Myopic decision tree construction begins at with a single root node associated with initial training set $E_0 = \mathcal{A}$. At each iteration, any "unprocessed" node n is evaluated and becomes processed: if n is pure, it becomes a leaf in the (final) tree. Otherwise, the gain $IG(E(n)|q)$ of each (nonredundant) query q is evaluated, and the query with greatest gain is applied to n, creating two new children of n, with the appropriate edge labels, and each associated with the positive (resp., negative) examples from $E(n)$. When no nodes remain unprocessed, all leaves in the tree are pure.

Processing a node n is linear in the number of training examples at n (at most $|\mathcal{A}|$) and the number of splits being evaluated (at most $|U|$). Hence the complexity of myopic induction is linear in (a possibly pruned) $\mathcal{A}$ and the size of the resulting tree. Since $\mathcal{A}$ has size $2^{|U|}$ in the worst case (if unpruned), complexity is $O(2^{|U|})$ for trees of bounded size, i.e., significantly more efficient than DP. The myopic method is not guaranteed to produce a policy with minimal expected query cost. However, it works well in practice (see below); and it is guaranteed to provide a *correct* policy that determines the true winner.

Various forms of policy approximation can be used if we are willing to admit the possibility of error in declaring a winner. One approximation allows terminating the querying process at *impure leaves*, requiring only that we be "sure enough" about its identity to allow winner prediction despite residual uncertainty. This is analogous to cost-sensitive classification (Greiner, et al. 1992; Turney 1995), where both tests and *prediction errors* have costs. In our model, we have two types of misclassification errors: (a) if we choose a winner who turns out to be unavailable; (b) if we choose a winner that is available, but is not the true winner given the actual (unknown) available set. In general, we expect the former to be much worse than the latter. This can be implemented in both DP and the myopic algorithm. In the latter, we simply stop splitting leaves when one winner has sufficiently high probability.

Another approximation uses of *sampled availability sets*, with examples A drawn from the distribution P over $\mathcal{A}$, thereby reducing the number of training examples to make myopic tree construction fully tractable. Sample complexity results then become vital (Greiner, et al. 1992). We leave these approximations to future research.

Query Complexity

Apart from optimizing query policies, the theoretical question of both worst-case and average-case query complexity is of interest. Here we sketch some partial results that suggest the types of questions one might ask in our model.

Worst-case results take the form: given a voting rule r and availability distribution P, what is the greatest (over vote profiles $\mathbf{v}$) expected (over availabilities) query cost of the optimal query policy? If availability is highly likely, we can construct profiles where determining the winner requires almost m queries in expectation, for both plurality and Borda. For plurality, consider candidates $X = \{c_1, \ldots, c_{2m}\} \cup \{x, y\}$, and known available set $Y = \{x, y\}$. Define a profile over $2m$ voters where x and y are each ranked second in exactly half of the rankings: $c_i \succ x \succ \ldots$ for voters $i = 1, \ldots, m$, and $c_i \succ y \succ \ldots$ for $i = m+1, \ldots, 2m$ (other candidates are ordered arbitrarily). Let $p = 0.5$. The plurality score of x is the number of candidates in $\{c_i\}_{i=1}^{m}$ that are unavailable (similarly for y). As m increases, one of x or y wins with high probability. However, one can show (due to concentration of the binomial distribution) that the difference in their plurality scores becomes sub-linear, with high probability, as m grows. Hence, $\Omega(m)$ queries are needed to determine the winner. Similar constructions work for Borda and Copeland. As a result, we have:

Proposition 6 *For plurality, Borda and Copeland, worst-case (over profiles) expected query complexity for determining a robust winner is* $\Omega(m)$.

One can also analyze expected optimal query cost for vote profiles drawn from particular distributions (e.g., impartial culture, Mallows models, mixtures, etc.). As availability probabilities approach 1 (i.e., unavailability is rare), constructing optimal policies becomes easier, as does analysis of query complexity. Assume $p_x = p = 1 - O(\varepsilon)$ for all x and all query costs are identical. The query policy $Extreme(\mathbf{v})$ (informally) proceeds as follows: initialize the *potential set* X with all candidates, the *known set* $Y = \emptyset$, and the *current winner* $w = r(\mathbf{v}(X)) = r(\mathbf{v})$. Then repeat:

1. find a minimum-size subset Z of $X \setminus Y$ s.t. w is a robust winner for $Y \cup Z$ ($w \in Z$ if w is not known to be available);
2. check availability of all candidates in Z; add to Y those that are available, and remove from X those that are not;
3. if all candidates in Z are available, stop and output w;
4. if not, recompute $w = r(\mathbf{v}(X))$, and go back to step 1.

It is not hard to show that $Extreme(\mathbf{v})$ terminates, and returns the true winner $r(\mathbf{v}(A))$ for the actual available set A if at least one candidate is available. If $Y \subseteq X$ is a smallest (cardinality) set of candidates such that $r(\mathbf{v})$ is a robust winner for Y, then its expected query cost is $|Y| + O(\varepsilon)$. We can also show that any r-sufficient policy has an expected cost of at least $|Y| - O(\varepsilon)$.[4] These facts prove:

Proposition 7 *The policy* $Extreme(\cdot)$ *is asymptotically optimal as* $\varepsilon \to 0$.

Empirical Evaluation

We now discuss experiments that test the effectiveness of our algorithms for computing query policies, and examine the expected costs of these policies for various voting rules, preference distributions and availability distributions. We generate vote profiles using *Mallows distributions* over rankings (Mallows 1957), given by a modal ranking σ over X and dispersion $\phi \in (0, 1]$: the probability of vote v is $\Pr(v|\sigma, \phi) \propto \phi^{d(r,\sigma)}$, where d is Kendall's τ-distance. Smaller ϕ concentrates mass around σ while $\phi = 1$ gives the uniform distribution (i.e., *impartial culture*). We use $m = 10$ candidates and $n = 100$ voters, generating profiles for $\phi \in \{0.3, 0.8, 1.0\}$, and consider three voting rules: plurality, Borda and Copeland. We vary the availability probabilities p with each candidate having the same p. Results for

[4] This bound is discontinuous at $\varepsilon = 0$, but then all candidates are available, so the query cost is zero. Thanks to a reviewer for pointing this out.

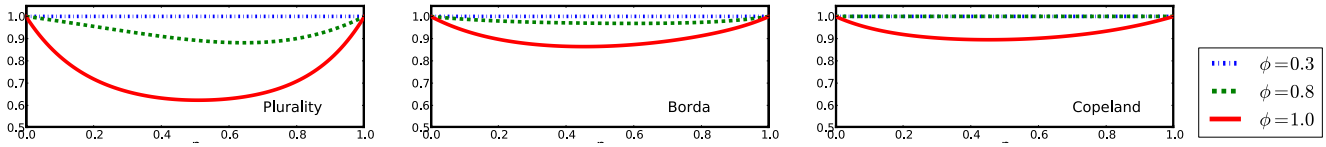

Figure 2: Probability an available naïve winner is the true winner.

	Query Cost. $\phi = 0.8$			Query Cost. $\phi = 1.0$			Tree Size. $\phi = 0.8$			Tree Size. $\phi = 1.0$		
p	0.3	0.5	0.9	0.3	0.5	0.9	0.3	0.5	0.9	0.3	0.5	0.9
Plur-DP	4.1 (3.2,5.2)	3.4 (2.0,5.4)	2.7 (1.1,5.4)	6.7 (5.9,7.6)	6.6 (4.9,7.4)	5.4 (2.4,7.9)	52.0 (9,128)	49.6 (9,124)	57.0 (9,148)	233 (133,311)	221 (121,302)	249 (136,318)
Borda-DP	3.7 (3.2,4.5)	2.7 (2.0,3.9)	1.7 (1.1,5.0)	5.4 (4.4,6.7)	4.8 (3.2,6.4)	3.3 (1.2,6.2)	24.4 (9,61)	24.0 (9,57)	26.2 (9,63)	114 (42,209)	110 (41,197)	125 (44,226)
Cope-DP	3.2 (3.2,3.6)	2.0 (2.0,2.5)	1.1 (1.1,1.3)	4.6 (3.4,5.9)	3.6 (2.1,5.6)	2.2 (1.1,4.5)	10.3 (9,19)	10.3 (9,19)	10.3 (9,19)	58.4 (17,161)	57.8 (17,160)	63.1 (17,185)
Plur-IG	4.1 (3.2,5.2)	3.5 (2.0,5.5)	2.8 (1.1,5.6)	7.0 (6.2,8.0)	6.9 (5.0,7.7)	5.6 (2.4,8.2)	59.5 (9,140)	55.4 (9,140)	62.2 (9,163)	290 (163,379)	258 (145,351)	296 (171,402)
Borda-IG	3.7 (3.2,4.6)	2.7 (2.0,3.9)	1.7 (1.1,5.0)	5.5 (4.5,7.0)	4.9 (3.2,6.7)	3.3 (1.2,6.2)	26.8 (9,63)	24.1 (9,59)	26.5 (9,68)	136 (49,264)	117 (42,229)	135 (46,253)
Cope-IG	3.2 (3.2,3.6)	2.0 (2.0,2.5)	1.1 (1.1,1.3)	4.7 (3.5,6.7)	3.7 (2.1,5.9)	2.2 (1.1,4.5)	10.3 (9,19)	10.3 (9,19)	10.4 (9,20)	67.1 (21,211)	60.8 (18,178)	70.2 (18,213)

Table 1: Avg. query cost and tree size (min, max) for optimal (DP) and myopic (IG) query policies

each problem instance (combination of voting rule, ϕ, p) are averaged over 25 random vote profiles.

Before exploring query policies, we measure the probability of selecting an incorrect winner using a policy that selects the *naïve winner*, $r(\mathbf{v})$, ignoring candidate unavailability. An obvious lower bound on this error is $1-p$ (i.e., when the winner is unavailable). Fig. 2 shows this error probability *conditional on the winner being available* for the three voting rules considered and different ϕ, as we vary p. For p near 1, the naïve winner is, of course, almost always correct. At the other extreme, the naïve winner is also usually correct, since it is highly likely to be the only available option. When preferences are very peaked ($\phi = 0.3$), candidate availability has little impact (most voters have similar rankings; but as they become more uniform the impact is dramatic. This suggests that testing availability is important even for reasonably high values of p. These results give only a crude sense of the "degree of robustness" of a winner *who is assumed to be available*, even for low p, and provide minimal insight into the value of intelligent availability testing.

We now consider the expected number of queries needed to determine the winner in the settings described above (using the same values of ϕ) under different availability probabilities: $p = 0.3, 0.5, 0.9$. Results for all three voting rules and six of nine parameter settings, with average expected cost (min, max) over 25 trials, are shown in the left half of Table 1.[5] In most settings, optimal query policies offer significant savings relative to the approach that first tests the availability of all ten candidates. The myopic heuristic tends to produce trees with costs very close to the optimum: even in problems with the largest gap (i.e., plurality with $\phi = 1$), myopic trees have an average expected cost of only 0.3 more queries than optimal; in most cases, myopic trees are almost identical to the optimum; so the more efficient myopic approach effectively minimizes query costs in practice. Not surprisingly, we see strong (negative) correlations between cost and availability probability in all three rules. Query cost is also correlated with dispersion ϕ: when ϕ is greater (more uniform), costs are higher since preferences are more diverse. When dispersion ϕ is low, given a fixed p, expected cost is the essentially the same for each rule, and the myopic approach is virtually optimal.

The right half of Table 1 shows the sizes of the decision trees that result when running both of our algorithms: tree size is only indirectly related to expected query cost, since the relative balance of the trees also impacts costs. Nonetheless we see an expected correlation, though plurality tends to result in larger trees, especially for $\phi = 1$. The myopic trees are not significantly larger than the optimal trees, though the differences in size are somewhat more pronounced than the differences in query cost.

We also ran experiments to test the effectiveness of querying for *approximate robustness*, that is, terminating the querying process when the information set ensures that a specific candidate is the true winner with probability at least $1-\delta$. Space precludes a full discussion, but using a modified version of the DP algorithm, we computed optimal query policies for values of $\delta \in \{0.001, 0.01, 0.1\}$ (i.e., exactly optimal policies given the goal of finding a $1-\delta$-robust winner). With plurality, dispersion $\phi = 1.0$ and $p = 0.9$, fully robust policies had an expected cost of 5.42 queries on average. Allowing $1-\delta$-robust winners greatly reduced the expected cost: with $\delta = 0.001$ average expected cost was 4.97 queries; for $\delta = 0.01$, 4.36 queries; and for $\delta = 0.1$, 3.04 queries. For $p = 0.3$, setting $\delta = 0.1$ saw a reduction to 5.28 queries (compared to 6.73 for exact robustness). Other voting rules exhibited similar patterns.

Future Directions

We have offered a new perspective on voting in the unavailable candidate model, assuming that testing the viability or availability of candidates is costly. Using robust winners, irrelevant candidates, and query policies, our algorithms for computing query policies were shown to be effective, and empirical results demonstrated the value of optimal querying. A number of important research directions remain, including: efficient methods for pruning available sets w.r.t. specific voting rules; sample-based methods for reducing training set size; further development of policies that "predict" winners; deeper theoretical study of query and communication complexity; and analysis of manipulation.

[5]Results for $\phi = 0.3$ are not shown, as they are identical for all three voting rules and both algorithms. For $p = 0.3$, avg. query cost is 3.2; $p = 0.5$, avg. cost is 2.0; and $p = 0.9$ avg. cost is 1.1. Tree sizes (right half of the table) for $\phi = 0.3$ are *constant* (size is always 9 nodes) across all rules, algorithms, and p values.

Acknowledgments: Thanks to the reviewers for helpful suggestions. This work was supported in part by NSERC and by MICINN project TIN2011-27652-C03-02.

References

Baldiga, K., and Green., J. 2013. Assent-maximizing social choice. *Social Choice and Welfare* 40(2):439–460.

Bartholdi, J.; Tovey, C.; and Trick, M. 1992. How hard is it to control an election? *Social Choice and Welfare* 16(8-9):27–40.

Chevaleyre, Y.; Lang, J.; Maudet, N.; and Monnot, J. 2011. Compilation/communication protocols for voting rules with a dynamic set of candidates. In *Proceedings of the 13th Conference on Theoretical Aspects of Rationality and Knowledge (TARK-11)*, 153–160.

Chevaleyre, Y.; Lang, J.; Maudet, N.; Monnot, J.; and Xia, L. 2012. New candidates welcome! possible winners with respect to the addition of new candidates. *Mathematical Social Sciences* 64(1):74–88.

Dutta, B.; Jackson, M. O.; and Breton, M. L. 2001. Strategic candidacy and voting procedures. *Econometrica* 69(4):1013–1037.

Erdélyi, G.; Fellows, M. R.; Piras, L.; and örg Rothe, J. 2011. Control complexity in Bucklin and fallback voting. arXiv 1103.2230.

Faliszewski, P.; Hemaspaandra, E.; and Hemaspaandra, L. 2011. Multimode control attacks on elections. *Journal of Artificial Intelligence Research* 40:305–351.

Faliszewski, P.; Hemaspaandra, E.; and Schnoor, H. 2008. Copeland voting: Ties matter. In *Proceedings of the Seventh International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS-08)*, 983–990.

Garey, M. R. 1972. Optimal binary identification procedures. *SIAM Journal of Applied Mathematics* 23:173–186.

Greiner, R.; Grove, A. J.; and Roth, D. 1992. Learning cost-sensitive active classifiers. *Artificial Intelligence* 139(2):137–174.

Hemaspaandra, E.; Hemaspaandra, L.; and Rothe, J. 2007. Anyone but him: The complexity of precluding an alternative. *Artificial Intelligence* 171(5-6):255–285.

Hyafil, L., and Rivest, R. L. 1976. Constructing optimal binary decision trees is NP-complete. *Information Processing Letters* 5:15–17.

Lu, T., and Boutilier, C. 2010. The unavailable candidate model: A decision-theoretic view of social choice. In *Proceedings of the Eleventh ACM Conference on Electronic Commerce (ACM EC-10)*, 263–274.

Mallows, C. L. 1957. Non-null ranking models. *Biometrika* 44:114–130.

Parkes, D. C., and Xia, L. 2012. A complexity-of-strategic-behavior comparison between schulze's rule and ranked pairs. In *Proceedings of the Twenty-sixth AAAI Conference on Artificial Intelligence (AAAI-12)*, 1429–1435.

Procaccia, A. D.; Rosenschein, J. S.; and Kaminka, G. A. 2007. On the robustness of preference aggregation in noisy environments. In *Proceedings of the Sixth International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS-07)*, 416–422.

Quinlan, J. R. 1993. *C45: Programs for Machine Learning*. Morgan Kaufmann.

Rastegari, B.; Condon, A.; Immorlica, N.; and Leyton-Brown, K. 2013. Two-sided matching with partial information. In *Proceedings of the Fourteenth ACM Conference on Electronic Commerce (ACM EC-13)*, 733–750.

Shiryaev, D.; Yu, L.; and Elkind, E. 2013. On elections with robust winners. In *Proceedings of the Twelfth Conference on Autonomous Agents and Multiagent Systems (AAMAS-13)*, 415–422.

Turney, P. D. 1995. Cost-sensitive classification: Empirical vvaluation of a hybrid genetic decision tree induction algorithm. *Journal of Artificial Intelligence Research* 2:369–409.

Wojtas, K., and Faliszewski, P. 2012. Possible winners in noisy elections. In *Proceedings of the Twenty-sixth AAAI Conference on Artificial Intelligence (AAAI-12)*, 1499–1505.

Xia, L. 2012. Computing the margin of victory for various voting rules. In *Proceedings of the Thirteenth ACM Conference on Electronic Commerce (ACM EC-12)*, 982–999.