Search versus Search for Collapsing Electoral Control Types

Extended Abstract

Benjamin Carleton
Department of Computer Science
Cornell University
Ithaca, USA
carleton@cs.cornell.edu

Michael C. Chavrimootoo
Department of Computer Science
University of Rochester
Rochester, USA
michael.chavrimootoo@rochester.edu

Lane A. Hemaspaandra
Department of Computer Science
University of Rochester
Rochester, USA
www.cs.rochester.edu/u/lane

David E. Narváez
Dept. of Electrical and Computer Eng.
Virginia Tech
Blacksburg, USA
david.narvaez@computer.org

Conor Taliancich
Property Matrix
Culver City, USA
ctalianc@u.rochester.edu

Henry B. Welles
Department of Computer Science
University of Rochester
Rochester, USA
hwelles@u.rochester.edu

ABSTRACT

Hemaspaandra et al. [6] and Carleton et al. [3, 4] found that many pairs of electoral (decision) problems about the same election system coincide as sets (i.e., they are *collapsing pairs*), which had previously gone undetected in the literature. While both members of a collapsing pair certainly have the same decision complexity, there is no guarantee that the associated search problems also have the same complexity. For practical purposes, search problems are more relevant than decision problems.

Our work focuses on exploring the relationships between the *search versions* of collapsing pairs. We do so by giving a framework that relates the complexity of search problems via efficient reductions that transform a solution from one problem to a solution of the other problem on the same input. We not only establish that the known decision collapses carry over to the search model, but also refine our results by determining for the concrete systems plurality, veto, and approval whether collapsing search-problem pairs are polynomial-time computable or NP-hard.

KEYWORDS

Computational Social Choice; Elections; Electoral Control; Problem Interreducibility; Search Complexity; Voting

ACM Reference Format:
Benjamin Carleton, Michael C. Chavrimootoo, Lane A. Hemaspaandra, David E. Narváez, Conor Taliancich, and Henry B. Welles. 2023. Search versus Search for Collapsing Electoral Control Types: Extended Abstract. In *Proc. of the 22nd International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2023), London, United Kingdom, May 29 – June 2, 2023*, IFAAMAS, 3 pages.

1 INTRODUCTION

This work is available as a full technical report [2]. We refer readers to that report for full definitions and complete results.

"Control" attacks on elections try to make a focus candidate win/lose/uniquely-win/not-uniquely-win through such actions as adding, deleting, or partitioning candidates or voters [1, 3, 4, 7]. A result of Hemaspaandra et al. [6], supplemented by an observation of Carleton et al. [3, 4], established that, surprisingly, seven pairs among the 44 (relatively) "standard" control types—for each election system, i.e., each mapping from candidates and votes to a winner set among the candidates—are *equal* as sets (i.e., are collapsing pairs).

Carleton et al. [3, 4] showed that those seven pairs are the only ones that collapse regardless of the election system. However, they discovered some additional collapses that hold specifically for veto or specifically for approval voting, and also found some additional collapses that hold for all election systems that satisfy certain axiomatic properties.

While it is evident that the members of a collapsing pair share the same decision complexity (because the sets are the same), it does not necessarily follow that their *search* complexities are the same. Why is it plausible that sets with the same decision complexity might have different search complexities (relative to some certificate/solution schemes)? Well, it indeed can and does happen if $\mathrm{P} \neq \mathrm{NP} \cap \mathrm{coNP}$ (and so certainly happens if integer factorization is not in polynomial time, since the natural decision version of that is in $\mathrm{NP} \cap \mathrm{coNP}$); indeed, although we draw on Hemaspaandra et al. [6]'s paper primarily for the collapses it proves, one of that paper's central results is that if $\mathrm{P} \neq \mathrm{NP} \cap \mathrm{coNP}$, then in some sense search "separates" from decision for many electoral manipulation problems (see also page 1 of [2]).

Our work shows that the collapsing (as decision problems) pairs of Hemaspaandra et al. [6] and Carleton et al. [3, 4] always have the same *search* complexity (given access to the winner problem).

Why is this important? In reality, one typically—e.g., if one is a campaign manager—wants not merely to efficiently compute whether there *exists* some action that will make one's candidate win, but rather one wants to get one's hands, efficiently, on *an actual such successful action*. Our results are establishing that the literature's existing collapsing control type pairs, though each in the literature is about and proved for the "exists" case, in fact have the property that (given access to the winner problem for the election system in question, although for the three concrete election systems we cover as examples that is not even needed as their winner problems are each in polynomial time) for both members of the pair the "getting one's hands on a successful action when one exists" issue is also of

Proc. of the 22nd International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2023), A. Ricci, W. Yeoh, N. Agmon, B. An (eds.), May 29 – June 2, 2023, London, United Kingdom.

the same complexity for both (and one can indeed usually efficiently use access to solutions for one to get solutions for the other).

2 OVERVIEW OF APPROACH AND RESULTS

In our quest to show that known collapsing electoral control types also have polynomially related search complexity (relative to the winner problem of the election system they are about), we need a notion of reductions between search problems. Fortunately, Megiddo and Papadimitriou defined a reduction notion between search problems that is close to what we need. Megiddo and Papadimitriou [10] say that a reduction from problem Π_R to problem Π_S is a pair of polynomial-time computable functions f and g such that, for any $x \in \Sigma^*$, $(x, g(y)) \in R \iff (f(x), y) \in S$. This in spirit is trying to say that we can map via f to an instance $f(x)$ such that given a solution relative to S of $f(x)$ we can via g map to a solution relative to R of x. Unfortunately, read as written, it does not seem to *do* that, regardless of whether one takes the omitted quantification over y to be existential or to be universal. Either way the definition leaves open a loophole in which on some x for which there *does* exist a solution relative to R, the value of $f(x)$ will be some string that has no solution relative to S, and all $g(y)$'s will be strings that are not solutions to x relative to R. So the "$\iff$" will be satisfied since both sides evaluate to False, but no solution transfer will have occurred. In the nightmare case, a given "reduction" could exploit this loophole on *every* x that has a solution relative to R.

In giving our framework, we carefully patch the issue above and modify that approach to better suit our study. In practice, what we want to know is that if two types $\mathcal{T}_1$ and $\mathcal{T}_2$ decision-collapse, whether a search algorithm for one of the two problems yields a search algorithm for the other. In such a setting, it does not make sense to fully adopt Megiddo and Papadimitriou [10]'s approach as (assuming one fixes the above loophole first) the function $f(x)$ allows the solution to x on the R side to be obtained via demanding a solution to a *different* (than x) instance $f(x)$, on the S side. But in our setting, collapsing types are the *same* set, just with differing "witnessing" relations. And our goal is to make connections via those witnesses. So in our definitions, we require their $f(x)$ to be the identity function! Second, since we wish to connect the solutions of collapsing pairs even when the winner problem of $\mathcal{E}$ is not in P, our reductions will have the winner problem as an oracle.

That access to the winner-problem oracle often seems quite important (e.g., see [2, Footnote 4]). But on the other hand, we find some cases where, even when an election system's winner problem is not polynomial-time computable, two collapsing control types about that election system are polynomially search-equivalent (i.e., with the reductions witnessing the equivalence never using the oracle); we return to this at the end of this section.

Before we introduce our definition, let us first give some intuition behind it. "$\mathcal{T}_1$ is polynomially search reducible to $\mathcal{T}_2$" means that $\mathcal{T}_2$'s solutions are so powerful that for problem instance I, given any solution for $\mathcal{T}_2$ with respect to I one can quickly build a solution to $\mathcal{T}_1$ with respect to I.

Definition 1 (Informal; see [2, part 1 of Definition 2.3] for full details). *For an election system $\mathcal{E}$ and two control types $\mathcal{T}_1$ and $\mathcal{T}_2$ that are both about $\mathcal{E}$ and have the same input types, we say that "$\mathcal{T}_1$ is polynomially search-reducible to $\mathcal{T}_2$" exactly if there is a reduction that runs in polynomial time and on each input (I, S), where I is an input to $\mathcal{T}_1$ and S is a solution for I with respect to $\mathcal{T}_2$, outputs a solution S' for I with respect to $\mathcal{T}_1$.*

Additionally, if $\mathcal{T}_1$ is polynomially search-reducible to $\mathcal{T}_2$ and $\mathcal{T}_2$ is polynomially search-reducible to $\mathcal{T}_1$, then we say that "$\mathcal{T}_1$ is polynomially search equivalent to $\mathcal{T}_2$."

Our full paper also defines the analogous notions for the case when we are given access to $\mathcal{E}$'s winner problem (which might not be in P).

Our paper proves that in every case under consideration, the decision collapses hold in the search model (if given access, for the case where the election's winner problem itself is not even in P, to an oracle for the election's winner problem), and indeed we show that in each case a solution for one can (again, given access to the winner problem) be polynomial-time transformed into a solution for the other. Thus the complexities are polynomially related (given access to the election system's winner problem).

Additionally, for the concrete cases of plurality's, veto's, and approval's collapsing pairs, we explore whether those polynomially equivalent search complexities are clearly "polynomial time," or are NP-hard, and we resolve every such case. In doing so, we establish new decision-complexities that were not previously proven in the literature. For example, we show that approval is immune with respect to destructive control by partition (and run-off partition) of candidates under the ties-promote and nonunique-winner models, and from that conclude that the two decision problems are in P. Similarly, we prove the analogous decision problems under plurality are in fact NP-complete. Thus we not only establish new hardness results, but we also give many new polynomial-time computable algorithms for the search problems under consideration.

Finally, as mentioned earlier, our results show rather surprisingly tight connections between search problems, namely, when dealing with collapses that hold for election systems satisfying the unique version of the Weak Axiom of Revealed Preferences (whose definition we omit here due to space constraints), we show polynomial search equivalence without ever using the oracle given to us (even though those election systems can, and indeed do, have winner problems that are not polynomial-time computable), see [2, parts 2 and 3 of Corollary 4.5]

3 CONCLUSIONS AND OPEN DIRECTIONS

Besides providing the first framework to relate the search complexities of decision-collapsing electoral control types, our paper both provides many algorithms to either compute witnesses to successful cases of electoral control or to compute those witnesses from the witnesses of related problems, and proves new hardness results in the decision (and, via a gateway we provide as [2, Theorem 4.9], the search) model.

An interesting open direction would be to seek more general results, such as dichotomy theorems covering broad collections of election systems. However, that may be difficult since not much is known as to dichotomy theorems even for the decision cases of (unweighted) control problems (however, see [5, 8, 9]), though the few known such cases would be natural starting points to look at in this regard.

ACKNOWLEDGMENTS

This work was supported in part by NSF grant CCF-2006496 and CIFellows grant CIF2020-UR-36.

REFERENCES

[1] John J. Bartholdi, III, Craig A. Tovey, and Michael A. Trick. 1992. How Hard is it to Control an Election? *Mathematical and Computer Modeling* 16, 8–9 (1992), 27–40.

[2] Benjamin Carleton, Michael C. Chavrimootoo, Lane A. Hemaspaandra, David E. Narváez, Conor Taliancich, and Henry B. Welles. 2022. *Search versus Search for Collapsing Electoral Control Types.* Technical Report arXiv:2207.03049 [cs.GT]. Computing Research Repository, arXiv.org/corr/. Revised, October 2022.

[3] Benjamin Carleton, Michael C. Chavrimootoo, Lane A. Hemaspaandra, David E. Narváez, Conor Taliancich, and Henry B. Welles. 2022. *Separating and Collapsing Electoral Control Types.* Technical Report arXiv:2207.00710 [cs.MA]. Computing Research Repository, arXiv.org/corr/. Revised, February 2023.

[4] Benjamin Carleton, Michael C. Chavrimootoo, Lane A. Hemaspaandra, David E. Narváez, Conor Taliancich, and Henry B. Welles. 2023. Separating and Collapsing Electoral Control Types. *In these proceedings.*

[5] Piotr Faliszewski, Edith Hemaspaandra, and Lane A. Hemaspaandra. 2015. Weighted Electoral Control. *Journal of Artificial Intelligence Research* 52 (2015), 507–542.

[6] Edith Hemaspaandra, Lane A. Hemaspaandra, and Curtis Menton. 2020. Search versus Decision for Election Manipulation Problems. *ACM Transactions on Computation Theory* 12, #1, Article 3 (2020), 1–42.

[7] Edith Hemaspaandra, Lane A. Hemaspaandra, and Jörg Rothe. 2007. Anyone but Him: The Complexity of Precluding an Alternative. *Artificial Intelligence* 171, 5–6 (2007), 255–285.

[8] Edith Hemaspaandra, Lane A. Hemaspaandra, and Henning Schnoor. 2014. A Control Dichotomy for Pure Scoring Rules. In *Proceedings of the 28th AAAI Conference on Artificial Intelligence.* AAAI Press, 712–720.

[9] Edith Hemaspaandra and Henning Schnoor. 2016. Dichotomy for Pure Scoring Rules Under Manipulative Electoral Actions. In *Proceedings of the 22nd European Conference on Artificial Intelligence.* IOS Press, 1071–1079.

[10] Nimrod Megiddo and Christos H. Papadimitriou. 1991. On Total Functions, Existence Theorems and Computational Complexity. *Theoretical Computer Science* 81, 2 (1991), 317–324.