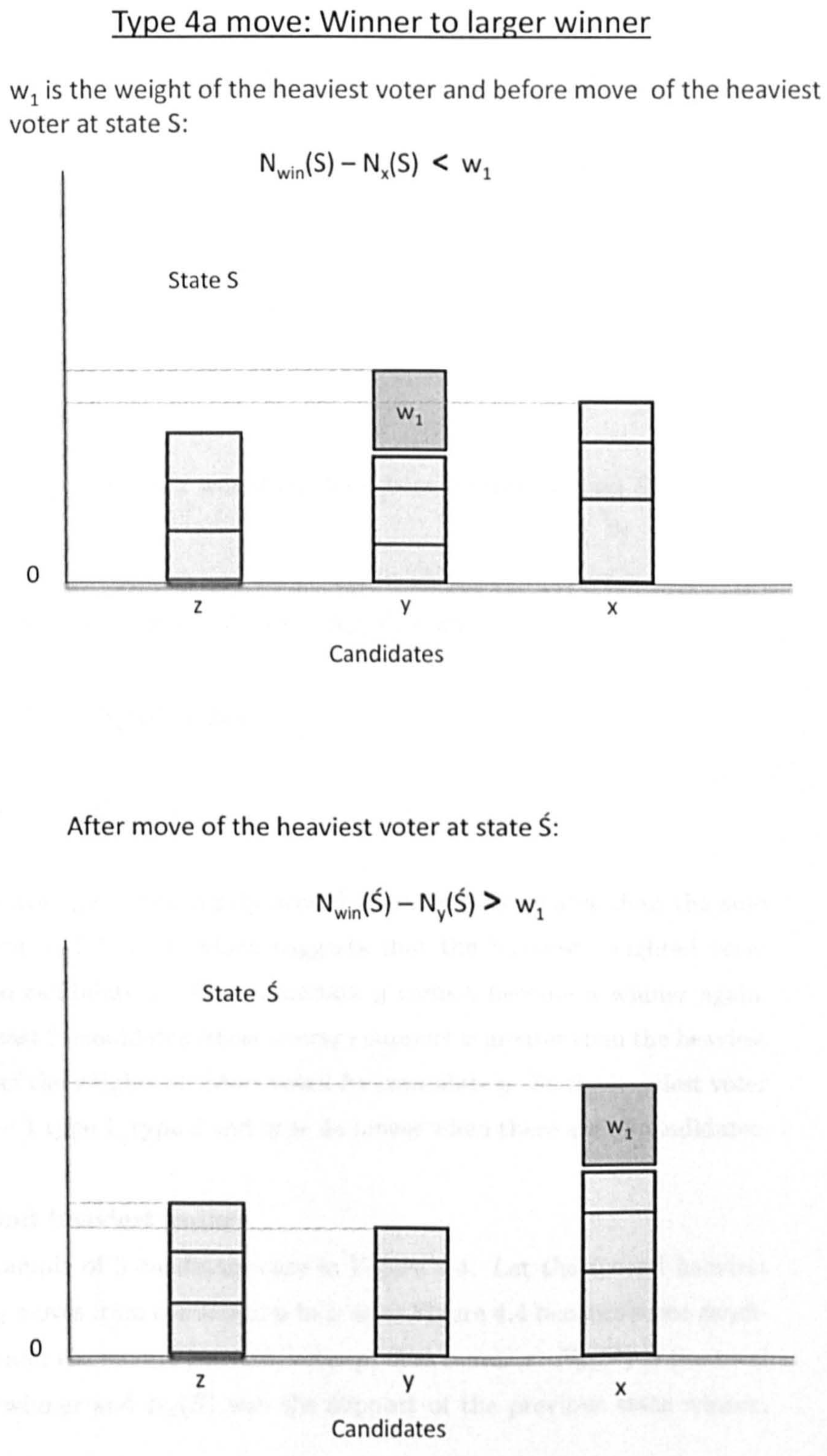

Figure 4.3: The heaviest voter moves.

When the heaviest voter switched from y to x then x is the new winner with increased size than the previous winner z (i.e, $N_x(S)$).

Claim 1 shows that the heaviest voter cannot move back to his previously supported candidate y. Also it is clear from Figure 4.1, 4.2 and 4.3 that the gap between the new winner's support i.e, $N_z(S)$ and $N_y(S)$ is greater than w_1, so moving back to y is impossible for the heaviest voter. So let's take Figure 4.1 to prove that,

$$N_x(S') - N_y(S') > w_1 \quad \text{(if not, } y \text{ could not beat } z \text{ in state } S'\text{)}$$

and also

$$N_z(S') - N_y(S') > w_1 \quad \text{(if not, } z \text{ would not have been winner in state } S\text{)}$$

$$\Longrightarrow N_x(S') > N_y(S') + w1 \quad \text{and} \quad N_z(S') > N_y(S') + w_1$$

$$\Longrightarrow N_x(S') + N_z(S') > 2N_y(S') + 2w_1$$

$$\Longrightarrow \frac{N_x(S')+N_z(S')}{2} > N_y(S') + w_1$$

This shows that the average of two highly scored candidates is greater than the sum of the heaviest weight and $N_y(S')$, which suggests that the heaviest weighted voter cannot move back to candidate y. Also candidate y cannot become a winner again. Hence, there are at least 2 candidates whose average support is greater than the heaviest weight plus the sum of the weights of voters voted for candidate y. So the heaviest voter can make at most $m-1$ type 1, type 2 and type 4a moves when there are m candidates.

Moves of the second heaviest voter:

Consider another example of 3 candidate case in Figure 4.4. Let the second heaviest voter with weight w_2 moves from candidate y to x as in Figure 4.4 because some candidate z is the winner and the second heaviest voter prefers x over z. $N_x(S')$ is the total support of the new winner and $N_z(S)$ was the support of the previous state winner.

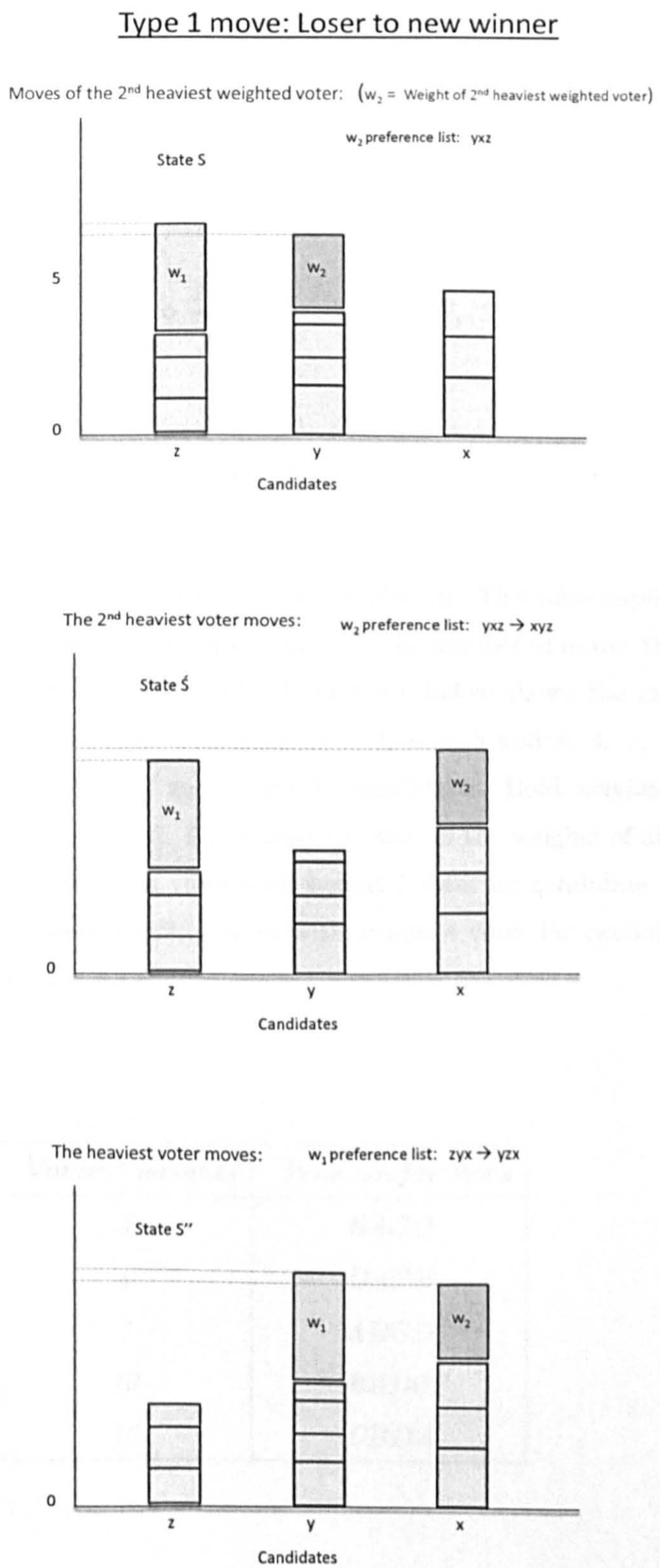

Figure 4.4: The second heaviest voter moves.

We know that

$$N_x(S') - N_y(S') > w_2$$

and also

$$N_z(S') - N_y(S') > w_2$$

$$\Longrightarrow \frac{N_x(S') + N_z(S')}{2} > N_y(S') + w_2$$

But it is possible that

$$N_x(S') - N_y(S') < w_1$$

Figure 4.4 clearly shows this type of manipulation. This also implies that the second heaviest voter moves twice to a candidate. So the number of moves that second heaviest voter can make are $2(m-1)$. Also Example 6 below shows the moves of the second heaviest voter with weight 10. Here $m = 4$, $n = 5$ and 3, 4, 7, 10 and 15 are the weights of voters. A, B C and D are the candidates. Bold weights in the table show the votes moved in a round. N_c denotes the sum of the weights of all voters who voted for candidate c. Initially, a voter with weight 7 votes for candidate A, two voters with weights 3 and 10 vote for B, a voter with weight 4 votes for candidate D and a voter with weight 15 votes for C.

Example 6

Voters' weights	*True preferences*
3	$BACD$
4	$DACB$
7	$ABCD$
10	$BADC$
15	$CBDA$

Rounds	*A*	*B*	*C*	*D*	*Winner*
0	*7*	*3+10=13*	*15*	*4*	*C with* $N_C = 15$
1	*7+10=17*	*3*	*15*	*4*	*A with* $N_A = 17$
2	*7+10=17*	*3+15=18*	-	*4*	*B with* $N_B = 18$
3	*4+7+10=21*	*3+15=18*	-	-	*A with* $N_A = 21$
4	*4+7=11*	*10+3+15=28*	-	-	*B with* $N_B = 28$

Changes in voters' declared preferences

Rounds	*Weight of manipulating voter*	*Declared preferences*
1	*10*	$BADC \rightarrow ABDC$
2	*15*	$CBDA \rightarrow BCDA$
3	*4*	$DACB \rightarrow ADCB$
4	*10*	$ABDC \rightarrow BADC$

Moves of second heaviest voter $v3$ having weight 10: B$ADC \rightarrow$ A$BDC \rightarrow$ BADC. Example 4.4 shows that the second heaviest voter can move back to her previously supported candidate once and at the end the second heaviest voter has the same true preferences as she had initially. All moves are of type 1 (loser to new winner) except the 4th move which is of type 4a (winner to larger winner). So Example 4.4 shows both types of moves.

When the 3rd heaviest voter makes a manipulative vote, in order for her to move back to her previously supported candidate, the gap between winner's value and her previously supported candidate's value is greater than the weight of the 3rd heaviest voter as per Claim 1 . Weights greater than the 3rd heaviest voter such as the second heaviest voter or the heaviest voter can move to fill in this gap. As we know the maximum possible number of moves for both the second heaviest voter and the heaviest weighted voter, so by adding all possible moves of both, we obtain the number of moves for the 3rd heaviest voter which is $3(m-1)$. Hence, the 3rd heaviest voter can move three times to the same candidate and the number of moves are $3(m-1)$. Similarly the fourth heaviest voter can make $6(m-1)$ moves and the fifth heaviest voter can make $12(m-1)$ moves and so on. To generalize the maximum possible number of moves for n voters, Let l denotes all possible moves of voters.

$n = 2 : l = (m-1) + 2(m-1) = 3(m-1) = 3.2^0(m-1)$
$n = 3 : l = (m-1) + 2(m-1) + 3(m-3) = 6(m-1) = 3.2^1(m-1)$
$n = 4 : l = (m-1) + 2(m-1) + 3(m-3) + 6(m-1) = 12(m-1) = 3.2^2(m-1)$
$n = 5 : l = (m-1)+2(m-1)+3(m-3)+6(m-1)+12(m-1) = 24(m-1) = 3.2^3(m-1)$

.

.

.

For n candidates: $3.2^{n-2}(m-1)$

This gives an exponential bound on the number of moves that can be taken in case of this type of manipulative dynamics.

Lemma 2 *In the general weight setting, if voters can make type 1, 2 and 4a moves then the heaviest voter can move* $\leq m-1$ *times, second heaviest voter can move* $\leq 2(m-1)$ *times, j-th heaviest voter can move* $\leq 2^{j-1}.(m-1)$ *times.*

Proof. Notice that for any pair x, y of candidates, if voter 1 (the heaviest voter) migrates from candidate x to y using a move of type 1,2 or 4a, then if S is the new state, we have $N_x(S) < N_{\text{win}}(S) - w_1$, i.e. the support of x is less than the support of the winner by more than w_1. Hence, thereafter x cannot possibly win, since the support of y is less than the support of the winning candidate by a quantity greater than w_1 (the largest weight of any voter). Candidate x cannot become a winner with type 1, 2 and 4a moves. Hence, voter 1 may only migrate at most $m-1$ times, and furthermore, may not return to a candidate that he previously supported as per Claim 1.

Consider voter 2. If voter 2 migrates from x to y, then in order for him to return to x, it is necessary for voter 1 to migrate to x beforehand. By an argument similar to the above, no voter with weight $\leq w_2$ is able to make x the winner. Since we have seen that voter 1 may only move to x once, it follows that (for any x) voter 2 may only move to x (at most) twice.

Applying the above idea repeatedly, for any candidate x, for $i \leq n$ voter i may only migrate to x at most 2^i times, which results in a bound of $2^i.m$ on the number of moves i may make. An upper bound on the number of moves of n voters is $m \cdot 2^n$. □

4.1.3.1 Upper bound for Bounded real weight setting

We consider the first type of manipulative dynamics with a restriction on weights of voters because we got an exponentially long sequence of moves for this type of voting dynamics. We try to find an upper bound for the bounded real weight setting. Suppose there are m candidates and n voters each voter with weight in the range $[1, w_{max}]$ and weights are fixed throughout. We are looking for an upper bound on the number of moves that is polynomial in n, m and w_{max}. A voter moves if he can improve the total support of the new winner.

Theorem 7 *In the bounded real weight setting, there are at most $mn^3(w_{max})^2$ steps required to terminate the process of making type 1 and type 4a moves.*

Proof. System consists of states and transitions. A transition from current state to next state occur when an individual voter makes a manipulative vote. We represent each state in the form of lexicographical order of numbers where each number shows the value of individual candidate in descending order. For positional scoring rules (apart from Plurality), the total weight of a candidate at a given state (say S) is given in Equation 4.1:

$$N_j(S) = \sum_{i \in V} s_i.w_i \tag{4.1}$$

$N_j(S)$ is the number that represents the sum of total weight of voters who voted for candidate j at state S, where s_i denotes the score of a candidate j in the declared preference list of voter i at state S and w_i represents the weight of voter i. In case of Plurality rule the equation is:

$$N_j(S) = \sum_{i \in V} w_i \tag{4.2}$$

Here, $N_j(S)$ is the number that represents the total weight associated with candidate j at state S and w_i represents the weight of voter i.

The total weights of all candidates are obtained from Equation 4.2 and then arranged in the following way: We define state as a sorted lexicographical order describing the total weights of candidates. State S has an associated sorted vector $N_1(S), \ldots, N_m(S)$ derived from the declared preferences of all weighted votes at state S (as per Equation 4.2) and

$$N_1(S), \ldots, N_m(S)$$

are the sum of weights of voters for candidates at state S, sorted in decreasing order such that $N_1(S)$ denotes the highest total weight that a candidate gained at state S and so on.

We introduce a potential function for type 1 and type 4a moves and demonstrate that it increases when a voter migrates. State S represented as $N_1(S), \ldots, N_m(S)$ is mapped into a real value by the potential function of that state. The potential function at a given state S is the sum of the squares of weights of voters voted for candidates [34]. We define the potential of the system at state S as,

$$\Phi(S) = \sum_{i=1}^{m} (N_i(S))^2 \tag{4.3}$$

The potential function of the sorted lexicographic order of candidates' weights always increases when a voter migrates. If we follow an iterative process where at each step one voter migration results in an increase of the total support of the winning candidate, then the potential function Φ will increase until it reaches a maximum value. The existence of the potential function Φ assures that the process will terminate after a finite number of steps at a state from which no user will have an incentive to deviate, i.e. at a PNE. Potential increases with each move (where move is the switching of voter from one candidate to another in order to make a manipulative vote). Let S and S' be two states, current state is S and when a voter makes a manipulation, transition occurs from state S to S'. The potential difference (increase in potential) between two successive states S and S' is:

$$\Phi(S') - \Phi(S) = 2.w_s.(N_x(S') - N_y(S)) \tag{4.4}$$

where w_s is the weight of a voter at state S who moved from candidate 'y' to candidate 'x' and w_s is in range $[1, w_{max}]$, $N_x(S')$ is the sum of weights of voters voted for candidate 'x' at state S' and $N_y(S)$ is the sum of weights of voters voted for candidate 'y' at state S.

Increase in potential between two successive states depends upon the value of $N_x(S') - N_y(S)$, if this value is less than $\frac{1}{2}$ then $\Phi(S') - \Phi(S) < w_s$ (where w_s is the weight of the voter at state S who makes a manipulative vote and is in range $[1, w_{max}]$).

Example below shows that $\Phi(S') - \Phi(S) \geq w_s$. We have, $m = 4$ and $n = 5$, where 1, 1.1, 1.5, 1.7 and 2.2 are the weights of voters. A, B, C and D are candidates. Weight range is $[1, w_{max}] = [1, 2.2]$. All moves are type 1 (i.e, loser to new winner) moves.

Example 7

Voters' weights	*True preferences*
1	$ACDB$
1.1	$ABDC$
1.5	$BACD$
1.7	$CDAB$
2.2	$DABC$

Rounds	*A*	*B*	*C*	*D*	*Winner*
0	*1+1.1=2.1*	*1.5*	*1.7*	*2.2*	D *with* $N_D = 2.2$
1	*1*	*1.1+1.5=2.6*	*1.7*	*2.2*	B *with* $N_B = 2.6$
2	-	*1.1+1.5=2.6*	*1+1.7=2.7*	*2.2*	C *with* $N_C = 2.7$
3	-	*1.5*	*1+1.7=2.7*	*1.1+2.2=3.3*	D *with* $N_D = 3.3$

Changes in voters' declared preferences

Rounds	*Weight of manipulating voter*	*Declared preferences*
1	*1.1*	$ABDC \rightarrow BADC$
2	*1*	$ACDB \rightarrow CADB$
3	*1.1*	$BADC \rightarrow DBAC$

Potential difference

Weight of manipulating voter (w_s)	$\Phi(S') - \Phi(S) = 2.w_s.(N_x(S') - N_y(S))$
1.1	$\Phi(1) - \Phi(0) = 1.1 = w_s$
1	$\Phi(2) - \Phi(1) = 3.4 > w_s$
1.1	$\Phi(3) - \Phi(2) = 1.54 > w_s$

Length of sequence of moves when $\Phi(S') - \Phi(S) \geq w_s$:

Example 7 shows the kind of moves where $\Phi(S') - \Phi(S) \geq w_s$. The maximum possible potential being attained when n voters all of weight w_{max} vote for the same candidate is $(n.w_{max})^2$. With each move the potential increases, since we are considering the case where increase in potential between two successive states $(\Phi(S') - \Phi(S)) \geq w_s$ and we have $w_s \geq 1$. So the maximum possible number of steps are:

$$n^2.(w_{max})^2$$

Example below shows that $\Phi(S')-\Phi(S)$ can be less than w_s. Here, $m = 3$ and $n = 5$, where 1, 1.15, 1.15, 1.25 and 1.35 are the weights of voters. A, B and C are candidates. Weight range is $[1, w_{max}] = [1, 1.35]$. All moves are type 1 (i.e, loser to new winner) moves except the last (i.e, 4th) move which is of type 4a (winner to larger winner move).

Example 8

Voters' weights	*True preferences*
1	*CBA*
1.15	*ABC*
1.15	*ABC*
1.25	*BAC*
1.35	*CAB*

Rounds	*A*	*B*	*C*	*Winner*
0	*1.15+1.15=2.3*	*1.25*	*1.35+1=2.35*	*C with* $N_C = 2.35$
1	*1.15*	*1.15+1.25=2.4*	*1.35+1=2.35*	*B with* $N_B = 2.4$
2	*1.35+1.15=2.5*	*1.15+1.25=2.4*	*1*	*A with* $N_A = 2.5$
3	*1.35+1.15=2.5*	*1+1.15+1.25=3.4*	-	*B with* $N_B = 3.4$
4	*1.15+1.35+1.15=3.65*	*1+1.25=2.25*	-	*A with* $N_A = 3.65$

Changes in voters' declared preferences

Rounds	*Weight of manipulating voter*	*Declared preferences*
1	*1.15*	$ABC \to BAC$
2	*1.35*	$CAB \to ACB$
3	*1*	$CBA \to BCA$
4	*1.15*	$BAC \to ABC$

Potential difference

Weight of manipulating voter (w_s)	$\Phi(S') - \Phi(S) = 2.w_s.(N_x(S') - N_y(S))$
1.15	$\Phi(1) - \Phi(0) = 0.23 < w_s$
1.35	$\Phi(2) - \Phi(1) = 0.405 < w_s$
1	$\Phi(3) - \Phi(2) = 4.8 > w_s$
1.15	$\Phi(4) - \Phi(3) = 0.575 < w_s$

How long is the sequence of moves for which $\Phi(S') - \Phi(S) < w_s$?

However, not always $\Phi(S') - \Phi(S) = 2.w_s.(N_x(S') - N_y(S)) \geq w_s$. Increase in potential can be quite low as weights are real numbers. At each iteration, we choose such a voter to perform a move that causes a minimum increase in potential. By doing so, we force the potential function Φ to increase as little as possible and thus we maximize the number of iterations, so as to be able to better estimate the worst-case behavior of the process. We know that $\Phi(S') - \Phi(S) = 2.w_s.(N_x(S') - N_y(S)) < w_s$ where w_s is in range $[1, w_{max}]$. This shows that potential difference between two successive states i.e., $\Phi(S') - \Phi(S)$ depends upon the value of $N_x(S') - N_y(S)$. If $N_x(S') - N_y(S) < \frac{1}{2}$ then $\Phi(S') - \Phi(S) < w_s$.

$\Phi(S') - \Phi(S) = 2.w_s.(N_x(S') - N_y(S))$, since $N_x(S') = N_x(S) + w_s$

$\Phi(S') - \Phi(S) = 2.w_s.(N_x(S) + w_s - N_y(S))$

$\Phi(S') - \Phi(S) = 2.w_s.(w_s + N_x(S) - N_y(S))$

$\Phi(S') - \Phi(S) = 2.w_s.(w_s - (N_y(S) - N_x(S)))$

Now $\Phi(S') - \Phi(S) = 2.w_s.(w_s - (N_y(S) - N_x(S)))$

In order for $\Phi(S') - \Phi(S) < w_s$, the value of $w_s - (N_y(S) - N_x(S))$ must be less than $\frac{1}{2}$.

$w_s - (N_y(S) - N_x(S)) < \frac{1}{2}$
$w_s - \frac{1}{2} < (N_y(S) - N_x(S))$

We already know that $N_y(S) - N_x(S) < w_s$.

For a low increase in potential i.e, $\Phi(S') - \Phi(S) < w_s$,

$w_s - \frac{1}{2} < (N_y(S) - N_x(S)) < w_s$

Since $N_y(S)$ and $N_x(S)$ are very close to each other. For $\Phi(S') - \Phi(S) < w_s$, gap between $N_y(S)$ and $N_x(S)$ should be greater than $w_s - \frac{1}{2}$ and less than w_s.

To prove there is a polynomial-length sequence of moves for which $\Phi(S') - \Phi(S) < w_s$. We introduce another potential function (Ψ) and demonstrate that it decreases with each migration of a voter.

$$\Psi(S) = \sum_{i \in \mathcal{V}} |\{x \in \mathcal{A} : N_{win}(S) - N_x(S) < w_i\}| \tag{4.5}$$

where w_i is the weight of voter $i \in \mathcal{V}$. $N_{win}(S)$ and $N_x(S)$ are the total support of the winning candidate and any other candidate x at state S, respectively. If initially at state S, the gap between the winning and all other candidates is less than the weight of voters, then the potential at initial state S is $\leq mn$. When a voter $i \in \mathcal{V}$ migrates from candidate y to the new winning candidate x to make a manipulative vote at state S, where for all $x, y \in \mathcal{A}$. The necessary condition for the type of manipulation we consider here is that a voter's move should result in increasing the total support of the winning candidate. For any voter i to migrate from candidate y to a new winning candidate x at state S, where $N_y(S) \leq N_{win}(S)$.

$$N_{win}(S) - N_x(S) < w_i$$

Where w_i is the weight of voter i, candidate x should be the winner of the new state S'. $N_y(S)$ and $N_x(S)$ are influenced by the migration of voter i at state S.

Lemma 3 *If a move by a voter with weight in $[1, w_{max}]$ reduces Φ by an amount less than w_s (the weight of migrating voter), then it reduces Ψ by at least 1.*

Proof. First consider the case where $N_y(S) = N_{win}(S)$. At state S, when a voter i migrates from winning candidate to a new candidate x, we know that

$$N_{win}(S) - N_x(S) < w_i \quad \text{OR,} \quad N_y(S) - N_x(S) < w_i$$

After migration at new state S', from claim 1 (Once voter i leaves candidate j, it can only move back to j if a heavier weighted voter i' such that $w_{i'} > w_i$, migrates to j).

$$N_x(S') - N_y(S') > w_i$$

Since according to this type of manipulation $N_x(S') = N_{win}(S')$, so, $N_{win}(S') - N_y(S') > w_i$.

This shows the potential decreases with migration of voter i. Because at state S, $N_{win}(S) - N_x(S) < w_i$ was true. After migration of voter i, at state S' there is at least one voter i for which $N_{win}(S') - N_y(S') < w_i$ is false. So, clearly potential Ψ reduces by at least 1.

$$\Psi(S) > \Psi(S')$$

In this case the difference between the new winner (i.e, $N_x(S')$ or $N_{win}(S')$) and previous state winner (i.e, $N_y(S)$ or $N_{win}(S)$) is equal to $N_{win}(S') - N_{win}(S)$.

Now consider the second case where $N_y(S) < N_{win}(S)$, for voter i to make a manipulative vote and migrate from candidate y to a new winning candidate x at state S, the difference $N_{win}(S) - N_x(S) < w_i$. We also know that for a low increase in potential, $w_i - \frac{1}{2} < N_y(S) - N_x(S) < w_i$. It is clear that the difference $N_y(S) - N_x(S)$ is a positive value as weights are positive numbers. So, $N_y(S) > N_x(S)$ and since

$N_{win}(S) - N_x(S) < w_i$ this implies $N_{win}(S) - N_y(S) < w_i$.

After migration of voter i, according to claim 1
$N_x(S') - N_y(S') > w_i$ and as $N_x(S') = N_{win}(S')$, thus $N_{win}(S') - N_y(S') > w_i$.

Migration of voter i causes an increase of gap between the values of the winning candidate and y. So potential still decreases with this migration by at least 1. It shows the potential drops after each migration. If voter successively perform moves, with each new state the reduction in the potential is at least 1. With each migration, the gap between the new state winner and all other candidates' value (except y) increases with an amount equal to $N_{win}(S') - N_{win}(S)$. This process continues, until the gap between winner and all $m-1$ candidates becomes greater than the weight of the heaviest voter (w_{max}). Thus, there are m moves per voter, and since there are n voters, therefore, at least mn moves are possible altogether.

□

Thus, the value of Ψ can actually increase, in steps when Φ increases by more than 1. Consequently, an upper bound of $(mn)\cdot(n.w_{max})^2$ is obtained on the number of possible moves, because Φ never decreases, and in every mn consecutive steps it increases by at least 1.

$$m.n^3.(w_{max})^2$$

So the bound is polynomial in n, m and w_{max}. □

Theorem 8 *In the bounded real weight setting with the lexicographical tie-breaking rule, at most $m^3n^4(w_{max})^2$ steps are required to terminate the process of making type 1, type 4a and 4c moves.*

Proof. We use the same potential functions as we have used in Theorem 7 Φ for larger weights and Ψ for smaller weights. However, potential function Ψ remains the same when voter i with weight w_i moves and $N_{win}(S) - N_x(S) = w_i$. In other words potential

Ψ may stay the same (when a loser to new winner move increases the number of joint winners). So we use another potential function Φ_1,

$$\Phi_1(S) = \ell(S)$$

where $\ell(S)$ is the number of joint highest-scoring candidates at state S so that l takes values in $\{1, \ldots, m\}$. The potential Φ_1 increases each time a loser to winner move results in a new winner with the same support as that of the previous winner. The possible number of consecutive moves that increases the number of joint highest-scoring candidates is at most m.

However, potential Φ_1 remains the same in 2 cases: 1) if a voter makes a winner to winner move where the new winner is of the same size as the previous winner then the potential Φ_1 does not increase, also 2) when one of the joint highest-scoring candidates who is actually a loser makes a move to another winner and the new winner is of the same size as the previous one then potential Φ_1 still stays the same. In other words when the number of joint highest-scoring candidates as well as the score of the new winner remains the same then the potential Φ_1 does not increase. In the case when the potential Φ_1 remains the same, we introduce another potential function Φ_2 as below,

$$\Phi_2(S) = \sum_{v \in V} r_v(vote_v(S))$$

where r_v is the declared rank ordering of voter v and $vote_v(S)$ is the candidate supported by voter v at state S. The potential Φ_3 goes down in both cases as mentioned above. In both cases a voter moves from a less preferred candidate to a more-preferred one. So the number of possible consecutive moves are $n \cdot m$. With each migration Φ_2 reduces by at least 1 and Φ_2 is at most $n \cdot m$. Hence, there can be at most mn steps of type 1 and 4c between other occurences of improvements.

Hence, potential Φ increases with each migration or stays the same, when Φ stays the same for the smaller weights, potential Ψ increases with each migration or stays the same, if potential Ψ stays the same then potential Φ_1 stays the same or goes up and if Φ_1 stays the same then Φ_2 goes down. This results in the overall bound of $m^3n^4(w_{max})^2$ on the number of move of all voters. □

Theorem 9 *In the bounded real weight setting, the process of making type 1 and type 4a moves terminates in $2^k m^2 n^3 (w_{max})^2$ steps when k voters may have weights greater than w_{max}.*

Proof. Suppose there are m candidates and n voters. We partitioned the n voters into two categories. The first category is that of $n-k$ voters and all $n-k$ voters have weights in the range $[1, w_{max}]$. We assume a second category in which there are k different voters who have weights greater than w_{max}, where k is a constant.

We find separate bounds for category 1 voters and a separate bound for category 2 heavy voters. First we find a sequence of migrations of category 1 voters where category 1 voters are all $n-k$ voters whose weights are in range $[1, w_{max}]$. Category 1 voters' moves are bounded by the number of migrations of $n-k$ voters in range $[1, w_{max}]$. We already proved in Theorem 7 the number of moves for n voters where all voters are in range $[1, w_{max}]$ and we obtained the expression $m.n^3.(w_{max})^2$. This implies that the number of moves for the first category is bounded by the maximum possible number of migrations of $n-k$ voters i.e, $m.(n-k)^3.(w_{max})^2$. So the moves by category 1 voters are bounded by $m.(n-k)^3.(w_{max})^2$.

Secondly we consider the migrations of category 2 voters where category 2 voters are k heavy voters having weights greater than w_{max}. This sequence of migrations starting out at a state when one of the voter from k voters migrates from one candidate to another candidate. In order to find a bound on the maximum possible moves of k heaviest voters, we have an initial observation that the number of states are at most m^k. States are not revisited, so a bound on the number of steps is m^k. We worked to get a better bound. Let's consider the moves of the k heaviest voters as in Lemma 2. Notice that for any pair x, y of candidates, if voter 1 (the heaviest voter) migrates from candidate x to y using a move of type 1,2 or 4a, then if S is the new state, we have $N_x(S) < N_{\text{win}}(S) - w_1$, i.e. the support of x is less than the support of the winner by more than w_1. Hence, thereafter x cannot possibly win, since the support of y is less than the support of the winning candidate by a quantity greater than w_1 (the largest weight of any voter). (As an aside, x could possibly become the winner if a move of type 4b was allowed.) Hence, voter 1 may only migrate at most $m-1$ times, and furthermore, may not return to a candidate that he previously supported.

Consider voter 2. If voter 2 migrates from x to y, then in order for him to return to x, it is necessary for voter 1 to migrate to x beforehand. By an argument similar to the above, no voter with weight $\leq w_2$ is able to make x the winner. Since we have seen that voter 1 may only move to x once, it follows that (for any x) voter 2 may only move to x (at most) twice.

Applying the above idea repeatedly, for any candidate x, for $i \leq k$ voter i may only migrate to x at most 2^i times, which results in a bound of $2^i.m$ on the number of moves i may make. From the maximum possible moves of the voters we obtained an upper bound on the number of moves of k voters who have weights greater than w_{max} which is $m.2^k$.

Hence, the moves of this second category grows exponentially with k, so we will assume k is a constant. These moves are bounded by the maximum number of migrations of k heavy voters which is $m.2^k$. All possiblities for k voters to combine with each other to make a new winner at each state is at most 2^k and since voters can move to m different candidates. Therefore, we derive a $m.2^k$ bound on the moves of the second category of voters . The expression $m.2^k$ is independent of $n-k$ voters with weights in range $[1, w_{max}]$ and the moves they make.

Sequence of moves of category 1 voters can occur in between any pair of moves by the k heavy voters. The bound on the moves of the k heavy voters $m.2^k$ is unaffected by the number of voters with weight in range $[1, w_{max}]$ and their moves. Hence, the Category 1 voters' move sequence can take place in between any pair of k heavy voters.

From the maximum possible number of migrations of both categories, we derive an upper bound on the number of moves of n voters. This results in the overall bound on the number of moves of all voters as:

$$2^k.m^2.(n-k)^3.(w_{max})^2$$

where k is a constant. □

4.1.4 Upper bound when the smallest weight is $\epsilon < 1$

Let there be m candidates and n voters. All voters have weights in range $[1, w_{max}]$ and there is one voter who has weight less than 1. Let the smallest weighted voter be ϵ where $\epsilon < 1$. Let system has a state S and when a voter with weight w_s moved from candidate y to a new candidate x, system migrates to a new state S'. Potential difference between two successive states S and S' is

$$\Phi(S') - \Phi(S) = 2.w_s.(w_s - (N_y(S) - N_x(S)))$$

where w_s is the weight of a voter at state S who moved from candidate 'y' to candidate 'x' and $N_x(S)$ is the sum of weights of voters voted for candidate 'x' at state S and $N_y(S)$ is the sum of weights of voters who favored candidate 'y' at state S.

Let $2.w_s.(w_s - (N_y(S) - N_x(S))) = 1$
$w_s - (N_y(S) - N_x(S)) = \frac{1}{2.w_s}$
$N_y(S) - N_x(S) = \frac{w_s - 1}{2.w_s}$
$N_y(S) - N_x(S) = \frac{2.(w_s)^2 - 1}{2.w_s}$

Maximum length of sequence of moves when $\Phi(S') - \Phi(S) \geq 1$:
The maximum possible potential being attained when n voters all of weight w_{max} vote for the same candidate is $(n.w_{max})^2$ and since the minimum potential difference between two successive states is 1 so the maximum possible number of moves are:

$$(n.w_{max})^2$$

Maximum length of sequence of moves when $\Phi(S') - \Phi(S) < 1$:
According to this type of manipulative dynamics, $N_y(S) - N_x(S) < w_s$.

To prove there is a polynomial-length sequence of moves for which $\Phi(S')-\Phi(S) < 1$ and given one smallest weight $\epsilon < 1$. We use another potential function and demonstrate that it reduces when a voter migrates.

$$\Psi(S) = \sum_{i \in \mathcal{V}} |\{x \in \mathcal{A} : N_{win}(S) - N_x(S) < w_i\}|$$

where w_i is the weight of voter $i \in \mathcal{V}$. $N_{win}(S)$ and $N_x(S)$ are the total support of the winning candidate and any other candidate x at state S, respectively. If initially at state S, the gap between the winning and all other candidates is less than the weight of voters, then the potential at initial state S is $\leq mn$. When a voter $i \in \mathcal{V}$ migrates from candidate y to the new winning candidate x to make a manipulative vote at state S, where for all $x, y \in \mathcal{A}$. For any voter i to migrate from candidate y to a new winning candidate x at state S, where $N_y(S) \leq N_{win}(S)$.

$$N_{win}(S) - N_x(S) < w_i$$

where w_i is the weight of voter i, candidate x should be the winner of the new state S'. $N_y(S)$ and $N_x(S)$ are influenced by the migration of voter i at state S.

Lemma 4 *If a move by a voter with weight in $[1, w_{max}]$ increases Φ by an amount less than 1, then it reduces Ψ by at least 1.*

Proof. First consider the case where $N_y(S) = N_{win}(S)$. At state S, when a voter i migrates from winning candidate to a new candidate x, we know that

$$N_{win}(S) - N_x(S) < w_i \quad \text{OR,} \quad N_y(S) - N_x(S) < w_i$$

After migration at new state S', from claim 1,

$$N_x(S') - N_y(S') > w_i$$

Since according to this type of manipulation $N_x(S') = N_{win}(S')$, so, $N_{win}(S') - N_y(S') > w_i$. This shows the potential decreases with migration of voter i. Because at state S, $N_{win}(S) - N_x(S) < w_i$ was true. After migration of voter i, at state S' there is at least one voter i for which $N_{win}(S') - N_y(S') < w_i$ is false. So clearly potential Ψ reduces by at least 1 and

$$\Psi(S) > \Psi(S')$$

In this case the difference between the new winner (i.e, $N_x(S')$ or $N_{win}(S')$) and previous state winner (i.e, $N_y(S)$ or $N_{win}(S)$) is equal to $N_{win}(S') - N_{win}(S)$.

Now consider the second case where $N_y(S) < N_{win}(S)$, for voter i to make a manipulative vote and migrate from candidate y to a new winning candidate x at state S, the difference $N_{win}(S) - N_x(S) < w_i$. We also know that for an increase in potential (Φ) to be less than 1, $N_y(S) > N_x(S)$. As when $N_y(S) < N_x(S)$ then we know the mimimum weight to be moved is 1, if a voter with weight ≥ 1 moves from a low supported candidate towards a high supported candidate then potential (Φ) increases by at least 1. Therefore, for a potential difference between two successive state (i.e; $\Phi(S') - \Phi(S)$) to be less than 1, $N_y(S) > N_x(S)$. It is clear that the difference $(N_y(S) - N_x(S))$ is a positive value. So, this implies that if $N_{win}(S) - N_x(S) < w_i$ then clearly $N_{win}(S) - N_y(S) < w_i$.

After migration of voter i, according to claim 1 (Once voter i leaves candidate j, it can only move back to j if a heavier weighted voter i' such that $w_{i'} > w_i$, migrates to j).

$N_x(S') - N_y(S') > w_i$ and as $N_x(S') = N_{win}(S')$, thus $N_{win}(S') - N_y(S') > w_i$.

Migration of voter i causes an increase of gap between the values of the winning candidate and y. So potential still decreases with this migration by at least 1. It shows the potential drops after each migration when weights of voters are in range $[1, w_{max}]$. If voters successively perform moves, with each new state the reduction in the potential is at least 1. With each migration, the gap between the new state winner and all other candidates' value (except y) increases with an amount equal to $(N_{win}(S') - N_{win}(S))$.

Thus each move of a voter causes an increase of the gap between the winning candidate and all other candidates' support. This process continues, until the gap between winner and all $m-1$ candidates becomes greater than the weight of the heaviest voter (w_{max}). However, if $w_s = \epsilon$, potential (Ψ) remains the same or goes down according to the Lemma 5 proved below. Reduction in potential is at least 1 when all weights are in the range $[1, w_{max}]$. Thus, at least mn moves are possible. □

Lemma 5 *If the migrating voter is the one with weight ϵ, then potential (Ψ) remains the same or goes down.*

Proof. Let w_s be the weight of the migrating voter and $w_s = \epsilon$, where $\epsilon < 1$. Suppose a voter with weight ϵ supports candidate y. At state S a voter with weight ϵ moves from candidate y to another candidate x in order to make candidate x a winner of the new state and $N_x(S)$ and $N_y(S)$ represent the total support of candidate x and y at state S respectively.

Case 1:

Let's consider the first case, where $N_y(S) > N_x(S)$. If $N_y(S)$ is greater than $N_x(S)$ by an amount greater than or equal to ϵ, then a weight greater than ϵ is required to migrate to candidate x to make a manipulative vote and make him a winner of the new state.

However, if $N_y(S)$ greater than $N_x(S)$ by an amount less than ϵ and also $N_{win}(S) - N_x(S) < \epsilon$ then ϵ can move from candidate y to x. In this case, since at state S

$$N_y(S) - N_x(S) < \epsilon \quad \text{and also} \quad N_{win}(S) - N_x(S) < \epsilon$$

$$\text{Since } N_y(S) > N_x(S) \implies N_{win}(S) - N_y(S) < \epsilon$$

After migration of a voter with weight ϵ system migrates to new state S' and at state S' potential definetely reduces because now,

$$N_x(S') = N_{win}(S') = N_x(S) + \epsilon \quad \text{and} \quad N_y(S') = N_y(S) - \epsilon$$

So $N_{win}(S') - N_y(S') > \epsilon$

This shows potential (Ψ) still reduces by at least 1 in this particular case.

Case 2:

Now consider a second case, where $N_y(S) < N_x(S)$, which shows that $N_y(S) - N_x(S)$ is a negative value. We know the potential difference between two successive states is

$$\Phi(S') - \Phi(S) = 2.w_s.(w_s - (N_y(S) - N_x(S)))$$

In this case where $w_s = \epsilon$.
$\Phi(S') - \Phi(S) = 2.\epsilon.(\epsilon - (N_y(S) - N_x(S)))$

let $N_y(S) - N_x(S) = w_d$, and since the difference $N_y(S) - N_x(S)$ is a negative value, so

$\Phi(S') - \Phi(S) = 2.\epsilon.(\epsilon + w_d)$

This shows when ϵ is very small then the potential difference between two consecutive states can be less than 1. Question is *whether potential (Ψ) still reduces by at least 1 in this case?*

Here we have $N_y(S) < N_x(S)$, we know that $N_{win}(S) - N_x(S) < \epsilon$ in order for a voter with weight ϵ to move from candidate y to x. Suppose $N_{win}(S) - N_y(S) < \epsilon$. So after migration of a voter with weight ϵ, at state S'

$N_x(S') = N_{win}(S') = N_x(S) + \epsilon$ and $N_y(S') = N_y(S) - \epsilon$

So now at state S', $N_{win}(S') - N_y(S') > \epsilon$

It is clear that potential (Ψ) still reduces by at least 1. Now suppose $N_{win}(S) - N_y(S) \geq \epsilon$. while we have $N_y(S) < N_x(S)$. A voter with weight ϵ migrates from candidate y to x when $N_{win}(S) - N_x(S) < \epsilon$. After migration,

$N_x(S') = N_{win}(S') = N_x(S) + \epsilon$ and $N_y(S') = N_y(S) - \epsilon$

and $N_{win}(S') - N_y(S') > \epsilon$

This shows potential (Ψ) may not drop with this migration of voter with a very small weight ϵ although potential (Φ) can be less than 1. However, in this particular case where $N_y(S) < N_x(S)$ potential (Φ) increases by at least 1 with the move of any other voter with weight in the range $[1, w_{max}]$. Potential (Ψ) does not reduce with migration of a voter with weight ϵ from candidate y to x in the case, when $N_y(S) < N_x(S)$ and $N_{win}(S) - N_y(S) \geq \epsilon$. It is clear that candidate y at state S' now requires a weight heavier than ϵ to become a winner. As we know with each move, the gap between the new state winner and all other candidates increases as each time a new winner has more support than the previous state winner. This gap increases with an amount equal to $N_{win}(S') - N_{win}(S)$ whether or not the potential (Ψ) reduces. This implies that double the number of steps since voter with weight ϵ moves only once. □

Theorem 10 *Under the bounded real weight setting, where k voters have weights < 1, there is a polynomial bound on the number of moves of type 1 and type 4a.*

Proof. Let there be m candidates and n voters. $n - k$ voters have weights in range

$[1, w_{max}]$ and there are k voters who have smaller weights less than 1. Let the system have state S and when a voter with weight w_s moved from candidate y to a new candidate x, system migrates to a new state S'. Potential difference between two successive states S and S' is

$$\Phi(S') - \Phi(S) = 2.w_s.(w_s - (N_y(S) - N_x(S)))$$

where w_s is the weight of a voter at state S who moved from candidate 'y' to candidate 'x' and $N_x(S)$ be the sum of weights of voters voted for candidate 'x' at state S and $N_y(S)$ be the sum of weights of voters who favored candidate 'y' at state S.

Length of sequence of moves when $\Phi(S') - \Phi(S) \geq 1$:

The maximum possible potential being attained when n voters all of weight w_{max} vote for the same candidate is $(n.w_{max})^2$. Since the minimum potential difference between two successive states is 1 so the maximum possible number of moves are:

$$(n.w_{max})^2$$

Length of sequence of moves when $\Phi(S') - \Phi(S) < 1$:

As from Lemma 4 when $\Phi(S') - \Phi(S) < 1$, potential Ψ reduces by at least 1 when a voter with weight in the range $[1, w_{max}]$ migrates from candidate y to candidate x at state S. However, Lemma 5 shows that potential Ψ reduces by at least 1 with the migration of the smallest voter with weight ϵ in all cases except one case. Lemma 5 is also true for k voters that have weights less than 1. Every migration of a voter with weight less than 1 reduces potential Ψ by at least 1 except the case when any of the smaller voter moves from candidate y to candidate x and $N_y(S) < N_x(S)$ and also $N_{win}(S) - N_y(S) \geq$ the weight of any of the k smaller migrating voter, then potential Ψ may not drop with this kind of migration. While potential Φ can still increase with an amount less than 1 with this type of migration by any of the smaller voter.

$N_y(S) < N_x(S)$ and $N_{win}(S) - N_y(S) \geq$ the weight of any of the k smaller migrating voter, shows that this kind of moves can only be made by the k smaller voters only. However, if $N_y(S) < N_x(S)$ and $N_{win}(S) - N_y(S) \geq 1$ then increase in potential Φ is greater than or equal to 1. This shows that if the move is performed by any voter in range $[1, w_{max}]$ then Φ increases by at least 1 which is already covered in the part when $\Phi(S') - \Phi(S) \geq 1$. So we focus on the length of sequence of moves by the k smaller voters where potential Ψ does not reduce and potential Φ increases by less than 1. We have already proved that the maximum potential Ψ is at most $m \cdot n$ when weights are in the range $[1, w_{max}]$. For k small weights the number of steps are at most $(k+1)mn$.

□

4.1.5 An upper bound under Bounded integer weight setting

Theorem 11 *In bounded integer weight setting, the bound on the number of type 1, type 2 and type 4a move is* $\frac{1}{2}(n \cdot w_{max})^2$.

Proof. We are looking for an upper bound on the number of moves that applies to the case where weights of voters are integers and belong to the set $\{1, 2, \ldots, w_{max}\}$. The potential difference between two consecutive states S and S' is $\Phi(S') - \Phi(S) = 2.w_s.(N_x(S') - N_y(S))$. Since weights are integers, thus with each move the potential function Φ increases by at least 2. Also, $(n \cdot w_{max})^2$ is the maximum potential being attained when n voters all of weight w_{max} vote for the same candidate. Hence, under the discrete integer weight setting at most $\frac{1}{2}(n \cdot w_{max})^2$ number of type 1, type 2 and type 4a moves are required to terminate the process.

□

4.1.6 Efficient process

We use that type of mechanism of "manipulative dynamics", in which at each step some voter switches to a better winning alternative and results in an increase in the total support of that winning alternative. Voter moves can be viewed as a sequence of improvments. Suppose there are m candidates and n voters each voter with weight in the range $[1, w_{max}]$ and weights are fixed throughout. The question is: *starting from an initial state, does this type of manipulative dynamics terminate rapidly?*

For rapid termination we are looking for moves, where at each step a move is made by the voter with the largest increase in potential. We use the same real valued potential function

$$\Phi(S) = \sum_{i=1}^{m} (N_i(S))^2 \tag{4.6}$$

which increases with each move of a voter. If we follow an iterative process where at each step one voter migration results in an increase of the total support of the winning candidate, then the potential function Φ will increase until it reaches a maximum value. The existence of the potential function assures that the process will terminate after a finite number of steps at a state when potential reaches its maximum value. We find the moves of the voters that results in rapid termination of the process. Moves that significantly increase the potential function (Φ) at each step. *For efficient processes which player is allowed to move at each step?*

We know the potential difference between two successive states S and S' is

$$\Phi(S') - \Phi(S) = 2.w_s.(N_x(S') - N_y(S))$$

where w_s is the weight of a voter in range $[1, w_{max}]$, who moved from candidate 'y' to candidate 'x' at state S.

Since $N_x(S') = N_x(S) + w_s$, thus

$$\Phi(S') - \Phi(S) = 2.w_s.(N_x(S) + w_s - N_y(S))$$
$$\Phi(S') - \Phi(S) = 2.w_s.(w_s + N_x(S) - N_y(S))$$
$$\Phi(S') - \Phi(S) = 2.w_s.(w_s - (N_y(S) - N_x(S)))$$

For a high potential change, $N_x(S) > N_y(S) \implies N_y(S) - N_x(S) < 0$.
This ensures that for higher increase in potential a voter must switch from a low supported candidate towards a highly supported candidate. Let $N_y(S) - N_x(S) = w_d$, then

$$\Phi(S') - \Phi(S) = 2.w_s.(w_s + w_d)$$

$\Phi(S') - \Phi(S) = 2.w_s.(w_s + w_d)$
$\Phi(S') - \Phi(S) = 2.w_s{}^2 + 2.w_s.w_d$

If only this type of moves are allowed where a voter migrates from a low supported candidate to a highly supported candidate then this ensures that every move of the dynamics increases the potential function by at least $2.w_s{}^2$ (as $w_s \geq 1$ and $2.w_s.w_d$ can be less than 1). Also, heavier the weight of the voter is, greater the value of $2.w_s{}^2$ is. The upper bound that we obtained previously in discrete real weight setting is $m.n^3.(w_{max})^2$. With high potential change this upper bound can be reduced, as now with each new state the increase in potential function is at least $2.w_s{}^2$ where $w_s \geq 1$, thus the maximum possible number of moves are reduced to

$$\frac{m.n^3.(w_{max})^2}{2}$$

The heaviest weighted voter first:
At each state we sort candidates in nonincreasing order of their total support (in form of sum of weights of all voters favoured the candidate) and, if at each iteration, we choose the minimum weighted voter to perform a move. By doing so, we force the potential function (Φ) to increase as little as possible and thus we maximize the number of iterations, so as to be able to better estimate the worst-case behavior of the process. Therefore, when a heavier voter moves, the change in potential is larger.

If the heaviest weighted voter (w_{max}) moves first, the process still terminates more quickly. At a state S, we sort candidates in nonincreasing order of total weight of voters favoured candidates.

$$N_1(S), \ldots, N_x(S), \ldots, N_m(S)$$

Let the heaviest weighted voter i switch from candidate y to candidate x. Then y cannot compete anymore no matter whatever is the position of candidate y at state S. Candidate y is out of race because in order for him to become a winner at some state, a voter heavier than the heaviest weighted voter is required which is not possible. Also the gap between the new winner x at state S' and all those candidates who has support less than $N_x(S)$ at state S, is greater than w_{max}. So, for all $x'' > x \Longrightarrow N_{x''}(S) < N_x(S)$, no voter can move to any of those candidates who has support less than $N_x(S)$ at state S when a voter with heaviest weight (w_{max}) moves to candidate x at state S. Thus all x'' where $N_{x''}(S) < N_x(S)$ are no more in competition. For all $x' < x \Longrightarrow N_{x'}(S) > N_x(S)$ only moves to those candidates are possible who has support greater than $N_x(S)$.

Each time a voter with the heaviest weight w_{max} moves, at least 1 candidate is out of competition because of this movement. So, the more frequently a heaviest weighted voter moves, the more quickly the process will terminate. Also if the heaviest weighted voter moves to a more highly supported candidate, more the potential will increase in a single step and more quickly the process will finish.

4.2 Other voting rules like Copeland

In the previous section we considered manipulative dynamics using Plurality, now we are extending our work to find a bound on the number of moves using the Copeland voting scheme.

Suppose there are n voters and voters have preferences on a set of m alternatives. To be precise, each voter is assumed to reveal linear preferences- a ranking of the alternatives. The outcome of the election is determined according to Copeland rule.

Let us fix the set of alternatives $\mathcal{A}$, where $|\mathcal{A}| = m$, a set of n voters $\mathcal{V} = \{1, 2, \ldots, n\}$. The system has "true preferences"(fixed) for each voter $i \in \mathcal{V}$. Let $\mathcal{L} = \mathcal{L}(\mathcal{A})$ be the set of linear preferences over $\mathcal{A}$; each voter $i \in \mathcal{V}$ has true preferences $\succ_i \in \mathcal{L}$. Voters declared preferences at state S is denoted as $\succ_i^S$.

The Copeland voting scheme ranks the candidates according to the number of pairwise contests they win minus the number they lose [45, 47]. When all candidates are compared against each other pairwise (so that they participate in the same number of contests), this is equivalent to scoring simply by the number of contests won. Copeland's winner is a candidate who maximizes the number of victories minus the number of defeats in pairwise elections.

Previous work shows that many standard voting schemes can be manipulated with only polynomial computational effort [4]. For Plurality, Borda, Maximin and Copeland it is always the case that a voter can, within polynomial time, either construct a strategic preference or else conclude that none exists [4].

4.2.1 Process termination

We consider a weighted voting system. A weighted voting system is one in which the preferences of some voters carry more weight than the preferences of other voters [10, 12]. In the weight setting we consider here, each vote has an associated weight in form of a positive numbers and is fixed throughout. A weight function is a mapping $w : \mathcal{V} \to \mathbb{R}^{+}$.

We are working on the type of manipulative dynamics- where a voter may be able to make a manipulative vote when all moves result in the winning candidate, having higher total score than the previous winner, it's a restriction to ensure termination. We are working on individual migration of voters for weighted voters' setting.

The system starts in a state where voters have complete rankings (initially, declared ranking is equal to true ranking, for each voter). A voter migrates to another candidate, if and only if, after the migration the total support of the winner strictly increased to ensure termination. The reason of changing preference list can be if his favourite candidate can't become a winner or he does not like the existing winner and prefer some other candidate over the existing winner.

At one state there is one manipulator. A manipulator chooses such a preference list that increases the total score of the new state's winner. At state S for any declared preference list $\succ^{S}$ and candidates i and j, let $i \succ^{S} j$ means that i is preferred to j with respect to declared preferences at state S. Let $Score(\succ^{S}, i)$ denotes the Copeland's score of candidate i w.r.t. declared preferences of all voters at state S. A candidate with the largest $Score(\succ^{S}, i)$ is a winner of the Copeland's election when all voters have identical (unit) weights. Copeland's score of a candidate is a function:

$$Score(\succ^{S}) : \mathcal{A} \to \mathbb{R}$$

In pairwise competition between every pair of alternatives. We assign 1 points to an alternative for winning, -1 for losing and zero for tie. The winner is the alternative with the highest score in case of unit weights. Therefore, the score is now an integer in range $\{-(m-1),\ldots,m-1\}$. We choose a manipulation using greedy manipulation that constructs a preference list for a manipulator in polynomial time such that will make a specified candidate a winner or else conclude that it is impossible [4].

The total score of a winner always increase when a voter migrates. If we follow an iterative process where at each step one voter migration results in an increase of the value of the winning candidate, then the process will terminate until a winner reaches the maximum Copeland's score. The point that score increase at each step assures that the process will terminate after a finite number of steps at a state from which no user will have an incentive to deviate, i.e. at a PNE.

4.2.2 A few examples of manipulative dynamics with Copeland voting scheme

Let voters be equi-weighted. A winner of the state is the candidate with the highest value of $Score(\succ^S)$. Migration of voters proceeds in rounds.

Voters have true preferences and declared preferences. Let m be the number of candidates and n is the number of voters. In Example 9, we have $m = 3$ and $n = 3$. A, B and C are the candidates and a set of voters $\mathcal{V} = \{v_1, v_2, v_3\}$. Voters have "true preferences" initially. In Example 9, voter 1 has preference ABC, voter 2 has a preference list CBA and voter 3 has preference BAC. When a voter makes a manipulative vote, her true preferences changes by moving her favourite candidate (candidate she wants as a winner) to the top of her preference list and place all other candidates in her preference list in such a position that don't prevent her favourite candidate from winning. With each move of a voter, a candidate becomes a winner with increased value of $Score(\succ^S)$ (more than the previous state winner).

Example 9

Voters	*Voters' true preferences*
v_1	ABC
v_2	CBA
v_3	BAC

Round	$Score(\succ^{Round}, A)$	$Score(\succ^{Round}, B)$	$Score(\succ^{Round}, C)$	*Winner*
0	*0*	*2*	*-2*	*B*

No voter can make a manipulative vote as the winner already has maximum score which cannot be further improved.

In Example 10, $m = 4$ and $n = 5$. A, B, C and D are the candidates and a set of voters $\mathcal{V} = \{v_1, v_2, v_3, v_4, v_5\}$. All voters have their declared preferences e.g. voter 1 has a preference $ABCD$, voter 2 has a preference $BDAC$, voter 3 has a preference $CABD$, voter 4 has a preference list as $BCDA$ and voter 5 has a preference $DACB$. Voter's preference list changes when he makes a manipulative vote.

Example 10

Voters	*Voters' true preferences*
v_1	$ABCD$
v_2	$BDAC$
v_3	$CABD$
v_4	$BCDA$
v_5	$DACB$

Round	$Score(\succ^{Round}, A)$	$Score(\succ^{Round}, B)$	$Score(\succ^{Round}, C)$
0	*1*	*1*	*-1*
1	*3*	*1*	*-1*

Round	$Score(\succ^{Round}, D)$	*Winner*
0	*-1*	*A and B*
1	*-3*	*A*

Changes in voters' declared preferences

Rounds	*Manipulator*	*Declared preferences*
1	v_5	$DACB \rightarrow ADCB$

Voter v_5 is the manipulator as she does not like candidate B so she manipulates to make candidate A as the only winner of round 1. No further manipulations are possible as the maximum score of a candidate is 3 which cannot be further improved.

In Example 11 , $m = 5$ and $n = 5$. A, B, C, D and E are the candidates and a set of voters $V = \{v_1, v_2, v_3, v_4, v_5\}$. All voters have their declared preferences e.g. voter 1 has a preference $ACBED$, voter 2 has a preference $BCDAE$, voter 3 has a preference $CEDBA$, voter 4 has a preference list as $DBAEC$ and voter 5 has a preference $EACDB$. Voter's preference list changes when he makes a manipulative vote.

Example 11

Voters	*Voters' true preferences*
v_1	*ACBED*
v_2	*BCDAE*
v_3	*CEDBA*
v_4	*DBAEC*
v_5	*EACDB*

Round	$Score(\succ^{Round}, A)$	$Score(\succ^{Round}, B)$	$Score(\succ^{Round}, C)$
0	*0*	*0*	*2*
1	*4*	*-2*	*2*

Round	$Score(\succ^{Round}, D)$	$Score(\succ^{Round}, E)$	*Winner*
0	*0*	*-2*	*C*
1	*-2*	*-2*	*A*

Changes in voters' declared preferences

Rounds	*Manipulator*	*Declared preferences*
1	v_4	$DBAEC \rightarrow ADBEC$

Voter v_4 changes her preferences because she does not like C, A is the only candidate she can make her a winner of the new state. So she changes her preference list so make A a winner. No other voter can manipulate, as A has the maximum score.

Now let's suppose voters are weighted. In Example 12, $m = 3$ and $n = 3$. A, B

and C are the candidates and voters have weights 2, 3 and 4. All voters have their declared preferences e.g. a voter with weight 2 has a preference ABC, a voter with weight 3 has a preference BAC and a voter with weight 4 has a preference CBA.

Example 12

Voters' weights	*Voters' true preferences*
2	ABC
3	BAC
4	CBA

Round	$Score(\succ^{Round}, A)$	$Score(\succ^{Round}, B)$	$Score(\succ^{Round}, C)$	*Winner*
0	*-4*	*6*	*-2*	B
1	*-4*	*14*	*-10*	B
2	*-8*	*18*	*-10*	B

Changes in voters' declared preferences

Rounds	*Manipulator's weight*	*Declared preferences*
1	4	$CBA \rightarrow BCA$
2	2	$ABC \rightarrow BAC$

In the first round a voter with weight 4 notices that candidate C cannot become a winner so she changes her preference list in favour of B as she does not like A. In the second round a voter with weight 2 changes her preferences in favour of B as her favourite candidate A cannot become a winner. No further manipulation is possible as winner got the maximum possible total score.

In Example 13, $m = 4$ and $n = 5$. Voters have weights 1, 2, 3, 3 and 4. All voters have their true preferences as declared preferences.

Example 13

Voters' weights	*Voters' true preferences*
1	*DABC*
2	*ABCD*
3	*BCAD*
3	*ABCD*
4	*CDAB*

Round	$Score(\succ^{Round}, A)$	$Score(\succ^{Round}, B)$	$Score(\succ^{Round}, C)$
0	*9*	*1*	*7*
1	*3*	*-5*	*13*

Round	$Score(\succ^{Round}, D)$	*Winner*
0	*-17*	*A*
1	*-11*	*C*

Changes in voters' declared preferences

Rounds	*Manipulator's weight*	*Declared preferences*
1	3	$BCAD \rightarrow CBDA$

Since a voter with weight 3 prefer C over A so she changes her preferece list in favour of C.

If you assume that the next winner's Copeland score must be higher than the previous winner's (when a manipulation takes place) in unweighted voters case, then since the Copeland score is an integer in $[-(m-1), m-1]$, so that results in a bound of $2m-1$ on the length of a sequence. In case of weighted voters the bound on the number of moves is $2Wm$, where W is the total weight of all voters and all weights are integers.

Now if we only allow manipulations that result in a clear winner rather than a tied winner, just to simplify the situation. But with this restriction, it is also possible for a manipulation to lead to a new winner having lower Copeland score than the previous one. We assume that with each migration winner changes and there is a unique winner at each state.

For such manipulations, we are looking for a polynomial bound on the number of moves. If we consider the 3 candidates case, a manipulation can only leads to a clear winner if the candidate is the last choice of the manipulator and manipulator changes his preference list in favour of him. However, this kind of move is not desirable as initially voters true preferences and declared preferences are the same and a voter is going to make his least favourite candidate a winner. Also a manipulation can only take place if a manipulator is a voter who currently voted in favour of winner and then changes his preference list in favour of some other candidate which again is not possible move. Consider an example,

Example 14

Voters	*Voters' true preferences*
v_1	$ABCD$
v_2	$ABCD$
v_3	$BDAC$
v_4	$CADB$
v_5	$DBCA$
v_6	$DBAC$

Round	$Score(\succ^{Round}, A)$	$Score(\succ^{Round}, B)$	$Score(\succ^{Round}, C)$
0	*1*	*1*	*-2*
1	*1*	*2*	*-2*

Round	$Score(\succ^{Round}, D)$	*Winner*
0	*0*	*A and B*
1	*-1*	*B*

Changes in voters' declared preferences

Rounds	*Manipulator*	*Declared preferences*
1	v_5	$DBCA \rightarrow BDCA$

Moving back voter v_5 to the same preference list does not make candidate D a winner. As D score at round 0 was already low and now with this transition at round 1 D's score goes further down, which suggests that if the manipulator move back to the same preference list, D would not become a winner with this transition. D may only become a winner if he is the least favourite candidate in a preference list and the voter manipulates by moving him to the top which is not a desirable move when voters true and declared preferences are the same, for example if voter v_1 or v_2 changes their preference list in favour of D, only then D can become a winner. This implies that for a manipulator to move back to the same preference list is not possible. So each state occurs only once. This suggests a bound of $(2m-1)^m$ and since the number of moves grows exponentially with m, so we will assume m is a constant.

4.3 Decreased support manipulative dynamics

There is a second type of manipulative dynamics in weighted vote setting, where a voter may be able to make a manipulative vote that decreases the total weight of the winning candidate and now the support of winner is lower than the previous winner (if, in the previous state, the voter supports the winner, but then changes to a new candidate who becomes the winner because he prefer new winner over the previous winner), and a move is only allowed when at each new state there is a new winner, different from previous state's winner. Only type 4b (i.e, winner to smaller winner move) is possible in this type of dynamics. The restriction of decrease in total support of the winner at each new state is to ensure the termination of process of making manipulations. We consider type 4b moves separately because moves of type 4b are troublesome and they are inconsistent with the potential functions considered so far. Can this type of move take place? Yes, if we can choose any initial declared preferences, however it may not be possible in case if initially voters' declared preferences = true preferences. In Example 15 below, voters have true preferences and declared preferences. Initially, true and declared preferences are same. Throughout the process of mind changes, true preferences are fixed and declared preferences of individual voter changes at each state. Here $m = 4$, $n = 5$ and 2, 3, 5, 7 and 8 are the weights of voters. A, B C and D are the candidates. Bold weights in the table show the votes moved in a round. N_c denotes the sum of the weights of all voters who voted for candidate c. Initially, a voter with weight 2 votes for candidate D, a voter with weight 3 votes for candidate B, a voter with weight 5 votes for candidate A and two voters with weights 7 and 8, both vote for C.

Example 15

Voters' weights	*True preferences*
2	*DCBA*
3	*BCAD*
5	*ACDB*
7	*CDBA*
8	*CABD*

Rounds	*A*	*B*	*C*	*D*	*Winner*
0	*5*	*3*	*8+7=15*	*2*	C *with* $N_C = 15$

In this type of manipulation a voter makes a move that reduces the total support of the winner. For such manipulation, in the example above either a voter with weight 7 or a voter with weight 8 should make a manipulative vote. But they both support their favourite candidate C, who is a winner also. Clearly, neither a voter with weight 7 nor a voter with weight 8 can make a manipulative vote.

This type of manipulation is only possible if in the previous state the voter supports the winner, but then changes to a new candidate who becomes the winner. In other words a manipulator is always a voter who supports the current winner and that is only possible if the current winner is not her first choice. The case where initially, true and declared preferences of voters are same. It is impossible for a voter who supports the winner and winner is her favourite candidate, to make a move that increases/decreases the total support of the new winner. So a winner-reducing move is not possible at a very initial state, which makes the process of constructing a manipulative voter sequence impossible. No other voter apart from winner's supporter can a make a move. For such voting dynamics, it seems impossible to construct a sequence of moves from an initial assignment where voters have their true preferences as declared preferences.

However, starting from a truthful state, a sequence of improvement steps is possible. We have also proved that how long this sequence of moves is. Voters whose favourite candidate is the winner, cannot make a manipulative vote at the initial state. So any other voter who can raise the total support of a new candidate by making her a winner with increased value, can make a move. In such dynamics, any voter can make a manipulative vote, not necessarily the one who support winner. However, sequence of move is not possible in case when at starting state, the total support of the winner is greater than the sum of support of all other candidates, as voters with winner cannot switch and no other voter is able to make a manipulative vote.

It can be concluded that given any set of voters and candidates, where the voters start out by supporting their favourite candidates, it is always possible to choose a sequence of switches in which each switch increases the support of the winning candidate.

For this second of type of manipulative dynamics, let's consider that voters have true preferences and his declared preferences $\neq$ true preferences. In other words voters declare false preferences to decieve other voters. Consider the same Example 15 where now voters have different true and declared preferences.

Example 16

Voters' weights	*True preferences*	*Declared preferences*
2	*BDAC*	*DCBA*
3	*CABD*	*BCAD*
5	*CADB*	*ACDB*
7	*DBCA*	*CDBA*
8	*ABCD*	*CABD*

Rounds	*A*	*B*	*C*	*D*	*Winner*
0	*5*	*3*	*8+7=15*	*2*	C *with* $N_C = 15$
1	*5+8=13*	*3*	*7*	*2*	A *with* $N_A = 13$
2	*8*	*3*	*7+5=12*	*2*	C *with* $N_C = 12$
3	*8*	*3*	*5*	*2+7=9*	D *with* $N_D = 9$

Example shows that when voters have false declared prefereces only then this type of migrations are possible. For type 4b moves, the process is reverse as the total support of the winner decreases with each move but still the process starts from a peak value and decreases until it reaches to a lowest value. The restriction of strictly decrease in total support of the winner with each migration is to ensure the termination of moves.

4.3.1 How long is the sequence of moves?

Let us fix the set of alternatives $\mathcal{A}$, where $|\mathcal{A}| = m$, a set of n voters $\mathcal{V} = \{1, 2, \ldots, n\}$where voters are weighted and we allow individual migrations. The system consists of states and transitions. Transition from current state to next state occurs when an individual voter makes a manipulative vote. We obtained a bound that is all possible subsets of votes i.e, 2^n where n is the number of voters. Set of voters who support the winner must change at each step as winner changes at each step with having lower total weight than the previous winner. At each state an individual voter switches to a new winner and her weight is added to the value of new winner. All possiblities for n voters to combine with each other to make a new winner at each state is at most 2^n. Now we are trying to get a bound that is polynomial in n.

To prove there is a polynomial-length sequence of moves for this type of manipulative dynamics. We use another potential function and demonstrate that it reduces when a voter migrates. Let S and S' be two states, current state is S and when a voter makes a manipulative vote, transition occurs from state S to S'. $N_{win}(S)$ is the sum of weights of voters voted for winner at state S and $N_y(S)$ is the sum of weights of voters voted for candidate 'y' at state S. Let w_s is the weight of a voter at state S who moved from the current winner (let say candidate y) to her favourite candidate 'x'. As we know, a manipulator is always a supporter of the current winner. According to this type of manipulative dynamics, $N_{win}(S) - N_x(S) > w_s$ so that the total support decreases. The potential function that we use is,

$$\chi(S) = \sum_{i \in \mathcal{V}} |\{x \in \mathcal{A} : N_{win}(S) - N_x(S) > w_i\}|$$

where w_i is the weight of voter $i \in \mathcal{V}$. $N_{win}(S)$ and $N_x(S)$ are the total support of the winning candidate and any other candidate x at state S, respectively. If initially at state S, the gap between the winning and all other candidates is greater than the smallest weighted voter, then the potential at initial state S is $\leq mn$. When a voter $i \in \mathcal{V}$ migrates from current winner to the new winning candidate x to make a manipulative vote at state S, where for all $x \in \mathcal{A}$. The condition for the type of manipulation we consider here is that a voter's move should result in decreasing the total support of the new winning candidate. So for voter i to move from current winner to a new winning candidate x, the condition is $N_{win}(S) - N_x(S) > w_i$.

Lemma 6 *Each migration reduces χ by at least 1 in a single move.*

Proof. Let's consider two successive states S and S'. When a voter $i \in \mathcal{V}$ migrates from current winner (say y) to new winning candidate x at state S, the necessary condition for migration is $N_{win}(S) - N_x(S) > w_i$. After migration of voter i, we know that candidate x is now the winner of state S'. Hence, $N_{win}(S') = N_x(S) + w_i$ and let $N_y(S') = N_{win}(S) - w_i$

Since $N_{win}(S) > N_x(S) + w_i$ as support of the new winner decreases with each migration so if voter i with weight w_i at state S wants to move back to previous winner with support $N_y(S')$ then this move is not allowed as it will increase the total support of the new winner. This implies that a voter with weight smaller than w_i is required to move to candidate y in order to make her a winner, which shows that $N_{win}(S') - N_y(S') < w_i$. So after migration of voter i from state S to S', there is at least one voter for which the condition $N_{win}(S) - N_x(S) > w_i$ becomes false. This proves that potential χ reduces by at least 1 with each move. □

Theorem 12 *For the second type of manipulative dynamics (i.e, type 4b move) under the real weight setting, at most mn number of steps are required to terminate the process of making type 4b move.*

Proof. As from Lemma 6, potential χ reduces by at least 1 when a weighted voter migrates from current winner to a new winner while the total support of the new winning candidate is always less than the previous state winner (i.e, 4b move). We know the maximum initial potential is $n(m-1)$. If we follow an iterative process where at each step one voter migration results in deccrease of the total support of the winning candidate, then the potential function χ will reduce until it reaches a minimum value. The existence of the potential function (χ) assures that the process will terminate after a finite number of steps at a state from which no voter will have an incentive to manipulate, i.e. at a PNE. Since according to Lemma 6 potential (χ) reduces by at least 1 with each migration so number of moves are at most $n.(m-1)$.

□

4.4 Conclusions

We considered the key problem voting schemes are confronted with, i.e, manipulation where a voter lies about their preferences in the hope of improving the election's outcome. We analyze the sequences of votes that may result from various voters performing "first and second type of manipulations" in weighted votes setting. We show that the process of making manipulative vote terminates at some point. We studied the number of steps required to reach a state where no voter has incentive to migrate. For manipulative dynamics the only restriction is that a voter migrates to a new winner with increased support or decreased support than the previous winner. We consider the voting protocols that can be manipulated in polynomial time like Plurality and Copeland.

5

Manipulative voting dynamics II

This chapter contains some results about manipulative dynamics when we allow a mixture of different types of moves. We have improved bounds for some results in terms of a new parameter K where K is the number of distinct weights. Also we have a bound on the number of moves when voters are unweighted. Section 5.1 is about the bounds obtained when different moves are allowed and also bounds with lexicographic tie-breaking rule. We also have an example of a cycle when all moves are allowed. With a lexicographic tie-breaking rule we are also allowing moves where the winner changes but the total score of the winner does not change. In Section 5.2 the bounds are dependent on parameter K. Conclusions and open questions are given in Section 5.3.

5.1 Mixture of different moves

We allow various different types of moves to take place and obtain bounds on the length of sequences of manipulations, depending on what types of manipulation are allowed. The system is modeled as a sequence of steps and in each step one voter switches from one candidate to another. We allow a mixture of different types of moves and look for bounds on the length of sequences of manipulations that can take place in the case of Plurality rule. One method of convergence in a pure Nash equilibrium is, starting from an initial state, to allow all voters to change their preferences to obtain a desirable outcome (one after the other) until they reach a pure Nash equilibrium. We are interested in the convergence time to pure Nash equilibria, that is the number of these moves. However, the length of such a sequence may be exponential in n [28]. We seek polynomial bounds that are expressed as a function of the number of voters n and number of candidates m. We also use an additional parameter K, we consider K as the number of distinct weights in general weight setting. In this chapter in some of our results, we are allowing moves where the total score of the winner does not change e.g, winner to winner move (type 4c) and loser to new winner move (type 1). The classification of various different types of moves defined for Plurality rule has been given in Chapter 2.

Let us fix the set of alternatives $\mathcal{A}$, where $|\mathcal{A}| = m$, a set of n voters $\mathcal{V} = \{1, 2, \dots, n\}$. Let $\mathcal{L} = \mathcal{L}(\mathcal{A})$ be the set of linear preferences over $\mathcal{A}$.

Lexicographic tie-breaking Voters are weighted and can only make improvement steps and if such a step is not available then they keep their current preferences. Voters have true and declared preferences. The choice of tie breaking rule has a significant impact on the outcome. Ties are broken using a priority ordering on the candidates, if there is more than one winner then the candidate who is first in the sequence is the winner. Let $\mathcal{A} = \{a_1, \dots, a_m\}$ where a_{i-1} beats a_i in event of a tie.

We have an example below (Example 17) in which $\mathcal{A} = \{A, B, C, D\}$ and there are 5 voters with weights 1, 2, 5, 5 and 6. The tie-breaking rule applies if winners receive the same level of support. Priority ordering of candidates is $A \succ B \succ C \succ D$ in case of a tie.

Example 17

Voters' weights	*Voters' true preferences*	*Declared preferences*
1	$BDAC$	$DBAC$
2	$CADB$	$ACDB$
5	$ADBC$	$ADBC$
5	$CDAB$	$CDAB$
6	$BDAC$	$BDAC$

The tables below show the sum of weights for voters of each candidate and the right-hand column indicates the winning candidate, using the declared preferences. Each state is derived from the previous state via a valid manipulation move by some voter. Above each table we indicate what manipulation was made by a voter to reach the new state. The notation "$S : i : X \longrightarrow Y$" means at state S voter i switched his support from X to Y to obtain the state indicated in the table.

State S_1

A	*B*	*C*	*D*	*Winner*
7	6	5	1	A

State $S_2 : 2 : ACDB \longrightarrow CADB$*(Winner to winner move)*

A	*B*	*C*	*D*	*Winner*
5	6	7	1	C

State $S_3 : 1 : DBAC \longrightarrow BDAC$*(Loser to winner move)*

A	*B*	*C*	*D*	*Winner*
5	7	7	0	B

State $S_4 : 2 : CADB \longrightarrow ACDB$*(Loser to winner move)*

A	*B*	*C*	*D*	*Winner*
7	7	5	0	A

5.1.1 Combination of move types that can lead to cycles

The types of moves a voter can make are: type 1 (Loser to new winner), type 4a (winner to larger winner), type 4b (winner to smaller winner) and type 4c (winner to new winner of the same size). Type 3 (winner to loser) moves are not inlcuded in the moves allowed as Meir et al. [40] has a cycle of length 4 with this type of move for Plurality voting. Type 3 moves can be replaced with type 4 (winner to winner) moves that are more natural. Also type 2 (loser to existing winner) moves are not allowed as it is not a valid manipulation (i.e; the outcome of the election remains the same), and we have an example of a cycle of length 4 below (Example 18) if this type of moves are allowed.

Let there be 3 candidates A, B and C and candidates have some fixed weighted voters who do not change their declared preferences. Candidate A has a fixed voter of weight 5.5, Candidate B has a fixed voter of weight 4.5 and candidate C has a fixed voter of weight 3.1. There are also 2 voters v_1 and v_2 who make manipulations.

Example 18

Voters	*True preferences*	S_1 *declared preferences*	*Weights*
v_1	BAC	A	*0.5*
v_2	CBA	B	*2.5*

The tables below show the sum of weights for voters of each candidate and the right-hand column indicates the winner, using the declared preferences. Each state is derived from the previous state via a manipulation move by some voter. Above each table we indicate what manipulation was made by a voter to reach the new state. The notation "$S_2 : v_1 : A \longrightarrow B$" means voter v_1 changes his support from candidate A to B at state S_2.

State S_1

A	*B*	*C*	*Winner*
6	*7*	*3.1*	*B*

State $S_2 : v_1 : A \longrightarrow B$ *(Type 2 move)*

A	*B*	*C*	*Winner*
5.5	*7.5*	*3.1*	*B*

State $S_3 : v_2 : B \longrightarrow C$*(Type 4b move)*

A	*B*	*C*	*Winner*
5.5	*5*	*5.6*	*C*

State $S_4 : v_1 : B \longrightarrow A$*(Type 1 move)*

A	*B*	*C*	*Winner*
6	*4.5*	*5.6*	*A*

State $S_5 : v_2 : C \longrightarrow B$*(Type 1 move)*

A	*B*	*C*	*Winner*
6	*7*	*3.1*	*B*

Hence, a cycle is completed in 4 steps and state $S_1 = S_5$. So we continue by considering sequences of moves where type 2 moves are disallowed.

Type 4b (winner to smaller winner) moves are the main problem for convergence. We have an example (Example 19) of a cycle with moves of type 1, type 4a, type 4b and type 4c with 3 moving voters. We here prove that there is no cycle with 2 moving voters as per Claim 2. Voters are weighted and can only make improvement steps and if such a step is not available then they keep their current preferences. Voters have true and declared preferences. The lexicographic tie-breaking rule is applied. We consider Plurality voting rule under general weight setting.

Claim 2 *Type 1 moves cannot happen in a cycle when only 2 voters can make manipulations of type 1, 4a, 4b and 4c.*

Proof. Let v_1 and v_2 be 2 voters who make manipulations and let there be two candidates x and y. Suppose voter v_1 makes a type 1 move from candidate x to y. Let S and S' be the previous and current states. At state S',

$N_{win}(S') \geq N_{win}(S)$, and x was a loser at state S, so

$$N_{win}(S') \geq N_{win}(S) \geq N_x(S),$$

$$\Rightarrow N_{win}(S') \geq N_{win}(S) > N_x(S'),$$

That means v_1 cannot move back to x to make him a winner through a valid manipulation. However, if $v_2 > v_1$ and v_2 moves to x to make him a winner, still v_1 cannot move back to x, as only 2 voters can make manipulation one after another (consecutive moves by v_1 or v_2 does not make sense). So, in order for v_1 to move back to x, voter v_2 has to leave x first, and if v_2 leaves x, then v_1 cannot move back to x as shown above. Hence, with 2 moving voters type 1 moves cannot happen in a cycle. □

If type 1 moves cannot occur in a cycle with 2 moving voters, then all other moves are winner to winner moves (i.e, type 4a, 4b and 4c) and we know that Theorem 14 shows that all winner to winner moves converges in $n \cdot (m-1)$ steps. This shows that there is no cycle with 2 moving voters when moves allowed are type 1, type 4a, type 4b and type 4c. However, we have an example of cycle with 3 moving voters as shown below.

Proposition 4 of [40] gives a simple cycle of manipulation moves involving just 2 manipulating voters, using moves of type 1 and 3. The following new example shows that cycles are also possible using only moves of types 1 and 4. The example given below contrasts with Theorem 3 of [40] that shows convergence in the case of deterministic tie-breaking and *unweighted* voters.

Let there be 5 candidates A, B, C, D and E and candidates have some fixed weighted voters. Fixed voters are: a voter with weight 1.6 supports candidate A, a voter with weight 1.9 supports candidate B, a voter with weight 2 supports candidate C, a voter with weight 1.9 supports candidate D and a voter with weight 1.8 supports candidate E. These voters are fixed and they don't change their declared preferences. There are also 3 voters v_1, v_2 and v_3 who make improvement steps and their weights and preferences are given in the table. Assume that ties are broken in favour of A, then

B, then C, and so on.

Example 19

Voters	*True preferences*	S_1 *declared preferences*	*Weights*
v_1	$BECAD$	B	*0.1*
v_2	$DEBAC$	E	*0.2*
v_3	$AECDB$	A	*0.4*

The tables below show the sum of weights for voters of each candidate and the right-hand column indicates which candidate wins, using the declared preferences. Each state is derived from the previous state via a valid manipulation move by some voter. Note that state $S_1 = S_9$, so complete a cycle. Above each table we indicate what manipulation was made by a voter to reach the new state.

State S_1

A	*B*	*C*	*D*	*E*	*Winner*
2	*2*	*2*	*1.9*	*2*	*A*

State $S_2 : v_1 : B \longrightarrow E$*(Type 1 move)*

A	*B*	*C*	*D*	*E*	*Winner*
2	*1.9*	*2*	*1.9*	*2.1*	*E*

State $S_3 : v_2 : E \longrightarrow D$*(Type 4c move)*

A	*B*	*C*	*D*	*E*	*Winner*
2	*1.9*	*2*	*2.1*	*1.9*	*D*

State $S_4 : v_1 : E \longrightarrow C$ *(Type 1 move)*

A	B	C	D	E	Winner
2	1.9	2.1	2.1	1.8	C

State $S_5 : v_2 : D \longrightarrow B$ *(Type 1 move)*

A	B	C	D	E	Winner
2	2.1	2.1	1.9	1.8	B

State $S_6 : v_3 : A \longrightarrow E$ *(Type 1 move)*

A	B	C	D	E	Winner
1.6	2.1	2.1	1.9	2.2	E

State $S_7 : v_1 : C \longrightarrow B$ *(Type 1 move)*

A	B	C	D	E	Winner
1.6	2.2	2	1.9	2.2	B

State $S_8 : v_2 : B \longrightarrow E$ *(Type 4a move)*

A	B	C	D	E	Winner
1.6	2	2	1.9	2.4	E

State $S_9 : v_3 : E \longrightarrow A$ *(Type 4b move)*

A	B	C	D	E	Winner
2	2	2	1.9	2	A

For the rest of our results we consider different subsets of moves for which the process converges under limited number of moves. Cycles can be avoided if we skip any move of type 1 (loser to new winner) or type 4a (winner to larger winner) or type 4b (winner to smaller winner). We have results in this chapter later that show that the process converges when any one of these three types of moves are skipped.

Observation below is similar to Observation 1 but with lexicographical tie-breaking rule.

Observation 3 *Moves of type 1, 2, 4a and 4c all either increase the score of the winner or the score of the winning candidate stays unchanged.*

Theorem 13 *In unrestricted real weight setting with lexicographic tie-breaking, a mixture of type 1, type 2, 4a and type 4c moves terminates.*

Proof. If we allow a mixture of type 1 (loser to new winner), type 2 (loser to existing winner), type 4a (winner to larger winner) and type 4c (winner to new winner of the same size) moves but not type 4b (winner to smaller winner) moves then of course the support of the winning candidate either increases with each move or stays the same and the score of the winner can be used as the potential function. The potential function also establishes that pure Nash equilibria can be found via sequences of moves, in which voters repeatedly switch to their preferred candidate. The set of voters who support the winner must change at each state as the winner changes when a voter moves and with each move the new winner must have the same or higher weight than the previous winner. The choice of tie-breaking has a significant impact on the outcome. Under the lexicographic tie-breaking rule, there is a priority sequence that determines the tie-breaking. For more than one winners the candidate who is first in the sequence is the winner. Let state S is the current state of the system. The potential function at state S is:

$$\Phi_1(S) = N_{win}(S)$$

where $N_{win}(S)$ is the total score of the winning candidate at a particular state S. All the four types of moves suggest that the score of the winner never decreases. The potential function Φ_1 increases at each state when an improvement move occurs (i.e, the score of the winning candidate increases). The number of different subsets of voters who can combine with each other to make a new winner at each state is 2^n, where n is the number of voters. Hence, from the potential function the possible number of steps are 2^n. However, potential Φ_1 may stay the same (when a loser to new winner move increases the number of joint winners). So we use another potential function Φ_2,

$$\Phi_2(S) = \ell(S)$$

where $\ell(S)$ is the number of joint highest-scoring candidates at state S and the value of ℓ is $1 \leq \ell \leq m$. The potential Φ_2 increases each time a loser to winner move creates a new winner with the same support as that of the previous winner. The possible number of consecutive moves that increases the number of joint highest-scoring candidates is at most m.

However, potential Φ_2 remains the same in 2 cases: 1). If a voter makes a winner to winner move where the new winner is of the same size as the previous winner then the potential Φ_2 does not increase, also 2). When one of the joint highest-scoring candidates who is actually a loser makes a move to another winner and the new winner is of the same size as the previous one then potential Φ_2 still stays the same. In other words when the number of joint highest-scoring candidates as well as score of the new winner remains the same then the potential Φ_2 does not increase. In the case when the potential Φ_2 remains the same, we introduce another potential function Φ_3 as below,

$$\Phi_3(S) = \sum_{votes_v} r_v(vote_v(S))$$

where r_v is the declared rank ordering of voter v and $vote_v(S)$ is the candidate supported by voter v at state S. The potential Φ_3 goes down in both cases as mentioned above. In both cases a voter moves from a less preferred candidate to a more-preferred one. So the number of possible consecutive moves are nm. With each migration Φ_3 reduces by at least 1 and Φ_3 is at most nm. Hence, there can be at most mn steps of type 1 and 4c between other occurences of improvements.

Theorem 13 applies that potential Φ_1 increases with each migration or stays the same, if potential Φ_1 stays the same then potential Φ_2 stays the same or goes up and if Φ_2 stays the same then Φ_3 goes down. This results in the overall bound of $2^n nm^2$ on the number of move of all voters.

□

We want results that work for type 4b moves because moves of type 4b are troublesome, since they are inconsistent with the potential functions considered so far. The following results apply alternative potential functions to restricted classes of moves that include type 4b moves.

Theorem 14 *In the unrestricted real weight setting, the process of making type 4a, 4b and 4c moves terminates within $n \cdot (m-1)$ steps.*

Proof. Notice that in a type 4 move (either 4a or 4b or 4c) a voter moves from one winner to another winner. We know these both types of move occur when a voter moves from a less-preferred candidate to a more-preferred one. Let us say voter i moves from candidate x to candidate y which is a winner to winner move that means that he truly prefers y to x so voter i will never move back from y to x which suggests that for voter i no winner to winner move from y to x is possible. Hence a voter can move only once to a candidate if all moves are winner to winner moves. So the number of possible winner to winner consecutive moves are $n \cdot (m-1)$. Technically we are using the potential function:

$$\chi(S) = \sum_{i \in V} |\{x \in \mathcal{A} : x \succ_i \text{votes}_S(i)\}| \tag{5.1}$$

where $\text{votes}_S(i)$ is the candidate supported by i in state S.

We complete the proof with the observation that each migration reduces χ by at least 1 in a single move of type 4, and χ is at most $n(m-1)$ (that upper bound occurs in a state where all voters vote for their least-preferred candidate).

□

Theorem 15 *A mixture of type 1, 4b and type 4c moves converges within $2^n mn$ steps.*

Proof. For a mixture of type 1, 4b and type 4c moves, we have the following observation.

Observation 4 *At every step the total weight of voters supporting the second-highest supported candidate either remains the same or increases when moves of voters are of type 1, 4b and 4c.*

If we allow a mixture of type 1 (loser to new winner), type 4b (winner to smaller winner) and type 4c (winner to new winner of the same size) moves then the support of the second-highest supported candidate never decreases but increases or stays the same. For example if we consider the loser to new winner type of move then after this type of move the winner of the previous state becomes the 2nd-highest supported candidate. So of course the support of the 2nd-highest scored candidate increases in that case. Let voter i switch from candidate x to candidate y at state S. At next state S', candidate y is the winner. Let w_i be the weight of voter i and $N_{win}(S)$ is the total support of the winner at state S. We use a similar potential Φ as we have used in one of our previous theorems where potential function is the support of the winning candidate. But this time instead of score of the winning candidate, our potential function is based on the score of the 2nd-highest candidate as per Observation 4, the support of the second-highest candidate increases or remains the same. Let $N_{2nd}(S)$ be the score of the second-highest candidate at state S. So the potential function is

$$\Phi(S) = N_{2nd}(S).$$

We know that

$$N_{win}(S) > N_{2nd}(S) \quad \text{and} \quad N_{win}(S) = N_{2nd}(S')$$

It is clear that type 1 move always increase the support of the second-highest supported candidate.

Now if we consider the winner to smaller winner type of move and if voter i switches from candidate x to candidate y at state S and w_i is the weight of voter i, then

If $N_{win}(S) - w_i > N_{2nd}(S)$ then

$N_{2nd}(S)$ increases.

Else

$N_{2nd}(S)$ stays the same.

Therefore, the support of the second-highest candidate (i.e, $N_{2nd}(S)$) never decreases and whenever the previous state winner becomes the second-highest supported candidate in the next state then it means the support of the second-highest candidate increased. Now the question is *how many times does the support of the second-highest supported candidate increase and how many times does it remain the same?*. From Observation 4, the score of the 2nd-highest candidate never decreases. So the largest number of times the size of the second-highest candidate can increase is 2^n as there are 2^n possible sets of voters. So the maximum possible number of times the size of the second-highest candidate increases is 2^n. The support of the 2nd-highest candidate can remain the same when a move is of type 4b (winner to smaller winner) or type 4c (winner to new winner of the same size). From Theorem 14 we know at most mn consecutive moves of type 4b and 4c are required to terminate the process of this type of manipulation. So the support of the second-highest candidate stays the same at most mn times and hence the possible number of type 1, 4b and 4c moves are $\leq 2^n mn$.

□

Let's consider it in the context of the 3-candidate case. Suppose we always prefer to make moves in which a voter moves from a less-preferred candidate to a more-preferred one. Can this limit the number of moves? Let's find the moves by the heaviest voter. Suppose voter i is the heaviest voter and he moves from candidate x to y then the next move of voter i can never be a winner to smaller winner move from y to x as he prefers y over x. So the next move possible for voter i is a loser to new winner move and after loser to new winner move, voter i cannot make any other move. To show this let's suppose the heaviest voter i makes a loser to new winner move from candidate x to y. We already know that if we allow only type 1 and type 3 moves then the support of the second highest candidate increases or remains the same. So if a loser to new winner move is made by i at state S. After the move, system migrates to new state S',

y becomes the new winner and the previous state winner becomes the second highest supported candidate. Of course at state S, the support of candidate x is less than the winner at state S. Now this winner is the second-highest supported candidate at state S'. So if next move is winner to smaller winner move then the winner's support should always be greater than the second highest supported candidate and the support of the second-highest candidate never decreases. If the next move is loser to new winner move still the heaviest voter i cannot move back to x, as it cannot make x winner. Because in case of loser to new winner move, the support of new winner must increase. Hence candidate x is out of the race and the heaviest voter i can only move twice to a candidate. However, if the heaviest voter i makes a loser to new winner move first then no further moves are possible for him. In that case he can only moves once to a candidate. Similarly second-heaviest voter can move four times to a candidate. Because if second-heaviest voter let's say j is a loser to new winner move (as winner to smaller winner move is not possible). Let say voter j moves from y to x to make a loser to new winner move. Then candidate y can only become a winner by the heaviest voter and we know the heaviest voter moves twice to a candidate. This implies that second-heaviest voter moves twice to a candidate. Similarly third-heaviest voter can move 8 times to a candidate. This can be genralized as

$$\sum_{i=1}^{n} 2^i$$

5.2 Bounds in terms of the number of distinct weights

For most of our results we used weighted system in which the preferences of some voters carry more weights than the preferences of other voters. Some of our results have dependence on weights. Here we use an additional parameter K for our general weight setting. Suppose there are K distinct weights where weights are positive real numbers and let there be n voters where $K \leq n$. For this setting we seek bounds in terms of K as well as m and n.

5.2.1 Manipulation dynamics with un-weighted voters

We consider in more detail the results obtainable in the case where the number of *distinct* weights K is small. We begin with a simple result for the case of unweighted voters (i.e. where $K = 1$).

Theorem 16 *With a lexicographic tie-breaking rule if all the voters have weight 1, any sequence of type 1 and type 2 moves has length at most n^2 or mn.*

Proof. Since with type 1 move the votes of the new winner either remains the same or increases and with type 2 move the total votes of the winner will always increase. So with both these types of moves the votes of the winner never decreases. The tie breaking rule has a significant impact on outcome. If there is more than one winner then the candidate who is first in the sequence is the winner. Applying the potential function we used before $\Phi(S) = \sum_{j=1}^{m}(N_j(S))^2$, and as proved earlier potential Φ increases with each such migration and is bounded by n^2.

□

For K different real weights we have improved bounds for Lemma 2, Theorem 7 and Theorem 15 in terms of K, n and m.

Lemma 2.

Lemma 2 can be improved when there are K different weights. We know from Lemma 2, the heaviest voter moves $m-1$ times where there are at most $n-K$ voters who can have the same weight as the heaviest voter's weight. Hence the number of moves by the heaviest weighted voters are at most $(n-K)\cdot(m-1)$. Similarly, the 2nd heaviest voter can move $2(m-1)$, let's say there are $n-K$ voters who has the second heaviest weight then the total possible moves are $\leq 2(n-K)\cdot(m-1)$ times and hence then the j-th heaviest voter can move $\leq 2^{j-1}\cdot(n-K)\cdot(m-1)$ times. Thus, the maximum possible number of moves for n voters are $3\cdot 2^{K-2}\cdot(n-K)\cdot(m-1)$, which is better than the previous bound which was exponential in n because $K \leq n$.

Theorem 7.

The bound of Theorem 7 can similarly be improved to $n^K \cdot mK$. The proof uses the two potential functions denoted Φ and Ψ. We noted that Φ may only take n^K

distinct values since it represents the support of the winning candidate. For Ψ, we rewrite Equation 5.2 as follows. Let $\mathcal{W}$ be the set of distinct weights; $|\mathcal{W}| = K$.

$$\Psi(S) = \sum_{w \in \mathcal{W}} |\{x \in \mathcal{A} : N_{\text{win}}(S) - N_x(S) < w\}| \tag{5.2}$$

For a voter i with weight w to move from the winning candidate to candidate x at state S

$$N_{win}(S) - N_x(S) < w$$

Let S' be the new state, after the move. At state S', the gap between the new winner x and previous winner is now less than w, so if previous winner was candidate y then,

$$N_{win}(S') - N_y(S') > w$$

At state S, $N_{win}(S) - N_x(S) < w_i$ was true, after migration of voter i, at state S' there is at least one weight for which $N_{win}(S') - N_x(S') > w$ is true and $n - K$ voters have no influence on the reduction of potential Ψ and Ψ drops after each migration of a weight. So, clearly potential Ψ reduces by at least 1 in a single move of 4a and potential Ψ cannot take a value larger than $m \cdot K$. Then we claim that in a similar way to the proof of Theorem 7, if Φ is not reduced in a manipulation move, then Ψ is reduced by at least 1.

Theorem 15.

From Observation 4, we know that the size of the second highest candidate never decreases and Theorem 15 uses the potential function consisting of the total weight of voters supporting the second-most supported candidate. For K different weights, we have a discrete set of weights $w_1, \ldots, w_K$ and the total weight of candidates can be represented as $\alpha_1 w_1 + \alpha_2 w_2 + \ldots + \alpha_K w_K$ where $\alpha \in [0, n]$. The general observation here is that the support of any candidate may only take at most n^K distinct values and the potential function used is $\Phi(S) = N_{2nd}(S)$, where $N_{2nd}(S)$ is the size of the second-highest supported candidate. So the possible number of steps in which the size of the second-highest always increases is n^K where $K \leq n$, which is a better bound if there are small number of distinct weights. However, when move is of type 4b, it is also that potential remains the same we already have an improved bound for this type of consecutive moves from Theorem 14 which is $K \cdot (m - 1)$. Therefore, in the general weight case, where there are K different weights, the possible number of moves are bounded by $n^K \cdot K \cdot (m-1)$.

5.3 Conclusions

Polynomial bounds have been obtained for a mixture of various different types of moves, depending on the types of manipulation allowed. The bounds obtained are dependent on parameter m, n and K. We have an example of a cycle if we allow all moves. So, allowing all types of moves and finding a sequence of moves for which the process of making manipulation terminates is still an open question. Our results help to identify what types of manipulation moves lead to cycles and how a mixture of different moves can be combined for which the sequence of moves termintes.

6

Cycles in manipulation dynamics

This chapter is about the manipulation dynamics with tie breaking rule when voters are un-weighted. We have shown with the help of examples that cycles exist for voting rules like Veto, Borda, k-Approval, Copeland and Bucklin. In Section 6.1 we give examples of Veto, Borda, k-Approval, Copeland, Bucklin rule and Plurality with runoff when in initial settings voters' true and declared preferences are different. In Section 6.2 we have examples of Borda, k-Approval, Copeland and Bucklin rules when initially true and declared preferences of voters are same. Section 6.3 concludes.

6.1 Termination with tie-breaking rule

We are working on different types of moves that voters can make to make manipulation possible. We consider positional scoring rules like Veto, Borda and k-approval voting, Bucklin rule and also Copeland's rule and Plurality with runoff. The types of moves voters can make are: loser to new winner, loser to existing winner, winner to loser, winner to winner, winner to larger winner and winner to smaller winner.

Voters are un-weighted and can only make improvement steps and if such a step is not available then they keep their current preferences. Voters have true and declared preferences. The tie breaking rule has a significant impact on outcome. Ties are broken according to an arbitrary fixed lexicographic order over the candidates. If there is more than one winner then the candidate who is first in the sequence is the winner. Meir et al.[40] have studied the phenomenon of manipulation as a dynamic process in which voters may repeatedly alter their preferences until either no further manipulations are available, or else the system goes into a cycle. Meir et al.[40] considered the possible steps of type 1, 3 and 4 moves under Plurality rule for un-weighted voters. In the paper they showed that using a simple Plurality voting rule, with a deterministic tie-breaking rule, voting dynamics will converge to a Nash equilibrium when voters always give the best response possible to the current situation. They also showed that with weighted voters, or when better replies are used, convergence is not guaranteed. We develop this line of enquiry for other voting schemes like Veto, Borda, k-approval, Copeland, Bucklin rule and Plurality with ruonff. In initial settings, true and declared preferences of voters are different. Voters can change their preferences in favour of another candidate to make a manipulative vote.

In an election, n voters express their preferences over a set of m alternatives. To be precise, each voter is assumed to reveal linear preferences- a ranking of the alternatives. The outcome of the election is determined according to a voting rule. A voting protocol is a function from the set of all preference profiles to the set of candidates. Meir et al.[40] identify cycles in the transition systems arising from alternative tie-breaking rules. We have similar examples to these that apply for alternative voting systems. Similar results for positional scoring rules like Veto, Borda, k-approval rules were also obtained independently in [39]. They also have a cycle for non positional scoring rules like Maximin rule and a result where Veto rule converges. We have cycles for other non positional voting rules like Copeland, Bucklin and Plurality with runoff.

Definition 28 *Positional scoring rule: Let $\vec{a} = \langle \alpha_1, \ldots, \alpha_m \rangle$ be a vector of integers such that $\alpha_1 \geq \alpha_2 \ldots \geq \alpha_m$. For each voter, a candidate receives α_1 points if it is ranked first by the voter, α_2 if it is ranked second etc. The score of the candidate is the total number of points the candidate receives.*

The Borda rule is the positional scoring rule with scoring vector $\vec{a} = \langle m-1, m-2, \ldots, 0 \rangle$. k-approval uses $\langle 1^k, 0^{m-k} \rangle$, and Veto uses $\vec{a} = \langle 1, 1, ..., 1, 0 \rangle$.

6.1.1 Veto Rule

Let there be 3 candidates A, B and C and there are 3 voters v_1, v_2 and v_3. Assume that ties are broken in favour of A, then B, then C. Voters v_1 and v_3 make manipultaion moves while v_2 is a "passive" voter that never moves.

Example 20

Voters	*Voters' true preferences*	*Declared preferences*
v_1	ABC	ABC
v_2	ABC	BCA
v_3	BCA	CAB

The tables below show the Veto scores obtained by each candidate and the right-hand column indicates which candidate wins, using the declared preferences. It can be checked that each state is derived from the previous state in a valid manipulation move by some voter. Note that $S5 = S1$, so complete a cycle. Above each table we indicate what manipulation was made by a voter to reach the new state. The notation "$S2 : v_3 : CAB \longrightarrow BAC$" means at state $S2$ voter v_3 changes his declared preferences from CAB to BAC.

State $S1$

A	*B*	*C*	*Winner*
2	*2*	*2*	*A*

State $S2 : v_3 : CAB \longrightarrow BAC$(Type 1 move)

A	*B*	*C*	*Winner*
2	*3*	*1*	*B*

State $S3 : v_1 : ABC \longrightarrow ACB$*(Type 3 move)*

A	*B*	*C*	*Winner*
2	*2*	*2*	*A*

State $S4 : v_3 : BAC \longrightarrow CAB$*(Type 1 move)*

A	*B*	*C*	*Winner*
2	*1*	*3*	*C*

State $S5 : v_1 : ACB \longrightarrow ABC$*(Type 3 move)*

A	*B*	*C*	*Winner*
2	*2*	*2*	*A*

The presence of a cycle shows that the sequence of moves may be infinite.

6.1.2 Borda Rule

Let there be 3 candidates A, B and C and 4 voters v_1, v_2, v_3 and v_4.

Example 21

Voters	*Voters' true preferences*	*Declared preferences*
v_1	*CBA*	*ABC*
v_2	*BCA*	*BCA*
v_3	*CBA*	*CBA*
v_4	*ABC*	*ACB*

The tables below show the Borda scores of each candidate using the declared preferences of voters.

State $S1$

A	*B*	*C*	*Winner*
4	*4*	*4*	*A*

State $S2 : v_1 : ABC \longrightarrow ACB$*(Type 1 move)*

A	*B*	*C*	*Winner*
4	*3*	*5*	*C*

State $S3 : v_4 : ACB \longrightarrow ABC$*(Type 3 move)*

A	*B*	*C*	*Winner*
4	*4*	*4*	*A*

State $S4 : v_1 : ACB \longrightarrow ABC$*(Type 1 move)*

A	*B*	*C*	*Winner*
4	*5*	*3*	*B*

State $S5 : v_4 : ABC \longrightarrow ACB$*(Type 3 move)*

A	*B*	*C*	*Winner*
4	*4*	*4*	*A*

6.1.3 k-Majority rule or k-Approval voting rule

Let there be 3 candidates A, B and C and 3 voters v_1, v_2 and v_3. Here $k = 2$.

Example 22

Voters	*Voters' true preferences*	*Declared preferences*
v_1	*CBA*	*ABC*
v_2	*BCA*	*BCA*
v_3	*ACB*	*CAB*

The numbers in the tables below show the k-approval scores of each candidate using the declared preferences of voters when $k = 2$.

State $S1$

A	*B*	*C*	*Winner*
2	*2*	*2*	*A*

State $S2 : v_1 : ABC \longrightarrow ACB$(Type 1 move)

A	*B*	*C*	*Winner*
2	*1*	*3*	*C*

State $S3 : v_3 : CAB \longrightarrow ABC$(Type 3 move)

A	*B*	*C*	*Winner*
2	*2*	*2*	*A*

State $S4 : v_1 : ACB \longrightarrow ABC$(Type 1 move)

A	*B*	*C*	*Winner*
2	*3*	*1*	*B*

State $S3 : v_3 : ABC \longrightarrow CAB$(Type 3 move)

A	*B*	*C*	*Winner*
2	*2*	*2*	*A*

6.1.4 Copeland's rule

Let there be 3 candidates A, B and C and 6 voters v_1, v_2, v_3, v_4, v_5 and v_6.

Example 23

Voters	*Voters' true preferences*	*Declared preferences*
v_1	*CBA*	*ABC*
v_2	*BCA*	*BCA*
v_3	*CAB*	*CAB*
v_4	*ACB*	*ACB*
v_5	*BAC*	*BAC*
v_6	*CBA*	*CBA*

The numbers on the bottom row of each table represents the Copeland's points of a candidate at a particular state using declared prefrences of voters. A candidate gets 1 point if it defeats an opponent in a pairwise election, 0 points in case of a draw and -1 ponits if a candidate loses a pairwise election.

State $S1$

A	*B*	*C*	*Winner*
0	*0*	*0*	*A*

State $S2 : v_1 : ABC \longrightarrow ACB$ *(Type 1 move)*

A	*B*	*C*	*Winner*
0	*-1*	*1*	*C*

State $S3 : v_4 : ACB \longrightarrow ABC$ *(Type 3 move)*

A	*B*	*C*	*Winner*
0	*0*	*0*	*A*

State $S4 : v_1 : ACB \longrightarrow ABC$ *(Type 1 move)*

A	*B*	*C*	*Winner*
0	*1*	*-1*	*B*

State $S5 : v_4 : ABC \longrightarrow ACB$ *(Type 3 move)*

A	*B*	*C*	*Winner*
0	*0*	*0*	*A*

Note, S5 = S1, so we complete the cycle for Copeland's rule.

6.1.5 Bucklin scheme

Let there be 3 candidates A, B and C and 6 voters v_1, v_2, v_3, v_4, v_5 and v_6.

Example 24

Voters	*Voters' true preferences*	*Declared preferences*
v_1	ACB	ABC
v_2	BCA	ACB
v_3	CAB	CAB
v_4	BCA	BCA
v_5	BAC	BAC
v_6	CBA	CBA

The numbers on the bottom row of each table represent the Bucklin votes of a candidate at a particular state obtained from the declared prefrences of voters. Bucklin votes are counted for all candidates ranked in the first place, in the case where no candidate has a clear majority, votes are recounted with candidates in both first and second place of voters' declared ranking.

State $S1$

A	B	C	*Winner*
4	*4*	*4*	A

State $S2 : v_2 : ACB \longrightarrow ABC$*(Type 1 move)*

A	B	C	*Winner*
4	*5*	*3*	B

State $S3 : v_1 : ABC \longrightarrow ACB$*(Type 3 move)*

A	B	C	*Winner*
4	*4*	*4*	A

State $S4 : v_2 : ABC \longrightarrow ACB$*(Type 1 move)*

A	*B*	*C*	*Winner*
4	*3*	*5*	*C*

State $S5 : v_1 : ACB \longrightarrow ABC$*(Type 3 move)*

A	*B*	*C*	*Winner*
4	*4*	*4*	*A*

6.1.6 Plurality with Runoff

Let there be 3 candidates A, B and C and 7 voters v_1, v_2, v_3, v_4, v_5, v_6 and v_7.

Example 25

Voters	*Voters' true preferences*	*Declared preferences*
v_1	*BCA*	*ACB*
v_2	*ACB*	*ACB*
v_3	*BAC*	*BAC*
v_4	*BAC*	*BAC*
v_5	*BCA*	*BCA*
v_6	*CAB*	*CAB*
v_7	*ACB*	*CBA*

The numbers in the table below represent the Plurality score of a candidate at a particular state using the declared prefrences of voters.

State $S1$

A	*B*	*C*
2	*3*	*2*

Round 1: *Two candidates with highest Plurality scores are A and B.*

Round 2: *Pairwise election between A and B.*

Candidate B beats A in a pairwise election.

State $S2 : v_7 : CBA \longrightarrow ACB$(Type 1 move)

A	B	C
3	3	1

Round 1: *Two candidates with highest Plurality scores are A and B applying the lexicographic tie-breaking rule.*

Round 2: *Pairwise election between A and B.*

A wins the pairwise election.

State $S3 : v_1 : ACB \longrightarrow CBA$(Type 3 move)

A	B	C
2	3	2

Round 1: *Two candidates with highest Plurality scores are A and B.*

Round 2: *Pairwise election between A and B.*

B wins pairwise election.

State $S4 : v_7 : ACB \longrightarrow CBA$(Type 1 move)

A	B	C
1	3	3

Round 1: *Two candidates with highest Plurality scores are B and C applying tie-breaking rule.*

Round 2: *Pairwise election between B and C.*

C **wins the election.**

State $S5 : v_1 : CBA \longrightarrow ACB$*(Type 3 move)*

A	*B*	*C*
2	*3*	*2*

Round 1: *Two candidates with highest Plurality scores are A and B.*

Round 2: *Pairwise election between A and B.*

B **beats** A **in pairwise election.**

Since $S5 = S1$, so we complete the cycle for Plurality with runoff.

6.2 Process termination when in initial settings, true and declared preferences of voters are the same

Elections are endangered by the voters (manipulation), who might be tempted to vote strategically (that is, not according to their true preferences) to obtain their preferred outcome. Voters can switch to another candidate to make a manipulative vote. The system starts in a state where voters' ranking is in favour of their most favourite candidate. If there is more than one winner then the candidate who is first in the sequence is the winner. Meir et al.[40] consider the possible steps of type 1, 3 and 4 moves under Plurality rule with deterministic tie-breaking for un-weighted voters and they showed that if $k = 2$ and if both agents use best replies or start from the tuthful state then the process of making these moves will converge. We are considering voting rules like Borda, k-approval voting, Bucklin rule and Copeland's rule when in initial settings, true and declared preferences of voters are same. Voters can make all 3 types of moves. Examples below show that cycles exist for all these voting schemes and the presence of cycle shows that the sequence of moves is infinite. Also like Plurality, Veto rule also converges if voters start from true preferences [39]. All moves of voters in examples below are the best responses but it's not necessary that a voter always use best replies. We define "best reply" as a move in which a voter always select their most-preferred candidate that can win. So in a best response, a voter optimizes the outcome (from his own perspective) if his preference list causes the election of the best possible candidate that can be elected.

6.2.1 Borda Rule

Let there be 3 candidates A, B and C and 4 voters v_1, v_2, v_3 and v_4. Ties favour A, B, C because ties are broken in favour of candidate who is first in the sequence.

Example 26

Voters	*Voters' true preferences*
v_1	ABC
v_2	CBA
v_3	ACB
v_4	BCA

State $S1$

A	*B*	*C*	*Winner*
4	*4*	*4*	*A*

State $S2 : v_2 : CBA \longrightarrow BCA$ (Type 1 move)

A	*B*	*C*	*Winner*
4	*5*	*3*	*B*

State $S3 : v_1 : ABC \longrightarrow ACB$ (Type 3 move)

A	*B*	*C*	*Winner*
4	*4*	*4*	*A*

State $S4 : v_2 : BCA \longrightarrow CBA$*(Type 1 move)*

A	*B*	*C*	*Winner*
4	*3*	*5*	*C*

State $S5 : v_1 : ACB \longrightarrow ABC$*(Type 3 move)*

A	*B*	*C*	*Winner*
4	*4*	*4*	*A*

All voters' responses are best responses.

6.2.2 k-Approval voting rule

There is a cycle for k-Approval voting rule when $m > 3$. For example let there be 4 candidates A, B, C and D and 4 voters v_1, v_2, v_3 and v_4 where $k = 2$. Ties favour A, B, C, D.

Example 27

Voters	*Voters' true preferences*
v_1	*CDBA*
v_2	*ABDC*
v_3	*BACD*
v_4	*DCAB*

State $S1$

A	*B*	*C*	*D*	*Winner*
2	*2*	*2*	*2*	*A*

State $S2 : v_1 : CDBA \longrightarrow CBDA$*(Type 1 move)*

A	*B*	*C*	*D*	*Winner*
2	*3*	*2*	*1*	*B*

State $S3 : v_2 : ABDC \longrightarrow ADBC$*(Type 3 move)*

A	*B*	*C*	*D*	*Winner*
2	2	2	2	*A*

State $S4 : v_1 : CBDA \longrightarrow CDBA$*(Type 1 move)*

A	*B*	*C*	*D*	*Winner*
2	1	2	3	*D*

State $S5 : v_2 : ADBC \longrightarrow ABDC$*(Type 3 move)*

A	*B*	*C*	*D*	*Winner*
2	2	2	2	*A*

All responses of voters are best responses. In "best responses" voters always change their preferences in favour of their most preferred candidate who can win.

6.2.3 Copeland's rule

Let there be 3 candidates A, B and C and 6 voters v_1, v_2, v_3, v_4, v_5 and v_6. Ties favour A, B, C.

Example 28

Voters	*Voters' true preferences*
v_1	ABC
v_2	ACB
v_3	BAC
v_4	BCA
v_5	CAB
v_6	CBA

State $S1$

A	*B*	*C*	*Winner*
0	0	0	*A*

State $S2 : v_4 : BCA \longrightarrow CBA$*(Type 1 move)*

A	*B*	*C*	*Winner*
0	*-1*	*1*	*C*

State $S3 : v_2 : ACB \longrightarrow ABC$*(Type 3 move)*

A	*B*	*C*	*Winner*
0	*0*	*0*	*A*

State $S4 : v_4 : CBA \longrightarrow BCA$*(Type 1 move)*

A	*B*	*C*	*Winner*
0	*1*	*-1*	*B*

State $S5 : v_2 : ABC \longrightarrow ACB$*(Type 3 move)*

A	*B*	*C*	*Winner*
0	*0*	*0*	*A*

All moves of voters are best responses.

6.2.4 Bucklin scheme

A cycle exists for Bucklin scheme when $m > 3$. Let there be 4 candidates A, B, C and D and 4 voters v_1, v_2, v_3 and v_4. Ties are broken in favour of candidate who is first in the sequence So ties favour A, B, C, D.

Example 29

Voters	*Voters' true preferences*
v_1	*ACDB*
v_2	*BDCA*
v_3	*CABD*
v_4	*DBAC*

State $S1$

A	*B*	*C*	*D*	*Winner*
3	*3*	*3*	*3*	*A (considering first 3 candidates in preference lists)*

State $S2 : v_2 : BDCA \longrightarrow BCDA$*(Type 1 move)*

A	*B*	*C*	*D*	*Winner*
2	*2*	*3*	*1*	*C (considering first 2 candidates in preference lists)*

State $S3 : v_1 : ACDB \longrightarrow ADCB$*(Type 3 move)*

A	*B*	*C*	*D*	*Winner*
3	*3*	*3*	*3*	*A (considering first 3 candidates in preference lists)*

State $S4 : v_2 : BCDA \longrightarrow BDCA$*(Type 1 move)*

A	*B*	*C*	*D*	*Winner*
2	*2*	*1*	*3*	*D (considering first 2 candidates in preference lists)*

State $S5 : v_1 : ADCB \longrightarrow ACDB$*(Type 3 move)*

A	*B*	*C*	*D*	*Winner*
3	*3*	*3*	*3*	*A (considering first 3 candidates in preference lists)*

Moves of both voters v_2 and v_1 are best responses.

6.2.5 Veto Rule

Veto rule is the positional scoring rule with scoring vector $\vec{a} = \langle 1, 1, ..., 1, 0 \rangle$. Example 20 shows that a cycle exists for Veto rule when voters don't start from their truthful state. The presence of a cycle shows that a sequence of moves may be infinite. However, the cycle might not be reachable from the state when voters' declared preferences are equal to true prefrences. In making manipulation, there is no reason to prefer one preference list over another if outcomes are the same. Veto rule converges to equilibrium when voters start from truthful state [39].

Below are a few observations for the three candidate case.

- Winner to loser: Assume we have 3 candidates A, B and C. Let's say manipulator has a preference list ABC and B is the current winner. So a winner to loser move is when manipulator changes his preference list from ABC to ACB in order to make A a winner by Vetoing the current winner (B) and improving the score of a loser C. Initially if voters have true preferences then the only move that is possible is winner to loser move. Each voter can make at most $m-1$ steps of winner to loser type.

- Loser to new winner and winner to winner move: Let's suppose a voter has true preference ABC. If A is the current winner, he does not need to manipulate. If B is the current winner, he cannot improve the score of his favourite candidate A by any means. If C is the current winner, he cannot improve the score of A or B. So with true preferences a voter cannot manipulate with loser to new winner move, as all a voter can do is to improve the score of his least favourite candidate C, which cannot give a favourable outcome. Hence, if the manipulator has already made a winner to loser move, only then he can make a loser to new winner move. Let's say voter i makes a manipulation by making a winner to loser move and changes his preference list from ABC to ACB. Later on when C becomes a winner and as according to voter's i true preferences, he prefers B over C. So he switched back to his previous preference list if he can make B a winner, so that is a winner to winner move i.e, $ACB \longrightarrow ABC$. So a loser to new winner move is not possible in the three candidate case and hence the score of the winner never increases.

6.3 Conclusions

We have considered manipulation dynamics with lexicographic tie-breaking rule for different voting schemes like Veto, Borda, k-Approval, Copeland, Bucklin and Plurality with runoff, when voters are unweighted. We have exhibited cycles to show that sequences of moves may be infinite. However, alternative valid moves exist which would bring the sequence to an end. It is an open question whether certain sets of voters exist for which infinite sequences of valid manipulations are unavoidable. For instance, in our examples if we disallow type 3 move, then the process will converge as in case of type 1 moves, if winner's score is the potential function then potential will either increase or stays the same with type 1 moves. We have a cycle for Plurality with runoff when voters' start from their declared preferences that are not their true preferences. An open question is whether Plurality with runoff starting from true preferences causes cycles? Considering Veto, Borda, k-Approval, Copeland, and Bucklin elections with runoff is not interesting because we have already shown cycles for these rules without runoff.

7

Summary

This chapter gives a brief summary of the core findings of the study and their practical implications. We also offer recommendations for further research on the topic and suggest some open problems. We answer the main questions stated in the introduction, does a system converge and if so, how quickly does it converge?

7.1 Summary of major findings

Our results apply to various subsets of types of possible manipulation moves and the potential functions used, guarantee the termination of the dynamic process in which voters repeatedly alter their declared preferences until no further manipulation is possible. Alternative types of moves seem to require alternative potential functions. We show how potential functions are useful for showing convergence in voting schemes. Our main focus is on thePlurality voting with weighted voters. We apply lexicographic tie-breaking rule in case of ties. We try to find results where speed of convergence is expressed as a function of the number of voters n, the number of candidates m, and other parameters, e.g. for weighted voters, we consider the number of distinct weights K.

In Chapter 3, we have proved with the help of a potential function that the process of making tactical vote terminates at some point under thePlurality rule and bound on the possible number of steps are in terms of number of candidates m and number of voters n. We also have extended the same result to other positional scoring rules like Borda and Veto. Process termination is analyzed for both weighted and unweighted setting.

In Chapter 4, we considered the key problem voting schemes are confronted with, i.e, manipulation where a voter lies about their preferences in the hope of improving the election's outcome. We analyze the sequences of votes that may result from various voters performing "first and second type of manipulations" in weighted votes setting. In the first type of manipulation, with each move the support of the new winner increases. Type 1 and 4a moves can take place in this type of manipulative dynamics. In the second type of manipulation, the support of the new winner decreases with each manipulation move. Only 4b type of moves are possible in this type of dynamics. We show that the process of making manipulative votes terminates at some point. We studied the number of steps required to reach a state where no voter has an incentive to migrate. Our bounds on the lengths of sequences of manipulations depending on what types of manipulations are allowed. In this chapter most of the moves allowed are similar types of moves e.g. all moves that always increase the score of the winner. We consider the voting protocols that can be manipulated in polynomial time likePlurality and Copeland voting rules.

In Chapter 5, we allow a mixture of different types of moves and polynomial bounds have been obtained for a mixture of these moves, depends on the types of manipulation allowed. The bounds obtained are dependent on parameter m, n and some results are also in terms of a new parameter K where K is the number of distinct weights. We have an example of a cycle with 3 moving voters when all types of moves are allowed and we show that the process of making manipulation terminates for different subsets of moves using different versions of potential functions. In case of multiple winners, we apply lexicographic tie-breaking rule to break ties. The problem being that in some cases, sequences of these self-improving moves may be exponentially-long. The following questions arise: can there be better bounds that are polynomial in terms of m and n?

In Chapter 6, we have considered manipulation dynamics with lexicographic tie-breaking rule for different voting schemes like Veto, Borda, k-Approval, Copeland and

Bucklin, when voters are unweighted. Meir et al. [40] identify cycles in the transition systems arising from alternative tie-breaking rules forPlurality. We have similar examples to these that apply for alternative voting systems. We give examples of Veto, Borda, k-Approval, Copeland, Bucklin rule andPlurality with runoff when in initial settings voters' true and declared preferences are different and also when initially both preferences of voters are the same. We have exhibited cycles to show that sequences of moves for these voting schemes may be infinite. However, to avoid cycles alternative valid moves exist but it is an open question whether certain sets of voters exist that would bring the sequence of valid moves to an end.

7.2 Implications of the findings

The voting dynamics that converges to a stable outcome in manipulative voting setting is interesting and relevent to AI as it tackle the fundamental problem of multi-agent decision making, where autonomous agents have to choose a joint plan of action. The study of dynamics in strategic voting helps to understand, control and design multi-agent decision making processes. Our work helps to develop analytical tools that are needed for this topic. This study is a necessary first step to help in developing tools that could help design such processes. The methods introduced can be extended to other situations. A similar process of iterative voting can be seen, "in action", online at various websites used to agree on a date for an event, such as www.doodle.com; following an inital vote, every participant can change his vote. If each participant can change his choice one at a time, this shows voting dynamics are more suited to a relatively small number of players, or an especially close election. An example is multi-agent resource allocation problems e.g, some work on multi-agent system has focussed on negotiation scenarios where agents approach a solution in small steps rather than computing the best solution in one go. The allocations of resources emerge as the result of a sequence of local negotiation steps. The objective of the negotiation is to find a feasible allocation [16, 55]. We use the analytical means to charaterize situations in which we can expect to see a convergence. This model can be regarded as a very simplistic model of a negotiation process amongst the voters, and we like to ensure that it does not end in deadlock.

7.3 Suggestions for further research

These results, in conjunction with the ones of [40] provide quite a complete knowledge of what combinations of types of manipulation move can result in cycles. In the cases where cycles cannot occur, we also obtain polynomial bounds on the lengths of sequences of manipulations. Meir et al. [40] have studied the dynamic process of making manipulations arising from tie-breaking rule for Plurality voting. We note that for alternative voting rules, we have some preliminary results that suggest that it is generally easier to find cycles (those examples require just 2 voters that change their reported preferences). We have exhibited cycles for other voting rules like Veto, Borda, k-Approval, Copeland, Bucklin and Plurality with runoff to show that sequences of moves may be infinite. However, alternative valid moves exist which would bring the sequence to an end. It is an open question whether certain sets of voters exist for which infinite sequences of valid manipulations are unavoidable. If a cycle exists in a transition system, one may still be able to reach a Nash equilibrium by choosing transitions that leave it. In cases where cycles exist the question arises: Can we leave the cycle by choosing the correct transitions? Where the answer is yes, one could ask further whether random choices are likely to find an equilibrium in a short sequence of steps.

In cases where polynomial bounds have been obtained that depends on parameters K and $w_{\max}$, there remains the possibility that polynomial bounds exist that do not depend on those parameters, but just on the number of candidates m and the number of voters n. We can ask: can there be better bounds that are polynomial in terms of m and n?

Example 19 and examples in [40] indicate that one might alternatively want to consider relaxing the assumption of worst-case selection of manipulation move, and show that where cycles exist, it is still possible to reach an equilibrium after a reasonably small number of steps. One reasonable question to investigate is the possible convergence of randomly-selected manipulations.

In cases where we have not shown that a system terminates, one could look for a weaker termination result using chosen manipulation moves. Questions that remain are: Does there always exist a sequence of manipulations ending at Nash equilibrium, if we start at truthful votes? Does there always exist one starting from any declared votes? If manipulations are chosen at random, could we bound the convergence rate?

A rich range of results have emerged in other game-theoretic contexts. We have fast convergence to approximate equilibria in congestion games that require exponentially-long paths to reach exact equilibria [8]. In the context of matching markets it is found that simple local search heuristics may be stuck in a cycle for exponentially many steps, even when there are short paths to Nash equilibria [1].

We did not include any result for lower bounds because we did not find any that could be considered "surprisingly" long (more than linear in the parameters). Linear bounds would not be very interesting.

Bibliography

[1] H. Ackermann, P. W. Goldberg, V. S. Mirrokni, H. Rglin and B. Vcking. **Uncoordinated Two-Sided Matching Markets.** *SIAM Journal on Computing*, 92–106, 2011.

[2] S. Airiau and U. Endriss. **Iterated Majority Voting.** *Proceedings of the 1st International Conference on Algorithmic Decision Theory (ADT)*, 2009.

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

[4] 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.

[5] J. J. Bartholdi III, and J. B. Orlin. **Single transferable vote resists strategic voting.** *Social Choice and Welfare*, 8(4): 341–354, 1991.

[6] A. Bassi. **Voting Systems and Strategic Manipulation: an Experimental Study.** *Paper presented at the annual meeting of the The Midwest Political Science Association, Palmer House Hilton, Chicago, Illinois*, 2006.

[7] V. Bilo. **On the packing of selfish items.** *Procs. of the 20th Internation Parallel and Distributed Processing Symposium (IPDPS)*, 25-29, 2006.

[8] S. Chien, and A. Sinclair. **Convergence to approximate Nash equilibria in congestion games.** *Proceedings of the 18th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, January 2007.

[9] S. Chopra, E. Pacuit, and R. Parikh. **Knowledge-theoretic properties of strategic voting.** *Presented in JELIA-04, Lisbon, Portugal*, 2004.

[10] V. Conitzer, T. Sandholm, and J. Lang. **How many candidates are needed to make elections hard to manipulate?.** *Proceedings of the 9th conference on Theoretical aspects of rationality and knowledge, ACM Press*, 201–214, July 2003.

[11] V. Conitzer, T. Sandholm, and J. Lang. **When are elections with few candidates hard to manipulate?.** *Journal of the ACM*, 54:1–33, 2007.

[12] V. Conitzer, and T. Sandholm. **Complexity of Manipulating Elections with Few Candidates.** *Proceedings of the 18th National Conference on Artificial Intelligence, AAAI Press*, 314–319, July/August 2002.

[13] J. Davies, G. Katsirelos, N. Narodytska, and T. Walsh. **An empirical study of Borda manipulation.** *In COMSOC10*, 91–102, 2010.

[14] Y. Desmedt, and E. Elkind. **Equilibria of plurality voting with abstentions.** *In Proceedings of EC-10*, 347–356, June 2010.

[15] A. Dhillon, and B. Lockwood. **When are plurality rule voting games dominance solvable?.** *Games and Economic Behavior*, 46: 55–75, 2004.

[16] U. Endriss, N. Maudet, F. Sadri and F. Toni. **Negotiating Socially Optimal Allocations of Resources.** *Journal of Artificial Intelligence Research*, 25: 15–348, 2006.

[17] E. Ephrati and Jeffrey. S. Rosenschein. **Deriving consensus in multiagent systems.** *Artificial Intelligence*, 87(1-2): 21–74. 1996.

[18] E. Ephrati and Jeffrey. S. Rosenschein. **A heuristic technique for multiagent planning.** *Annals of Mathematics and Artificial Intelligence*, 13–67, 1997.

[19] E. Ephrati and Jeffrey. S. Rosenschein. **The Clarke tax as a consensus mechanism among automated agents.** *In Proceedings of the 9th National Conference on Artificial Intelligence (AAAI Press)*, 173-178, 1991.

[20] E. EPHRATI AND JEFFREY. S. ROSENSCHEIN. **Multi-agent planning as a dynamic search for social consensus.** *In Proceedings of the 13th International Joint Conference on Artificial Intelligence*, 423–429, 1993.

[21] E. EVEN-DAR, A. KESSELMAN, AND Y. MANSOUR. **Convergence Time to Nash Equilibria.** In *Proceedings of 30th International Colloquium on Automata, Languages, and Programming (ICALP)*, 2003.

[22] A. FABRIKANT, C. PAPADIMITRIOU AND K. TALWAR. **The complexity of pure Nash equilibria.** *In Proceedings of the 36th Annual ACM Symposium on the Theory of Compting (STOC)*, 2004.

[23] R. FAGIN, R. KUMAR, AND D. SIVAKUMAR. **Efficient similarity search and classification via rank aggregation.** *In Proceedings of the 2003 ACM SIGMOD International Conference on Management of Data*, 301–312, June 2003.

[24] P. FALISZEWSKI, AND A. PROCACCIA. **AIs war on manipulation: are we winning?.** *In AI Magazine*, 31(4): 53–64, 2010.

[25] P. FALISZEWSKI, E. HEMASPAANDRA, AND LANE. A. HEMASPAANDRA. **Using Complexity to Protect Elections.** *Communications of the ACM*, 53(11), 74-82, 2010.

[26] R. FARQUHARSON. **Theory of Voting.** *Yale University Press, New Haven*, 1969.

[27] TIMOTHY J. FEDDERSEN, I. SENED, AND STEPHEN G. WRIGHT. **Rational voting and candidate entry under plurality rule.** *American Journal of Political Science*, 34(4): 1005–1016, 1990.

[28] R. FELDMANN, M. GAIRING, T. LUCKING, B. MONIEN AND M. RODE. **Nashification and the Coordination Ratio for a Selfish Routing Game.** *Proceedings of the 30th International Colloquium on Automata, Language, and Programming (ICALP '03), LNCS 2719, Springer*, 514-526, 2003.

[29] R. FORSYTHE, T. RIETZ, R. MYERSON AND R. WEBER. **An experimental work of voting rules and polls in three-candidate elections.** *International Journal of Game Theory*, 25(3): 355–383, 1996.

[30] D. FOTAKIS, S. KONTOGIANNIS, E. KOUTSOUPIAS, M. MAVRONICOLAS AND P. SPIRAKIS. **The structure and complexity of Nash Equilibria for a selfish routing game.** *Procs. of 29th International Colloquium on Automata, Languages, and Programming (ICALP), Malaga, Spain*, 123–134, 2002.

[31] D. FOTAKIS, S. KONTOGIANNIS, AND P. SPIRAKIS. **Selfish Unsplittable Flows.** *31st International Colloquium on Automata, Languages and Programming (ICALP04)*, 593–605, 2004.

[32] S. GHOSH, M. MUNDHE, K. HERNANDEZ AND S. SEN. **Voting for movies: The anatomy of a recommender system.** *Proceedings of the 3rd Annual Conference on Autonomous Agents (AGENTS)*, 434–435, 1999.

[33] A. GIBBARD. **Manipulation of voting schemes.** *Econometrica*, 41(4): 587–601, 1973.

[34] P. W. GOLDBERG. **Bounds for the Convergence Rate of Randomized Local Search in a Multiplayer Load-balancing Game.** *Procs of the 23rd Annual ACM SIGACT-SIGOPS Symp. on Principles of Distributed Computing (PODC)*, 131–140, July '04.

[35] T. HAYNES, S. SEN, N. ARORA AND R. NADELLA. **An automated meeting scheduling system that utilizes user preferences.** *Proceedings of the 1st Annual Conference on Autonomous Agents (AGENTS)*, 308–315, 1997.

[36] E. HEMASPAANDRA AND L. A. HEMASPAANDRA. **Dichotomy for voting systems.** *Journal of Computer and System Sciences*, 73–83, 2007.

[37] M. J. HINICH, J. O. LEDYARD, AND P. C. ORDESHOOK. **Nonvoting and the existence of equilibrium under majority rule.** *Journal of Economic Theory*, 4(2): 144–153, 1972.

[38] J. J. LAFFONT. **Incentives and the allocation of public goods.** *In Auerbach, A. J., and Feldstein, M., eds., Handbook of Public Economics*, chapter 10, 537–569, 1987.

[39] O. LEV AND J. S. ROSENSCHEIN. Tie-breaking limits on convergence of iterative voting. *Bar-Ilan Symposium on Foundations of Artificial Intelligence*, 2011.

[40] R. MEIR, M. POLUKAROV, J. S. ROSENSCHEIN AND N. R. JENNINGS. Convergence to Equilibria in Plurality Voting. *Proceedings of 24th Conference on Artificial Intelligence (AAAI-10)*, 2010.

[41] M. MESSNER, AND M. K. POLBORN. Strong and coalition-proof political equilibria under plurality and runoff rule. *International Journal of Game Theory*, 35(2): 287–314, 2007.

[42] FLAVIO K. MIYAZAWA AND ANDRE L. VIGNATTI. Convergence Time to Nash Equilibrium in Selfish Bin Packing?. *International Journal of Foundations of Computer Science*, 22: 565–582, 2011.

[43] D. MONDERER AND L.S. SHAPLEY. Potential Games. *Games and Economic Behavior*, 14, 124-143, 1996.

[44] ROGER B. MYERSON, AND ROBERT J. WEBER. A theory of voting equilibria. *The American Political Science Review*, 87(1): 102–114, 1993.

[45] R. G. NIEMI AND H. RIKER. The choice of voting systems. *Sci Am*, 234: 21–27, 1976.

[46] E. M. S. NIOU. Strategic Voting under Plurality Rule: When Does It Matter and Who should Vote Strategically?.

[47] H. NURMI. Voting procedures: a summary analysis. *Br J Polit Sci*, I3 : 181–208, 1983.

[48] S. OBRAZTSOVA, E. ELKIND, N. HAZON. Ties matter: complexity of voting manipulation revisited. *Twenty-second International Joint conference on artificial Intelligence*, July 2011.

[49] P. N. PANAGOPOULOU, AND P. G. SPIRAKIS. Efficient Convergence to Pure Nash Equilibria in Weighted Network Congestion Games. *In Proceedings of the 4th International Workshop on Efficient and Experimental Algorithms (WEA)*, 2005.

[50] CHRISTOS. H. PAPADIMITRIOU. Algorithms, games, and the internet. *In Proc. of STOC 2001*, 749–753, 2001.

[51] D. PENNOCK, E. HORVITZ, AND C. GILES. Social choice theory and recommender systems: Analysis of the axiomatic foundations of collaborative filtering. *In Proceedings of the 17th National Conference on Artificial Intelligence*, 729–734, 2000.

[52] H. PEYTON. YOUNG. Social choice scoring functions. *SIAM Journal of Applied Mathematics*, 1975.

[53] ARIEL. D. PROCACCIA, A. ZOHAR, Y. PELEG AND JEFFREY. S. ROSENSCHEIN. The learnability of voting rules. *Artificial Intelligence, In press*, 2009.

[54] ARIEL. D. PROCACCIA AND JEFFREY. S. ROSENSCHEIN. Junta distributions and the average-case complexity of manipulating elections. *Journal of Artificial Intelligence Research*, 157-181, 2007.

[55] JEFFREY S. ROSENSCHEIN AND G. ZLOTKIN. Rules of Encounter. *MIT Press, Cambridge*, 1994.

[56] ROBERT. W. ROSENTHAL. A Class of Games Possessing Pure-Strategy Nash Equilibria. *International Journal of Game Theory*, 65–67, 1973.

[57] DONALD. G. SAARI. Susceptibility to manipulation. *Public Choice*, 64: 21–41, 1990.

[58] M. SATTERTHWAITE. Strategy-proofness and Arrows conditions: Existence and correspondence theorems for voting procedures and social welfare functions. *Journal of Economic Theory*, 10(2):187–217, 1975.

[59] P. DE TRENQUALYE. An extension of bowens dynamic voting rule to many dimensions. *Social Choice and Welfare*, 15(1): 141–159, 1998.

[60] T. WALSH. Where are the really hard manipulation problems? The phase transition in manipulating the Veto rule. *In IJCAI09*, 324-329, 2009.

[61] L. XIA, M. ZUCKERMAN, A. PROCACCIA, V. CONITZER, AND J. ROSENSCHEIN. **Complexity of unweighted coalitional manipulation under some common voting rules.** *In IJCAI09*, 348–352, 2009.

[62] L. XIA AND V. CONITZER. **Stackelberg voting games: Computational aspects and paradoxes.** *In Proceedings of AAAI-10*, 921–926, July 2010.

[63] L. XIA, V. CONITZER, AND A. PROCACCIA. **A scheduling approach to coalitional manipulation.** *In EC10*, 275–284, 2010.