# NP-Hard Manipulations of Voting Schemes

**Elizabeth Cross**

December 9, 2005

## 1 Introduction

Voting schemes are common social choice function that allow voters to aggregate their preferences in a socially desirable way. Unfortunately, the Gibbard-Satterthwaite theorem states that many of these schemes are susceptible to manipulation. Thus, a highly desirable quality, such that a scheme is resistant to manipulaion, is lacking in voting schemes. Since non-manipulability cannot be guaranteed, researchers have investigated other ways to discourage manipulation. This research has led in the direction of voting schemes which are difficult to manipulate. The investigation of these schemes has greater importance in the area of multiagent systems. In this case, voters are computer agents who can determine strategic voting schemes much quicker than their human counterparts. Thus, having voting schemes which are computationally difficult to manipulate is a desirable quality in these settings.

### 1.1 Voting Schemes

A Voting Scheme which is given a set of candidates and a transitive preference ordering of the candidates from each voter will determine a subset of the candidates who are the winners of the vote. This subset of candidates should best represent the collective preferences of all voters.

Several voting schemes that are relevant to the discussions of this paper are: plurality, plurality with runoff, randomized cup, regular cup, Copeland, Borda, maximin, veto and single transferable vote (STV).

### 1.2 Manipulation

A voting scheme is manipulable if a voter can misrepresent his preference ordering to obtain a final preference ordering which is more desirable to himself [3]. 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. As voting schemes are striving for a socially desirable result, manipulability is not a desirable quality. However, voters will always misrepresent their preferences if it is in their own interest, therefore schemes should strive to be un-manipulable. Unfortunately, the Gibbard-Satterthwaite theorem states that a non-dictatorial voting scheme with more than three candidates is susceptible to manipulation [3]. This is a discouraging statement because in order to be fair the scheme should be both non-dictatorial and non-manipulable.

There are several different alternatives when it comes to the manipulation of voting schemes. The first alternative is an incomplete information setting versus a complete information setting. In an incomplete information setting the manipulators have little knowledge of the non-manipulators' votes. Conversely, in the complete information setting the manipulators are aware of all the non-manipulators' votes. A vote can be manipulated by a single individual voter or by a coalition (group) of voters. Voters can have weights associated with their vote or be unweighted. A weight can be considered as the collective votes for a group of size k. Constructive manipulation is when a voter is trying make a candidate win the election. Destructive manipulation is when a voter is trying to cause a candidate not to win the election [2].

## 1.3 Multiagent Systems

Multiagent systems are computer systems in which the decision makers are software agents. These systems are becoming more prominent in society for various purposes such as rank aggregation, recommender systems, planning among automated agents and decision makers in a AI context [2]. When considering the manipulation of voting schemes, the distinction between software agent voters and human voters is especially important. Software agents can be programmed to execute rational, strategic voting decisions which are not skewed by the irrationality or emotions that humans can demonstrate. Software agents are able to consider a large possibility of voting options in a shorter amount of time than a human, thus voting schemes need a greater resistance to manipulability when software agents are voting. Fortunately, software agents in multiagent systems usually need to make their voting decisions in real time, which gives researchers a headstart when developing schemes which are difficult for the agents to manipulate.

## 1.4 Computational Complexity

The headstart that researchers get by assuming software agents must make their voting decisions in real time enables them to develop voting schemes that are computationally complex to manipulate. Thus, the NP-completeness of a manipulation has become a desirable quality for a voting scheme. Thus in the worst case, software agents would have to perform NP-hard computations to determine a vote ordering which could manipulate the election in their favour.

## 2 Voting Schemes which are Difficult to Manipulate

Preliminary research in this area dealt with proving that schemes are NP-complete for an unbounded number of voters and candidates. One interesting result that was proved by Bartholdi, Tovey and Trick is that Copeland's method is easily manipulated but 2nd order Copeland's scheme is NP-complete to manipulate. This led to proving that the manipulation of 1st order Copeland with 2nd order tie breaks is NP-complete [1].

Research in this area resurfaced in the new millenium when Conitzer and Sandholm identified a possible usage for difficult manipulations in the area of multiagent systems. They extended previous research by examining the complexity of manipulating schemes with a smaller number of candidates. In a series of two papers [2][3], they found results for nine schemes in the cases of constructive (Table 1) and destructive (Table 2) manipulation. The nine schemes, shown in the tables, were either manipulated in polynomial time (P) or NP-complete (NP-c) to manipulate for the specified number of candidates. The results presented in the two tables deal with the case of a coalitional weighted manipulation scheme in the complete information setting. This choice of manipulation is made for the following reasons:

- Complete information setting was chosen because it is a special case of any uncertainty model. Thus hardness results will apply to the incomplete information case. Also the incomplete information case could add an extra layer of uncertainty that is not present in the complete information case.
- Coalition manipulation was chosen because in an election with many voters it is unlikely that a single individual could manipulate the results with their vote. Also hardness results for a manipulation by a coalition in a complete information setting can be used to prove hardness results for an individual coalition in the incomplete setting.
- Weighted votes were chosen because hardness results for a manipulation by a weighted coalition in a complete information setting can be used to prove hardness results for a manipulation by unweighted (but correlated) voters in the incomplete information case.

In order to prove NP-hard results for the majority of these schemes a reduction was shown from the PARTITION problem to a constructive or destructive coalitional weighted manipulation for a certain number of candidates. These proofs are not trivial.

Table 1: Complexity Results of Constructive Manipulations [2]

| Number of candidates | 2 | 3 | $4, 5, 6$ | $\geq 7$ |
|---|---|---|---|---|
| Borda | P | NP-c | NP-c | NP-c |
| veto | P | NP-c | NP-c | NP-c |
| STV | P | NP-c | NP-c | NP-c |
| plurality with runoff | P | NP-c | NP-c | NP-c |
| Copeland | P | P | NP-c | NP-c |
| maximin | P | P | NP-c | NP-c |
| randomized cup | P | P | P | NP-c |
| regular cup | P | P | P | P |
| plurality | P | P | P | P |

Table 2: Complexity Results of Destructive Manipulations [2]

| Number of candidates | 2 | $\geq 3$ |
|---|---|---|
| STV | P | NP-c |
| plurality with runoff | P | NP-c |
| randomized cup | P | ? |
| Borda | P | P |
| veto | P | P |
| Copeland | P | P |
| maximin | P | P |
| regular cup | P | P |
| plurality | P | P |

# 3 Creation of Schemes which are Difficult to Manipulate

The results of the previous section are somewhat promising, because some common voting schemes were shown to be NP-complete to manipulate. It would also be promising to develop new schemes which were difficult to manipulate. Striving in this direction researchers have investigated making small changes to existing schemes in order to guarantee the complexity of manipulation. This small change in the protocol involves adding a preround before the actual voting scheme is run. The first investigations of this sort involved adding a cup preround to several schemes which are know to be polynomial in manipulation [4]. In this modified protocol the candidates are paired and the losing candidate in the pairwise election is eliminated. In the case where there is an odd number of candidates, one of the candidates will get a bye. The original protocol is then run by adhering to the implicit votes of the remaining candidates.

The type of preround used determines the complexity results of the fabricated scheme. There are three types of prerounds: deterministic, randomize and interleaved. A deterministic preround involves deciding and publishing the pairwise schedule before the votes are collected. A randomized preround involves drawing a random schedule after the votes have been collected. Finally, an interleaved preround involves eliciting the votes incrementally and interleaving this elicitation with a random scheduling process.

The question that the constructive manipulator must ask is slightly different in each preround type. In a deterministic preround, the manipulator is trying to determine if they can cast their vote in order to make a certain candidate win. In the randomized and interleaved prerounds, the manipulator is trying to determine if they can cast their vote so that the probability of their preferred candidate winning is larger than a certain value. The results of this small change were positive in all types of prerounds. When a deterministic preround precedes the plurality, Borda, maximin and STV schemes, the constructive manipulation of the scheme is proved to be NP-complete. Similarly by adding a randomized preround the four schemes are shown to be #P-hard to manipulate and by adding a interleaved preround to the four schemes PSPACE-complete manipulation is achieved.

The idea of prerounds was generalized by Elkind and Lipmaa who developed a protocol which allows a voting scheme to be prefaced by a series of steps of a different voting scheme [5]. The previously described scheme was a hybrid of one step of the cup scheme followed by either the plurality, Borda, maximin or the STV scheme. The generalized hybrid scheme is Hyb($X_k$, Y). This notation represents executing k steps of X voting scheme followed by running the Y scheme on the remaining candidates. A single step of the scheme X is different for each scheme. In the case of STV, a step consists of eliminating the candidate with the lowest number of votes and transferring the votes for this candidate to the second highest choice of the voter. In the case of binary cup, a step consists of one round of pairwise elimination. Finally, in point based schemes one step consists of computing the scores of each candidate and eliminating the candidate with the lowest score without computing new scores between steps. This hybrid approach led to the development of many new schemes which are NP-hard to manipulate. Here is a list of those schemes:

- Hyb($\text{STV}_k$, Y) and Hyb($X_k$, STV) where X, Y $\in$ Plurality, Borda, Maximin, BC
- Hyb($\text{Borda}_k$, Plurality) and Hyb($\text{Maximin}_k$, Plurality) for infinitely many values of k
- Hyb($\text{Maximin}_k$, Plurality) for infinitely many values of k
- Hyb($\text{Borda}_k$, Borda) for infinitely many values of k
- Hyb($\text{Maximin}_k$, Borda) for infinitely many values of k

Even though this hybridization method creates many NP-hard schemes, it does not work for all combinations of schemes. If the first protocol does not give enough options to the manipulator to make the manipulation complex then the resulting hybrid will not be NP-hard. For example, Hyb($\text{Plurality}_k$, Y) where Y $\in$ Borda, Maximin, Bc, Plurality can be manipulated in polynomial time.

Along with being NP-hard to manipulate, these Hybrid protocols preserve some important qualities of the voting schemes. For example, if both X and Y are Condorcet-consistent then so is Hyb($X_k$, Y). Pareto optimality is also preserved, if X is pareto-optimal(strongly monotone) and Y is pareto-optimal(monotone) then Hyb($X_k$, Y) is pareto optimal(monotone).

# 4 Average Case Analysis

The previous sections all dealt with voting schemes that were difficult to manipulate in the worst case. The problem with this analysis is that it may have little bearing on how the schemes perform on average. If the schemes can be manipulated in polynomial time on average then the worst case analysis has little meaning. Introductory research in this subject has been performed by Procaccia and Rosenschein [6]. In their research they introduce the concept of a Junta distribution, which is a distribution over possible NP-hard manipulation problem instances. The use of a Junta distribution in average case analysis is to examine several of these problematic distributions in representation of all other distributions. A scheme is shown to be susceptible to manipulation on average if a polynomial time algorithm can usually find a manipulation instance in the Junta distribution. In this case a greedy algorithm was developed to demonstrate that sensitive scoring protocols, such as Borda and Veto, are susceptible to manipulation on average.

## 5 Analysis and Discussion

This research area is becoming more prevalent as multiagent systems gain a bigger role in decision making processes. Worst case analysis research has demonstrated that STV is one of the schemes that is the most difficult to manipulate. It also demonstrated that plurality is one of the easiest schemes to manipulate, but it is interesting to note that plurality is still being used widely in practice. Perhaps this is because plurality is a social norm in voting schemes and people have little interest in conforming to a new scheme. The hybrid schemes are the most positive new development in worst case analysis. They conform to the social norm of using plurality, but by tweaking plurality they can guarantee some sort of manipulation difficulty.

Although positive results have been found in the area of worst case analysis, the true research into this field is introductory at best. It is important to note that every researcher has suggested that the focus of hard manipulations should turn to average case analysis rather than worst case analysis. This is because researchers have identified that voting schemes should strive to conform to a stricter criteria than worst case analysis. Instead, these schemes should be able to dissuade manipulations in the average case. Unfortunately, average case analysis is hard to define and thus research into the area is slow to start. The Junta distribution, suggested by Procaccia and Rosenschein, is a new criteria for studying average case complexity, and even the authors of the paper are not completely sure that the distribution is defined accurately. The findings of their paper prove that one type of scheme was easy to solve on average, but there is no proof yet that any scheme is NP-hard to manipulate on average. Further research in the field should be directed towards identifying criteria that make the manipulation of schemes NP-hard on average, proving the existence of these schemes, and creating schemes which satisfy this criteria. The research is also lacking in practical implementation. It would be interesting if the resistancy of different schemes to average case manipulations were compared in practice.

## 6 Conclusion

The manipulation of a voting scheme is an important criteria to consider when deciding on a scheme. In the case of multiagent systems this criteria increases in importance because the voters are sophisticated computer agents who are able to carry out complex computations. In order to achieve some standard of non-manipulability in voting schemes, the complexity of the manipulation is considered. This article reviewed common schemes in terms of coalitional weighted manipulation schemes in the complete information setting. We found that manipulation of many common schemes is NP-complete in the case of constructive manipulation, but most destructive manipulations could be accomplished in polynomial time. The best scheme we found to resist manipulations in the worst case was STV. In order to find other schemes which were difficult to manipulate, hybrid schemes were created which preceded a protocol with several steps of another protocol. Most of these newly created schemes were found to resist manipulations well in the worst case. Unfortunately a common problem was found with the worst case analysis, and new research has started to investigate average case manipulations of voting schemes. This average case analysis is a difficult and complex problem for researchers to address but it is this direction of the field that will lead to the most promising results for deterring manipulations.

# References

[1] John J. Bartholdi, III, Craig A. Tovey, and James B. Orlin, "The Computational Difficulty of Manipulating an Election," *Social Choice and Welfare*, 6:227-241, 1989.

[2] Vincent Conitzer, Jerome Lang, and Tuomas Sandholm, "How many candidates are needed to make elections hard to manipulate?," *In Proceedings of the International Conference on Theoretical Aspects of Reasoning about Knowledge*, pages 201-214, Bloomington, Indiana, 2003.

[3] Vincent Conitzer and Tuomas Sandholm, "Complexity of Manipulating Elections with Few Candidates," *In Proceedings of the National Conference on Artificial Intelligence (AAAI)*, pages 314–319, Edmonton, Canada, 2002.

[4] Vincent Conitzer and Tuomas Sandholm, "Universal Voting Protocol Tweaks to Make Manipulation Hard," *In Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI)*, pages 781-788, Acapulco, Mexico, August 915 2003.

[5] Edith Elkind and Helger Lipmaa, "Hybrid Voting Protocols and Hardness of Manipulation," *The 16th Annual International Symposium on Algorithms and Computation, ISAAC 2005*, volume 3827 of Lecture Notes in Computer Science, pages 206–215, Sanya, Hainan, China, December 19–21, 2005.

[6] Ariel D. Procaccia and Jeffrey S. Rosenschein, "Junta Distributions and the Average-Case Complexity of Manipulating Elections," *The Fifth International Joint Conference on Autonomous Agents and Multiagent Systems*, Hakodate, Japan, May 2006.