Ranked Choice Voting, the Primaries System, and Political Extremism: Theory and Simulations*

Avidit Acharya, Rohan Cherivirala, Robin Truax, Karsen Wahal†

June 28, 2024

Abstract

We compare political extremism among election winners in a model of ranked choice voting (RCV) to winners in a model of the currently predominant electoral system (CES) in the U.S., which is based on partisan primaries and plurality rule. In three candidate elections, extremism under RCV is always bounded below what it is under CES. However, this does not hold in elections with four or more candidates. Still, in these many-candidate elections, the winners under RCV tend to be less extreme than those under CES, though RCV's performance in selecting moderate candidates relative to CES declines as the number of candidates increases. We show this by simulating a large number of elections for different distributions of voter ideology while varying the ideological positions of the candidates.

We also run simulations to compare the variants of RCV that are currently implemented in the U.S. We show that the Alaska variant in which a blanket nonpartisan primary is used to select the top four candidates that then compete in a general election in which RCV is used outperforms the nonpartisan single-round RCV system (the "standard" RCV system as implemented, for example, in Minneapolis and other municipal elections) in selecting moderate candidates. The standard RCV system in turn tends to yield more moderate election winners than the Maine system in which RCV is used in partisan primaries to nominate candidates for the general election.

Key words: ranked choice voting, primaries, extremism

*We are grateful to Peter Buisseret, Gary Cox, and Andy Hall for helpful comments and especially to Andy Eggers for detailed comments and suggestions.

†Acharya is the corresponding author, and Professor of Political Science at Stanford University (email: avidit@stanford.edu). Cherivirala, Truax, and Wahal are all Stanford undergraduates.

1 Introduction

Under the current electoral system (CES) that is in place in much of the United States, the two major parties nominate the winner of their respective primaries to compete against each other in a general election. Some commentators have argued that replacing this system with a system of ranked choice voting (RCV) is more likely to favor moderate candidates. For example, writing in the *New York Times*, Eric Maskin and Amartya Sen commend Maine's adoption of RCV for gubernatorial and congressional elections, referring to it as "a better electoral system" for reducing political extremism and improving representation.[1] Drutman (2020) and Horowitz (2004) make similar claims. RCV is the fastest growing electoral system in America. FairVote, an organization dedicated to promoting RCV as an electoral reform in the U.S., notes that by the end of 2022 sixty-three American jurisdictions had already adopted RCV.

In this paper, we investigate the arguments that RCV favors more moderate candidates by comparing the extremism of winning candidates in a model of RCV to the extremism of winning candidates in a CES model. Our models are based on the following assumptions, which we maintain throughout the paper, except when noted:

1. Voter ideal points are distributed on the one-dimensional bounded ideological space $[0, 1]$ according to some continuous (i.e. atomless) distribution F with full support and median M_F, and some finite number of candidates $\{C_1, ..., C_n\}$ are located at different positions in this interval. We identify a candidate C_i with their ideological location, so that $C_i \in [0, 1]$ for all i, and order them without loss so that $C_1 < ... < C_n$. Fixing M_F, we measure the extremism of any candidate C by $E_F(C) = |M_F - C|$ so that the most extreme left position is 0, the most extreme right position is 1, and the most moderate position is M_F.

2. Under CES, all candidates positioned to the left of M_F run in the Democratic primary, and those to the right run in the Republican primary. If there are no Democratic candidates, then the winner of the Republican primary automatically wins the general election uncontested; vice-versa if there are no Republicans. If there is at least one Democrat and one Republican, then the winners of the two primaries run against each other in the general. All voters vote for the candidate whose position is closest to theirs in the primary, and do so again in the general election. At each step, the candidate that has a plurality wins.

3. Under RCV, all voters rank all of the candidates. A voter ranks the candidate whose position is closest to their ideal point first, the second closest candidate second, and so on. If there is a candidate that wins a majority of votes for first place, that candidate is the winner of the election. If there is no such candidate, then the candidate that is ranked first by the smallest mass of voters is eliminated, and all candidates that were ranked below the eliminated candidate move up one space in all of the voters' rankings. If now a candidate has a majority of first place rankings, they win the election; otherwise, the process is repeated again until some candidate has a majority.

In addition to assuming that the distribution of voter ideology F is continuous with support $[0, 1]$, we also maintain the assumption throughout that the candidates' positions

[1] "A better electoral system in Maine," *New York Times*, June 10, 2018.

are *generic*, which means that no two candidates are ever tied in their number of votes at any stage/round of either electoral system, $C_1 \neq 0$, $C_n \neq 1$, and $C_i \neq M_F$ for all i. The restriction to generic positions is based on the idea that a random selection of candidates from the distribution of voters F will result in generic positions with probability 1.

We prove that in three candidate elections, the extremism of the winner under RCV is bounded below the extremism of the winner under CES. However, this is not the case with four or more candidates.[2] As an example, suppose that F is uniform on $[0, 1]$ and consider a four candidate election in which the candidates C_1, C_2, C_3 and C_4, are located at the points 0.17, 0.49, 0.79 and 0.81 respectively. The median of F is 0.50 so under CES, C_1 and C_2 would run against each other in the Democratic primary while C_3 and C_4 would run against each other in the Republican primary. C_1 would win the Democratic primary while C_3 would win the Republican primary. Then, C_3 would win the general election.

What would happen under RCV? No candidate would have a majority of first place ballots in the first tally. In fact, the candidate with the fewest first place votes would be C_3, and they would be eliminated in the first round. After eliminating C_3, still no candidate would have a majority. At this point, C_2 would be eliminated, leaving C_1 and C_4. C_4 would have more support than C_1 in this round, and therefore win the election. So while CES would select C_3, RCV would select C_4, a more extreme candidate than C_3. The table below shows the vote tallies for each round of each system (primary and general for CES and three elimination rounds of RCV).

		CES		RCV		
	Location	Primary	General	Round 1	Round 2	Round 3
C_1	0.17	66%	48%	33%	33%	49%
C_2	0.49	44%		31%	32%	
C_3	0.79	60%	52%	16%		
C_4	0.81	40%		20%	35%	51%

To get a handle on how often RCV produces more extreme winners than CES, we simulate a large number of elections selecting distributions F from the class of normal distributions truncated on $[0, 1]$, and drawing candidate locations generically from the same distributions. We do this for elections with 4 to 100 candidates. We find that political extremism of the winner is more often lower in RCV than in CES, regardless of the number of candidates or distribution. In fact, this feature is not specific to truncated normal distributions, which are all unimodal. In Figure 1, we repeat the same exercise but in each simulation of each election for a fixed number of candidates, we draw a different random cubic spline distribution with support $[0, 1]$ (see Appendix A for more details on how we draw these distributions). The figure shows that the relative overall performance of RCV in producing more moderate winners than CES is declining in the number of candidates, while the relative overall performance of CES in producing more moderate winners than RCV is increasing in the number of candidates. Nevertheless, RCV performance remains better than CES performance throughout. At 100 candidates, RCV produces more moderate

[2]These analytic results described in Section 2 for three- and four-candidate elections apply to all possible generic values of candidate locations $C_1, ..., C_n$ while in the election simulations of Section 3 that we describe below, we assume that $C_1, ..., C_n$ are drawn iid from the same distribution F as voter ideology—an assumption that we revisit in Section 6 while discussing the challenges to modeling strategic entry.

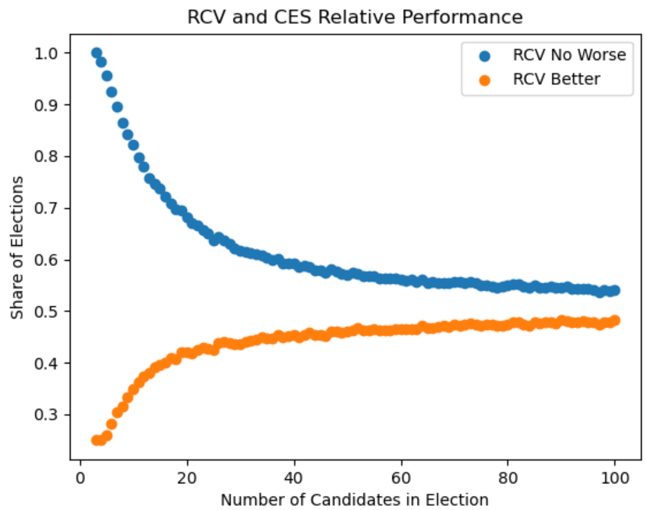


Figure 1: The figure shows the percent of times that the winning candidate under RCV is at least as moderate as the winner under CES (the "RCV no worse" case) and the percent of times that RCV is more moderate than CES (the "RCV better" case), as a function of the number of candidates in an election. For each number of candidates from 4 to 100, we simulate 50,000 elections and report the average performance of RCV relative to CES. We take 10,000 independent random draws of a cubic spline distribution (see Appendix A for details) and run 5 elections for each draw. In each election, we draw the platform locations of the candidates randomly and independently from the same distribution.

winners than CES approximately 48% of the time, while CES produces more moderate winners than RCV approximately 45% of the time. In the remainder of cases, political extremism of the winner is the same in RCV as in CES because the two electoral systems result in the same candidate winning office.

In a recent paper, Maskin (2022) advocates for majority rule over RCV, which in our one-dimensional model would guarantee selecting the most moderate candidate. In fact, because in our model a candidate C is majority preferred to C' if and only if C is more moderate than C', moderation and majority representation are concepts that perfectly coincide. Nevertheless, our interest is in studying the properties of RCV rather than those of majority rule given the growing interest and increased popularity of RCV as an electoral reform in the United States.

2 Analytic Results

In this section, we provide analytic results for three- and four-candidate elections maintaining all of the notation and assumptions on distribution of voter ideology and candidate locations mentioned in the introduction. In addition, we denote the vote mass of candidate C in a plurality election when all candidates are running by $V(C)$ and the vote mass of the candidate in their partisan primary by $V_p(C)$. The winner of RCV is denoted W_{RCV} and the winner of CES is W_{CES}.

Theorem 1. *For any continuous distribution of voters F on support $[0,1]$ and any set of $n = 3$ generically located candidates, $E_F(W_{\text{CES}}) \geq E_F(W_{\text{RCV}})$.*

Proof. The result is trivial when all three candidates are on the same side of M_F, as RCV will always select the most moderate candidate.

Therefore, assume that there is one candidate to one side of M_F and two to the other; by symmetry, it is without loss to assume that $C_1 < M_F < C_2 < C_3$. In this case, C_1 wins the Democratic primary, and let $P \in \{C_2, C_3\}$ denote the winner of the Republican primary. As $V_p(C)$ is the mass of votes won by $C \in \{C_2, C_3\}$ in the Republican primary, we have $V_p(C_2) + V_p(C_3) = \frac{1}{2}$.

Now, assume for the sake of contradiction that $E(W_{\text{CES}}) > E(W_{\text{RCV}})$. We cannot have $W_{\text{RCV}}, W_{\text{CES}} < M_F$, as then $W_{\text{RCV}} = W_{\text{CES}} = C_1$. So there are three cases.

Case 1: $W_{\text{RCV}} > W_{\text{CES}} > M_F$. Here, $W_{\text{RCV}} = C_3$ and $W_{\text{CES}} = P = C_2$. Then, either C_1 or C_2 must be eliminated first under RCV. However, if C_1 is eliminated first, then since $M_F < C_2 < C_3$, $W_{\text{RCV}} = C_2 \neq C_3$, a contradiction. On the other hand, if C_2 is eliminated first, then

$$V_p(C_2) > V_p(C_3) = V(C_3) > V(C_2)$$

The first inequality means that C_2 wins the primary. The equality means that C_3 cannot win any voters to the left of M_F. The second inequality means that C_2 gets eliminated first. It then follows that $V(C_2) + V(C_3) < V_p(C_2) + V_p(C_3) = \frac{1}{2}$, so $V(L) > \frac{1}{2}$ and $W_{\text{RCV}} = C_1 \neq C_3$, a contradiction.

Case 2: $W_{\text{RCV}} > M_F > W_{\text{CES}}$, and $\frac{W_{\text{RCV}} + W_{\text{CES}}}{2} > M_F$. In this case, $W_{\text{RCV}} = C_2$ or C_3 and $W_{\text{CES}} = C_1$. If $W_{\text{RCV}} = C_2$, then $\frac{C_1 + C_2}{2} > M_F$, whence $V(C_1) > \frac{1}{2}$ and $W_{\text{RCV}} = C_1 \neq C_2$, a contradiction. Thus, $W_{\text{RCV}} = C_3$. Then, either C_1 or C_2 are eliminated first under RCV. If C_1 is eliminated first, then C_2 wins RCV in the second round, a contradiction with $W_{\text{RCV}} = C_3$. Thus, C_2 is eliminated first. But then the winner of RCV is determined by a simple majoritarian election between C_1 and C_3, and by assumption $\frac{C_1 + C_3}{2} > M_F$, so $W_{\text{RCV}} = C_1 \neq C_3$, another contradiction.

Case 3: $W_{\text{RCV}} < M_F < W_{\text{CES}}$, and $\frac{W_{\text{RCV}} + W_{\text{CES}}}{2} < M_F$. In this case, $W_{\text{RCV}} = C_1$ and $W_{\text{CES}} = P \in \{C_2, C_3\}$. Now, since P beats C_1 in a simple election, P must be eliminated first in RCV for C_1 to win. If $P = C_2$, then C_2 is eliminated first but wins the CES primary, so by the same reasoning as in Case 1, C_1 wins RCV outright. But this is impossible, since if C_1 wins RCV outright, then C_1 would win in the general election under CES, a contradiction. Thus, $P = C_3$. Since C_3 is eliminated first, $V(C_2) > V(C_3)$. Yet because C_3 wins the primary, $V(C_3) = V_p(C_3) > \frac{1}{4}$. Combining these results, $V(C_2) + V(C_3) > \frac{1}{2}$. But since C_3 is eliminated first, all of C_3's voters vote for C_2 in the second round of RCV, meaning that C_2 receives $V(C_3) + V(C_2) > \frac{1}{2}$ votes and wins, a contradiction with $W_{\text{RCV}} = C_1$. ■

To state and prove our result for four-candidate elections we introduce some additional notation. L_i is the ith candidate from M_F on the left. R_i is the ith candidate from M_F on the right. The vote share of candidate C_i in a plurality election with candidates $\mathcal{S} \subseteq \{C_1, \ldots, C_n\}$ is $V_{\mathcal{S}}(C_i)$. $V_{-C_j}(C_i)$ denotes $V_{\{C_1, \ldots, C_{j-1}, C_{j+1}, \ldots C_n\}}(C_i)$ for $i \neq j$. All other notation remains the same; in particular, $V(C_i)$ is $V_{\{C_1, \ldots, C_n\}}(C_i)$ and $V_p(C_i)$ is candidate C_i's vote mass in their partisan primary under CES.

In our main result, we divide the analysis of four candidate elections into three cases: (i) elections, in which all four are on the same side of M_F, (ii) those in which only three candidates are on the same side of M_F, and (iii) those in which two are on one side of M_F and the other two are on the other side.

Theorem 2. *For any continuous distribution of voters F on support $[0,1]$ and any set of $n=4$ generically located candidates:*

(i) If all candidates are located on the same side of M_F (i.e. the candidates are L_4, L_3, L_2, L_1 or R_1, R_2, R_3, R_4), then $E_F(W_{\text{RCV}}) \leq E_F(W_{\text{CES}})$.

(ii) If the candidates are L_1, R_1, R_2, R_3, then $E_F(W_{\text{RCV}}) \leq E_F(W_{\text{CES}})$ unless

$$E_F(R_1) < E_F(L_1), \qquad V_p(R_1) > V_p(R_2), \qquad V_p(R_1) > V_p(R_3)$$

and one of the two cases:

(a) $V(R_2) > V(R_3)$ and $V_{-R_3}(R_2) > V(R_1)$

(b) $V(R_2) < V(R_3)$ and $V_{-R_2}(R_3) > V_{-R_2}(R_1)$

If the candidates are L_3, L_2, L_1, R_1 then the same result holds with R_i and L_i swapped for each i.

(iii) If the candidates are L_2, L_1, R_1, R_2, then $E_F(W_{\text{RCV}}) \leq E_F(W_{\text{CES}})$ unless

$$E_F(L_1) < E_F(R_1) < E_F(L_2), \quad V_p(R_2) < V_p(R_1), \quad V_p(L_1) < V_p(L_2),$$
$$V(R_1) < V(R_2), \quad V(R_1) < V(L_1), \quad V_{-R_1}(L_1) < V(L_2), \quad V_{-R_1}(L_1) < V_{-R_1}(R_2)$$

or the same with R_i and L_i swapped for each i.

Proof. (i) The case of all candidates being on the same side of M_F is trivial as in the three candidate case, as RCV will always select the most moderate candidate.

(ii) Next, suppose the candidates are L_1, R_1, R_2, R_3 and assume that $E_F(W_{\text{CES}}) < E_F(W_{\text{RCV}})$. We now have nine cases to handle.

Case 1: $W_{\text{CES}} = R_1$, $W_{\text{RCV}} = L_1$. Here, we have $E_F(R_1) < E_F(L_1)$. Thus, $V(R_1) > V_p(R_1)$, so R_1 is not eliminated first in RCV. Yet R_1 cannot be eliminated last as $E_F(R_1) < E_F(L_1)$ implies R_1 would win. Thus, we must eliminate R_2 or R_3, then R_1, then the other of R_2 or R_3. Thus, we require $E_F(R_1) < E_F(L_1)$, $V_p(R_1) > V(R_2)$, $V_p(R_1) > V(R_3)$, and one of

$$\begin{matrix} V(R_2) > V(R_3) \\ V_{-R_3}(R_2) > V(R_1) \\ E_F(L_1) < E_F(R_2) \end{matrix} \quad \textbf{or} \quad \begin{matrix} V(R_3) > V(R_2) \\ V_{-R_2}(R_3) > V_{-R_2}(R_1) \\ E_F(L_1) < E_F(R_3) \end{matrix} \tag{1}$$

Case 2: $W_{\text{CES}} = R_2$, $W_{\text{RCV}} = L_1$. R_2 cannot be eliminated in the RCV process before R_3 since $W_{\text{CES}} = R_2$. Therefore, in the final round, L_1 faces either R_1 or R_2. Yet, since $E_F(W_{\text{CES}}) < E_F(W_{\text{RCV}})$, $E_F(R_1) < E_F(R_2) < E_F(L_1)$, so L_1 loses in the final round of RCV, a contradiction.

Case 3: $W_{\text{CES}} = R_3$, $W_{\text{RCV}} = L_1$. $E_F(R_1) < E_F(R_2) < E_F(R_3) < E_F(L_1)$, so L_1 loses in the final round of RCV to any possible opponent, a contradiction.

Case 4: $W_{\text{CES}} = L_1$, $W_{\text{RCV}} = R_1$. In this case, $E_F(L_1) < E_F(R_1)$ by assumption. But then L_1 wins outright in RCV.

Case 5: $W_{\text{CES}} = R_1$, $W_{\text{RCV}} = R_2$. In this case, $E_F(R_1) < E_F(L_1)$. Then $V(R_1) > V_p(R_1) > V_p(R_2)$, so either L_1 or R_3 is eliminated first. Yet if L_1 is eliminated before R_1, R_1 wins outright. Therefore, R_3 must be eliminated first. Again, L_1 cannot be eliminated, so R_1 must be eliminated. Then, we must have $E_F(R_2) < E_F(L_1)$ so that R_2 wins. Now, all of this is possible, but only in the non-uniform case. Translated into inequalities, we obtain the following:

$$E_F(R_1) < E_F(L_1) \qquad V_p(R_1) > V_p(R_2) \qquad V_p(R_1) > V_p(R_3)$$

$$V(R_2) > V(R_3) \qquad V_{-R_3}(R_2) > V(R_1) \qquad E_F(R_2) < E_F(L_1)$$

Case 6: $W_{\text{CES}} = L_1$, $W_{\text{RCV}} = R_2$. In this case, $E_F(L_1) < R_2$. But then, if R_1 is eliminated before L_1, L_1 wins outright. On the other hand, if L_1 is eliminated before R_1, R_1 wins outright. Thus R_2 cannot win.

Case 7: $W_{\text{CES}} = R_1$, $W_{\text{RCV}} = R_3$. In this case, $E_F(R_1) < E_F(L_1)$. Then $V(R_1) > V_p(R_1) > V_p(R_3)$, so either L_1 or R_2 is eliminated first. Yet if L_1 is eliminated before R_1, R_1 wins outright. Therefore, R_2 must be eliminated first. Again, L_1 cannot be eliminated before R_1, so R_1 must be eliminated. Then, we must have $E_F(R_3) < E_F(L_1)$ so that R_3 wins. Translated into inequalities, we obtain the following:

$$E_F(R_1) < E_F(L_1) \qquad V_p(R_1) > V_p(R_2) \qquad V_p(R_1) > V_p(R_3)$$

$$V(R_2) < V(R_3) \qquad V_{-R_2}(R_3) > V_{-R_2}(R_1) \qquad E_F(R_3) < E_F(L_1)$$

Case 8: $W_{\text{CES}} = R_2$, $W_{\text{RCV}} = R_3$. In this case, $V(R_2) = V_p(R_2) > V_p(R_3) = V(R_3)$, and $V(R_3)$ cannot increase before $V(R_2)$ dies, so R_3 is always eliminated before R_2. Thus R_3 cannot win.

Case 9: $W_{\text{CES}} = L_1$, $W_{\text{RCV}} = R_3$. Since $E_F(L_1) < E_F(R_3)$, if R_1 and R_2 are eliminated before L_1, L_1 wins outright. On the other hand, if L_1 is eliminated before R_1 and R_2, either R_1 or R_2 (in that order) wins outright. Thus R_3 cannot win.

The claim in part (ii) that the same result holds with R_i replacing L_i and L_i replacing R_i for all i follows by symmetry.

(iii) Finally, suppose the candidates are L_2, L_1, R_1, R_2 and assume that $E_F(W_{\text{CES}}) < E_F(W_{\text{RCV}})$. We have five cases to handle.

Case 1: $W_{\text{CES}} = R_1$, $W_{\text{RCV}} = R_2$. Now, we cannot have $E_F(R_1) < E_F(L_1)$, as then regardless of whether L_1 or L_2 have been eliminated, $V(R_1) > V_p(R_1) > V_p(R_2) = V(R_2)$ whence $W_{\text{RCV}} \neq R_2$. Thus, $E_F(L_1) < E_F(R_1)$, so L_2 must win the left primary. L_1 cannot be eliminated first, as then $E_F(R_1) < E_F(L_2)$, and the same reasoning as above shows $W_{\text{RCV}} \neq R_2$. Also, $E_F(L_2)$ cannot be eliminated first, as then $V(L_1) > \frac{1}{2}$ and $W_{\text{RCV}} \neq R_2$. Thus, R_1 is eliminated first in RCV. Now, if L_2 is eliminated next in RCV, then $V(L_1) > \frac{1}{2}$, a contradiction. So it must be the case that L_1 is eliminated next in RCV, and then R_2 wins the head-to-head vs. L_2.

So the specific series of events which must happen is: R_1 wins the Republican primary and L_2 wins the Democratic primary, and then R_1 wins the head-to-head vs. L_2. Then, R_1 is eliminated first in RCV, followed by L_1, and then R_2 wins the head-to-head vs. L_2. This can all be encoded by the following formulas:

$$E_F(L_1) < E_F(R_1) < E_F(R_2) < E_F(L_2), \quad V_p(R_2) < V_p(R_1), \quad V_p(L_1) < V_p(L_2)$$

$$V(R_1) < V(R_2), \quad V(R_1) < V(L_1), \quad V_{-R_1}(L_1) < V(L_2), \quad V_{-R_1}(L_1) < V_{-R_1}(R_2)$$

These formulas are satisfiable.

Case 2: $W_{\text{CES}} = R_1$, $W_{\text{RCV}} = L_1$. Now, we have $E_F(R_1) < E_F(L_1)$ by assumption. But then, since R_1 wins the right primary, R_1 is never eliminated before R_2, at which point $V(R_1) > \frac{1}{2}$, a contradiction.

Case 3: $W_{\text{CES}} = R_1$, $W_{\text{RCV}} = L_2$. We cannot have $E_F(R_1) < E_F(L_1)$, since then by the same logic as above R_1 wins the RCV process. Thus, $E_F(L_1) < E_F(R_1)$. Then L_2 must win the Democratic primary, as otherwise $W_{\text{CES}} = L_1$. Now, of course L_2 cannot be eliminated from in RCV first. On the other hand, R_1 and L_1 win the head-to-head vs. L_2, so they must be eliminated in the first two rounds. Therefore, R_2 must be eliminated in the final round. It remains to determine which of R_1 and L_1 is eliminated first. Since $E_F(L_2) > E_F(R_1)$, we cannot eliminate L_1 first, as then $V(R_1) > V_p(R_1) > V_p(R_2) = V(R_2)$, and R_2 is eliminated before R_1, contradiction.

Thus, we must have $E_F(L_1) < E_F(R_1) < E_F(L_2) < E_F(R_2)$. We must also have $V_p(R_2) < V_p(R_1)$, as well as $V_p(L_1) < V_p(L_2)$. Furthermore, we require $V(R_1) < V(R_2)$, $V(R_1) < V(L_1)$, and then $V_{-R_1}(L_1) < V(L_2)$, and $V_{-R_1}(L_1) < V_{-R_1}(R_2)$. This is almost the same list of requirements as in Case 1.

Case 4: $W_{\text{CES}} = R_2$, $W_{\text{RCV}} = L_1$. $E_F(L_1) > E_F(R_2)$, implying $E_F(L_1) > E_F(R_1)$. But then whichever of R_1 or R_2 is eliminated first, the other will have at least half of the vote and win under RCV outright. Thus, this case is impossible.

Case 5: $W_{\text{CES}} = R_2$, $W_{\text{RCV}} = L_2$. If $E_F(L_2) > E_F(R_2)$, then L_1, R_1, and R_2 all win the head-to-head vs. L_2, so L_2 cannot win in RCV.

By symmetry the same statement holds with R_i and L_i swapped for each i. ■

Theorem 2 establishes the cases under which RCV produces more extreme winners than CES. But are there choices of F along with candidate locations that satisfy these cases? The numerical example in the introduction for uniform F shows that the conditions of part (iii) of the theorem can be satisfied. But what about the other conditions? And how frequently are these conditions satisfied or violated for reasonable choices of F, i.e. those we may consider to be descriptive of the actual distribution of voter ideology? For example, in the proof of the theorem, the second set of conditions in display (1) of Case 1 in part (ii) of the proof never occurs if F is the uniform distribution or any truncated normal distribution on $[0, 1]$. Similarly, for the conditions of Case 7 in part (ii) of the proof to hold, F cannot be uniform on $[0, 1]$. But since we know that RCV can produce more extreme winners than CES even when F is uniform, this raises the question of what we can say about *how frequently* RCV produces more and less extreme winners than CES. Of course, the answer will depend on the distribution F of voter ideology, and the distribution over candidate locations. We address this question in the next section.

3 Quantitative Study

3.1 Simulations with Truncated Normal Distributions

To address how frequently RCV produces more extreme winners, less extreme winners and equally extreme winners as CES, we run simulations of RCV and CES in which both F and the candidate locations are fixed in a simulation of each system. Although it is reasonable to think that the set of candidates who run under RCV would be different than those that run under CES, we take this comparison holding the set of candidates fixed across the two electoral systems to be a useful starting point— particularly given the challenges to modeling strategic entry.[3]

We assume that the distribution of voters follows a normal distribution truncated on $[0, 1]$. Candidates are randomly drawn from the same distribution. For each choice of F we take 500,000 draws of candidate locations. We vary the mean μ and standard deviation σ of the underlying (not truncated) normal distribution, allowing the distribution to be very concentrated or very diffuse (close to uniform). For each set of simulations for a given number of candidates, we test 25 different truncated normal distributions, with 5 different means and standard deviations. Each distribution is demonstrably different from the others, as seen in Figure B.1 in the appendix. A description of the code run in Python is given in Appendix B, and the code is publicly available on Github.[4]

The results for elections with between 4 and 9 candidates (inclusive) are summarized in Figures B.2-B.7 in Appendix B. The figures indicate that similar results hold for the variety of truncated normal distributions that we consider. One observation is that increasing the standard deviation of the underlying normal distribution tends to worsen the performance of RCV—i.e., the percent of simulated election where RCV produces a more extreme winner increases as standard deviation increases, with some exceptions. For example, for all $n \leq 7$, when the mean is 0.5 and the standard deviation is 0.1, RCV produces more extreme winners than it does when the mean is 0.5 and the standard deviation is 0.05. But overall, as the distribution becomes increasingly diffuse, RCV tends to generate a more extreme winner more often.

Another observation is that as the mean increases, RCV generates more extreme winners more often. Notably, RCV's worsened performance with increased mean and standard deviation tends to eat into cases where RCV and CES generate the same winner rather than the cases where RCV generates a less extreme winner. In nearly all elections, RCV performs strictly worse than CES and strictly better than CES at high rates when the mean and standard deviations are both high (e.g. $\mu = 0.9$ and $\sigma = 0.8$) and this is because RCV and CES are less likely to generate the same winner for high values of the mean and standard deviation.

A third and final observation is that even when RCV results in more extreme winners than CES, the winners are never as extreme as they are in the worst cases of CES. In fact, the winners for RCV seem to have a bound for their extremity.

Overall, varying the mean and standard deviation has little impact on the performance of RCV. For the 25 sets of parameter values that we consider, the results do not range significantly. These ranges are reported in Table 1. The table indicates not only that as the number of candidates increases, RCV extremism more often exceeds CES extremism; but also that CES extremism exceeds RCV extremism at higher rates. RCV and CES produce the same winner in fewer scenarios, which is not surprising given that there are more possible

[3]One view is that RCV would lead to more candidates entering than under plurality rule. However, here we are comparing RCV to the primaries system under plurality rule rather than to one-shot plurality, and it not as clear that there would be fewer candidates at the *primary* stage than under RCV.

[4]Visit `https://github.com/aviditacharya/RCV`.

n	% simulations with $E_F(W_{\text{RCV}}) > E_F(W_{\text{CES}})$	% simulations with $E_F(W_{\text{RCV}}) = E_F(W_{\text{CES}})$	% simulations with $E_F(W_{\text{RCV}}) < E_F(W_{\text{CES}})$
4	[1, 2]	[71, 73]	[25, 27]
5	[3, 6]	[66, 70]	[26, 29]
6	[6, 9]	[60, 66]	[28, 32]
7	[8, 12]	[55, 61]	[30, 34]
8	[11, 15]	[50, 56]	[32, 35]
9	[13, 18]	[45, 52]	[34, 37]

Table 1: Range of percentages where RCV generates a more extreme, equally extreme, and less extreme winner, for elections with 4 to 9 candidates, rounded to the nearest integer (floor for the lowerbound, ceiling for the upperbound).

winners in each election. In each case, however, RCV generates a more extreme winner at lower rates than CES.

Do these properties revealed in Table 1 hold as we increase the number of candidates beyond 9? Figure B.8 of Appendix B shows that they do. The figure summarizes RCV's performance relative to CES for up to 100 candidates. This figure replicates the same analysis summarized in Figure 1 in the introduction but setting the distribution to each one of the truncated normal distributions considered in Figure B.1 and holding it fixed across simulations, as the number of candidates is progressively increased. The figure reveals results broadly similar to those of Figure 1, showing that RCV's performance tends to decline and converge though on average it always selects more moderate winners than CES.

3.2 Simulations with Random Distributions

Our choice to focus on unimodal distributions is based on the fact that despite polarization in U.S. politics, the distribution of citizen ideology continues to look very much unimodal; see, e.g., McCarty (2019) and Fiorina and Abrams (2008). However, given the possibility that the distribution of voter ideology has or is becoming bimodal (see, e.g., Abramowitz and Saunders, 2008), it is reasonable to ask whether our results generalize to other distributions, including bimodal distributions.

To address this, we also report results from simulations using randomized cubic spline distributions, many of which are bimodal distributions (see Appendix A for more information on how we construct and draw these distributions randomly, as well as for a couple examples for what they can look like). The results of this exercise, summarized in Table 2, reveal that the share of simulated elections where RCV generates a more extreme, equally extreme, and less extreme winner fall into the same range for each type of election. For each number of candidates from 4 to 100, we simulate 50,000 elections and report the average performance of RCV relative to CES. We take 10,000 independent random draws of a cubic spline distribution (see Appendix A for details) and run 5 elections for each draw. The table indicates patterns that are similar to those seen in Table 1, and Figure 1 in the introduction shows patterns similar to those seen in Figure B.8 in the appendix.

n	% simulations with $E_F(W_{\text{RCV}}) > E_F(W_{\text{CES}})$	% simulations with $E_F(W_{\text{RCV}}) = E_F(W_{\text{CES}})$	% simulations with $E_F(W_{\text{RCV}}) < E_F(W_{\text{CES}})$
4	1.69	73.61	24.7
5	4.37	69.55	26.09
6	7.37	64.68	27.94
7	10.41	59.6	29.98
8	13.26	55.0	31.74
9	15.81	50.87	33.32

Table 2: Mean percentage where RCV generates a more extreme, equally extreme, and less extreme winner, for elections with 4 to 9 candidates, when tested on 5000 different spline distributions with 5000 draws of candidates each.

3.3 Exact Calculations for the Uniform Distribution

In Appendix C, we take the distribution of ideology and candidate locations, F, to be uniform and show that for this special case we can address the question of how frequently RCV produces more extreme winners than CES with a number of *exact* results. We provide an algorithm to produce a union of polytopes which constitutes all scenarios where RCV produces more extreme winners than CES, scenarios where the two electoral systems are tied, and those where CES produces more extreme winners. The algorithm works for any number of candidates n, but we highlight results only for $n \leq 7$. The results are all in line with what we see in the simulations for the truncated normal distributions and the random spline distributions discussed above.

4 Results for Different District Assumptions

4.1 Independents, and Open vs. Closed Primaries

We have assumed so far that all voters are either Democrats or Republicans, and voters vote in their party's primaries. However, in reality, there are different primary systems, and many voters are independents. In open primaries, all voters can vote in either primary, but each voter can only vote in one primary. In semi-closed primaries, independents can choose which primary to vote in while all other voters can only vote in the primary of the party they are registered with. In closed primaries, voters must be registered with a party to be eligible to vote in that party's primary.

To compare these different primary systems to each other and RCV, fix any continuous distribution of voter ideology F on $[0, 1]$ and assume that all voters to the left of some threshold $a < M_F$ are registered Democrats, all to the right of some threshold $b > M_F$ are registered republicans and independents are the ones in $[a, b] \subset [0, 1]$. Assume that in open and semi-closed primaries, independents to the left of M_F vote in the Democratic primary while those to the right of M_F vote in the Republican primary. Under these assumptions this extended model with independents is practically the same as the baseline model since all voters to the left of M_F will vote in the Democratic primary while all to the right will vote in the Republican primary, as assumed in the baseline setting. Thus, the general election winners under open and semi-closed primaries, denoted $W_{\text{CES-O}}$ and $W_{\text{CES-SC}}$ respectively, will be the same as the election winner W_{CES} in the baseline model under CES.

With closed primaries, however, we should expect RCV to result in lower extremism relative to CES even more often than in our baseline model which does not feature independents. RCV will generate the same winner, with or without independents, but a more extreme distribution of voters selects the CES primary winners, resulting in weakly more extreme primary winners. However, it is not clear that the general election winner under closed primaries would always be less extreme than in open primaries. In particular, the nomination of a more extreme candidate in one party's primary could result in a moderate from the other party winning the general election who would otherwise lose to the party's nominee under open primaries. This open primary nominee may be more extreme than the moderate who wins even if they are less extreme than the closed primary nominee.

For this reason, we run similar simulations as we did for the baseline model under the assumption of closed primaries. For each number of candidates from 3 to 100, we simulate 5000 random voter distributions. We draw the left bound of independents a uniformly from $[0, M_F]$ and the right bound b uniformly from $[M_F, 1]$. For each distribution, we run 5 elections, with candidates randomly drawn from the distribution. However, in our simulations we were unable to generate a single case in which closed primaries produced a more moderate winner than open primaries, indicating that RCV will continue to result in even lower extremism relative to CES under closed primaries.

4.2 District Tilt

Another assumption that we have maintained is that exactly half of the electorate votes in the Democratic primary and the other half votes in the Republican primary with the median of the distribution of ideology F being the threshold that separates Democrat voters from Republican voters. This assumption may be reasonable for "swing districts" but there are many districts that tilt either towards the Republicans or the Democrats.

We now examine what happens if a district is tilted towards one of the parties in the sense that some other centile p of F rather than the median divides Democrats from Republicans. For example, if $p = 0.60$, then all voters to the left of the 60th percentile of F will vote in the Democratic primary under CES, while voters to the right of the 60th percentile of F will vote in the Republican primary. This represents a district that is 60% Democrat, and hence tilts towards the Democrats. How is the relative performance of RCV over CES in selecting moderate candidates affected by district tilt?

To address this question, we continue to define extremism as the distance of a candidate's location to the median of F, even though now the median is no longer the partisan dividing point for modeling who votes in which primary. Without loss of generality, assume that the district is tilted towards the Democratic party. We consider six cases corresponding to $p \in \{0.50, 0.60, 0.70, 0.80, 0.90, 1.0\}$ and repeat the simulations that used to generate Figure 1. We find that in general, as tilt increases, the share of elections where RCV extremism is less than or equal to CES extremism increases. This makes sense: The further p is from the median, the more distorted the CES primaries become, resulting in a more extreme winner, while district tilt has no impact on the RCV winner. Results are reported in Figure 2.

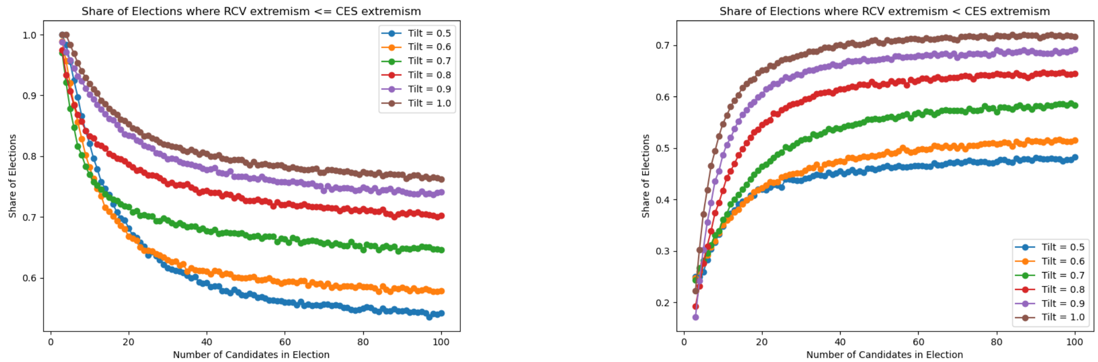


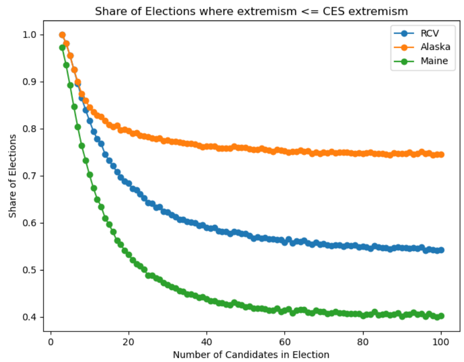
Figure 2: The figure on the left shows the percent of times that the winning candidate under RCV is at least as moderate as the winner under CES (the "RCV no worse" case) and the figure on the right shows the percent of times that the winning candidate under RCV is more moderate than CES (the "RCV better" case) as a function of the number of candidates in an election, across various values for p. For each election in the simulation we take an independent random draw of a cubic spline distribution (see Appendix A for details) and draw the platform locations of the candidates randomly and independently from the same distribution. For each number of candidates from 4 to 100, we simulate 10,000 voter distributions and 5 elections per distribution, and report the average performance of RCV relative to CES.

5 Other Implementations of RCV

For the analyses conducted so far, we have modeled the implementation of RCV as a blanket nonpartisan general election election system: all candidates, whether or not they affiliate with a party, directly run against each other in a general election in which RCV is used to elect the winner. We refer to this system simply as "RCV." In many local elections in the U.S., RCV is implemented exactly this way. Oakland, CA is a notable example, in which the 2010 mayoral election there resulted in a candidate winning under RCV that would not have won under the same tally if plurality rule had been used. Other examples include municipal elections in Aspen, CO, Portland, ME, and Minneapolis, MN.

However, in many American jurisdictions that have adopted RCV, its implementation is different. For example, in statewide-elections in Maine, RCV is implemented first to nominate candidates in each of the Republican and Democratic primaries, and then the nominees run against each other and possible independent challengers in a general election where RCV is used again to elect the winner. If there are no independent candidates, then of course RCV and plurality will coincide in the general election round. We refer to this as the "Maine system," though it is also implemented in the same or similar ways elsewhere, e.g. municipal elections in New York City, where voters can rank up to five candidates in the primaries and RCV is not used in the general election. (The Maine system does not restrict the number of candidates that voters can rank.)

Another case is that of Alaska, where all candidates first run against each other in a single blanket nonpartisan primary and plurality is used to advance the four best performing candidates for the general election, in which they run against each other under RCV. We refer to this as the "Alaska system," and note that Nevada has taken the first steps to put



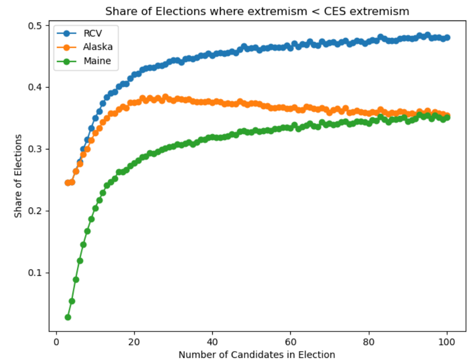


Figure 3: The figure on the left shows the percent of times that the winning candidate under an RCV variant is at least as moderate as the winner under CES (the "no worse" case) and the figure on the right shows the percent times that the winning candidate under an RCV is more moderate than the winner under CES (the "worse" case), as a function of the number of candidates in an election, across the three RCV variants. For each election, we take an independent random draw of a cubic spline distribution (see Appendix A for details) and draw the platform locations of the candidates randomly and independently from the same distribution. For each number of candidates from 4 to 100, we simulate 10,000 voter distributions and 5 elections per distribution, and report the average performance of RCV relative to CES.

in place the same system but in which the top five performing candidates advance to the general election where RCV is implemented.

In this section, we investigate how these systems compare to our implementation of RCV as a single-stage blanket nonpartisan general election system, and to CES. To do so, we simulate the Maine, Alaska, RCV, and CES systems. In each case, for each number of candidates from 4 to 100, we randomly draw 10,000 distributions and run five elections per distribution. The simulation results are reported in Figure 3, which shows that all of the RCV variants generate less extreme winners than CES. However, there are notable differences between the variants. The Maine system results in more extreme winners than the other systems. This is not surprising since RCV has election-effects only in the primaries. (With only two candidates running in the general election, RCV and plurality yield the same winner.) The primary winners under RCV may be more moderate than they are under plurality, but are likely to be more extreme than under a single stage non-partisan general election in which RCV is used.

Interestingly, the Alaska system, when compared directly to the standard RCV system, results in lower extremism more frequently, as shown in Figure 4. We hypothesize that this is the result of the following property of ranked-choice vs. plurality voting: while moderate candidates have to split their electoral support with neighboring moderate candidates on both sides, the most extreme candidates are able to capture all of the more extreme voters. Under RCV, an extreme candidate can use this extremist base to stay just above the cutoff for removal; if they are able to do this for sufficiently many elimination rounds, it is possible that they might end up winning. This final win occurs less often under RCV than under

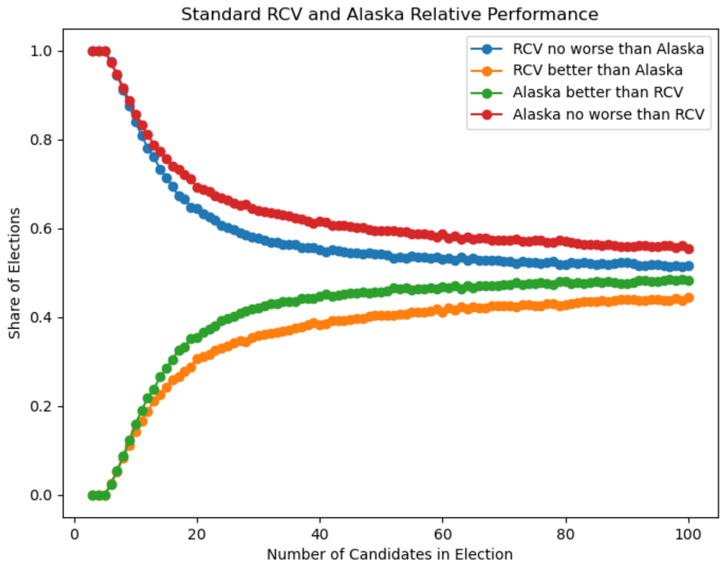


Figure 4: The figure shows a direct comparison between the extremism of standard RCV system and the Alaska system, as a function of the number of candidates in an election. For each number of candidates from 4 to 100, we simulate 10,000 voter distributions and 5 elections per distribution, and report the average performance of RCV relative to CES.

plurality (which is why RCV decreases extremism over plurality). In distributions where most of the weight is near the center, extreme candidates struggle to get over the hump. However, it is one of the main ways that extreme candidates can win under RCV. Pruning the candidate pool with plurality removes the candidates' ability to limp along via an extreme electoral base, leading to a decrease in the expected extremism.

6 Final Remarks

One of our key assumptions is that voters vote sincerely, and in the primaries they vote myopically. In our judgement these are reasonable assumptions for a large electorate.[5] Recent work by Eggers and Nowacki (2021) shows that although there are opportunities for strategic voting under instant runoff rules (of which RCV is one variant), the benefits tend to be very small (even with three candidates) mainly because of the many possible ways a voter can be pivotal. However, there is quite a bit of evidence that voters do vote strategically in plurality based systems, and some suggestive evidence that they vote strategically in other

[5]For a justification of sincere voting in large elections—albeit in a different context—see Acharya and Meirowitz (2017). In addition, with a continuum of voters, no voter can influence the outcome of the primary or general election, and so strategic voting makes little sense in our model. The assumption of myopic voting in the primaries may be hard er to justify, but Hall (2015) shows that voters regularly elect extremists in primaries despite it being the case that they are substantially less electable than moderates in the general election—a finding that is consistent with myopic voting, and seemingly inconsistent with strategic/instrumental non-myopic voting.

electoral systems as well.[6] Thus, an important avenue for future research is to develop models that compare RCV and CES accounting for strategic voting incentives.

Another important assumption in our analysis is that the same set of candidates are drawn iid from the distribution of voter ideology under RCV and CES. (We are agnostic as to exactly how many candidates run, which is why we study many cases.) The assumption of a common distribution of voters and candidates is based on the idea that more candidates will emerge from segments of the voter distribution that have a larger mass of voter ideology than those that are sparser. However, in reality, which politicians enter and what platforms they take are strategic choices that depend on political incentives that are themselves governed by the electoral system. Recent work by Buisseret and Prato (2022) studies these strategic choices for the case of three candidates. They show that RCV may under-perform in selecting moderate winners even in the three candidate case. The difference between their results and ours arises mainly from the fact that their model does not compare RCV directly to CES, and voters may abstain while candidates strategically choose their positions taking into account the possibility of voter abstention.

Other works such as Callander (2005) and Dellis et al. (2017) study the strategic entry decision under runoff procedures, building on the foundations of strategic entry models, e.g. Osborne and Slivinski (1996). These papers focus mainly on equilibria that feature at most three candidates, compare the runoff procedures to pluarlity rule. Modeling assumptions that generate robust equilibria with only up to three candidates have limited appeal, however, given that many RCV elections feature more than three candidates. The 2021 New York City Democratic mayoral primary, for example, had eight candidates on the ballot and voters were allowed to rank up to five of them. Given this, it seems desirable to develop RCV models that generate equilibria with many (i.e., more than 3) candidates, for studying and comparing extremism across the electoral systems.

Our attempts to develop such models, however, have failed to generate any sharp and robust predictions about the number and platform locations of the candidates that enter. In citizen-candidate models, it is well-known that the existence and structure of equilibria are both highly sensitive to parametric assumptions. However, in our RCV setting, we have found that even under specific paramteric assumptions the structure of equilibria can take almost any form, making it impossible to make comparisons across electoral systems.

Therefore, given the challenges to modeling strategic candidate entry for a system like RCV, we take the assumption that we have made in this paper (that some number of candidates are drawn randomly from the same distribution of voters) to be a useful starting point. This assumption has its obvious shortcomings, and more work needs to be done developing tractable models featuring strategic candidates that yield sharp and robust predictions; but despite these shortcomings, one important contribution of our work is that our publicly available code can be adapted to simulate elections under different assumptions about voter ideology, and the number and platform locations of the candidates. If future work develops sensible ways of modeling strategic entry and deriving sharp and robust predictions, that work can apply our code to compare political extremism across electoral systems.

[6] For evidence of strategic voting in plurality based systems, see Hall and Snyder (2015), Fujiwara et al. (2011), Kawai and Watanabe (2013), and Eggers and Vivyan (2020), and for some suggestive evidence of strategic voting across other electoral systems, see Cox (1997).

References

Abramowitz, A. I. and K. L. Saunders (2008): "Is polarization a myth?" *The Journal of Politics*, 70, 542–555.

Acharya, A. and A. Meirowitz (2017): "Sincere voting in large elections," *Games and Economic Behavior*, 101, 121–131.

Buisseret, P. and C. Prato (2022): "Politics Transformed? How Ranked Choice Voting Shapes Electoral Strategies," .

Callander, S. (2005): "Duverger's hypothesis, the run-off rule, and electoral competition," *Political Analysis*, 13, 209–232.

Cox, G. W. (1997): *Making votes count: strategic coordination in the world's electoral systems*, Cambridge University Press.

Dellis, A., A. Gauthier-Belzile, and M. Oak (2017): "Policy Polarization and Strategic Candidacy in Elections under the Alternative-Vote Rule," *Journal of Institutional and Theoretical Economics (JITE)/Zeitschrift für die gesamte Staatswissenschaft*, 565–590.

Drutman, L. (2020): *Breaking the two-party doom loop: The case for multiparty democracy in America*, Oxford University Press, USA.

Eggers, A. C. and T. Nowacki (2021): "Susceptibility to strategic voting: A comparison of plurality and instant-runoff elections," .

Eggers, A. C. and N. Vivyan (2020): "Who votes more strategically?" *American Political Science Review*, 114, 470–485.

Fiorina, M. P. and S. J. Abrams (2008): "Political Polarization in the American public," *Annu. Rev. Polit. Sci.*, 11, 563–588.

Fujiwara, T. et al. (2011): "A regression discontinuity test of strategic voting and Duverger's law," *Quarterly Journal of Political Science*, 6, 197–233.

Hall, A. and J. M. Snyder (2015): "Information and Wasted Votes: A Study of US Primary Elections," *Available at SSRN 2641884*.

Hall, A. B. (2015): "What happens when extremists win primaries?" *American Political Science Review*, 109, 18–42.

Horowitz, D. L. (2004): "The alternative vote and interethnic moderation: A reply to Fraenkel and Grofman," *Public choice*, 121, 507–517.

Kawai, K. and Y. Watanabe (2013): "Inferring strategic voting," *American Economic Review*, 103, 624–662.

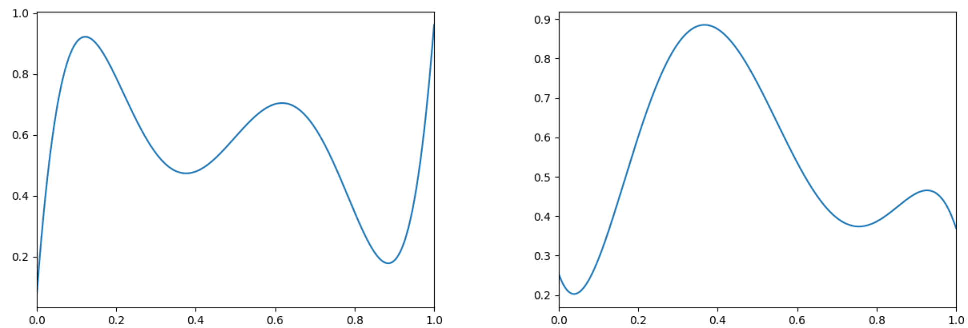
Maskin, E. (2022): "How to Improve Ranked-Choice Voting and Democracy," *Capitalism & Society*, 16.

McCarty, N. (2019): *Polarization: What Everyone Needs to Know ®*, Oxford University Press.

Osborne, M. J. and A. Slivinski (1996): "A model of political competition with citizen-candidates," *The Quarterly Journal of Economics*, 111, 65–96.

A Random Cubic Spline Distributions

To draw random cubic splines, we generate 10 random points that are equidistant in the x-axis and use a cubic spline to interpolate intermediate points, based on SciPy's univariate spline interpolation function. This allows us to create highly random and varied graphs, two examples of which are shown below. For our simulations, these splines are of course normalized so they are densities.



B Simulation Code Summary and Results

We use 5 primary functions (described below) in Python to simulate the elections:

- getVoterProportions
- findSimpleElectionWinnterValue
- findCESWinnerValue
- findRCVWinnerValue
- runGeneralDistributionVoters

The function below returns the proportion of votes that each candidate receives.

```
def getVoterProportions(prefixSum, candidates):
    leftBound = 0
    proportions = []

    for i, candidate in enumerate(candidates):
        rightBound = round((candidate + candidates[i + 1]) / 2
                   if i != len(candidates) - 1 else len(prefixSum) - 1)

        proportions.append(prefixSum[rightBound] - prefixSum[leftBound])
        leftBound = rightBound + 1

    return proportions
```

The function below returns the winner of a simple election. A simple election is an election where the winner is the candidate who receives the plurality of votes.

```
def findSimpleElectionWinnerValue(prefixSum, candidates):
    proportions = np.array(getVoterProportions(prefixSum, candidates))
    return candidates[np.argmax(proportions)]
```

The function below finds the winner under CES. It first finds the winner of the Democratic and Republican primaries, and then finds the winner in the general election.

```
def findCESWinnerValue(prefixSum, medianLoc, candidates, leftCandidates,
    ↪ rightCandidates):
    winners = []
    if leftCandidates > 0:
        winners.append(findSimpleElectionWinnerValue(prefixSum[:medianLoc],
                candidates[:leftCandidates]))

    if rightCandidates > 0:
        winners.append(medianLoc + findSimpleElectionWinnerValue(prefixSum[
            ↪ medianLoc + 1:],
            [candidate - medianLoc for candidate in candidates[leftCandidates
                ↪ :]]))

    if len(winners) == 1:
        return winners[0]

    proportions = getVoterProportions(prefixSum, winners)

    return winners[0] if proportions[0] >= proportions[1] else winners[1]
```

The function below finds the winner in an RCV election. To do so, it finds the voter share that each candidate receives and eliminates the candidates who receives the lowest share of votes, until we reach the final winning candidate.

```
def findRCVWinnerValue(prefixSum, candidates):
    if len(candidates) == 1:
        return candidates[0]

    proportions = np.array(getVoterProportions(prefixSum, candidates))

    del candidates[np.argmin(proportions)]
    return findRCVWinnerValue(prefixSum, candidates)
```

A snippet of the `runGeneralDistributionVoters` function is provided below. This function runs the simulation. In the snippet provided, we show the set-up to the simulation. To approximate a truncated normal distribution, we divide the range of voters into 50 million evenly spaced pieces, and use a normal distribution to find the "height" of each piece, thus approximating a continuous normal distribution. The variable `prefixSum` represents the CDF of the portion of the normal distribution that we are interested in, and `medianLoc` provides the median voter. Then, we run a number of "trials" (elections), based on the provided distribution. To do so, we randomly draw candidates from the distribution. At the end of the function (not shown), we return the polarization in the different election systems. Notably, this code works for any given distribution that we choose, even if the distribution is not normal; for example, for randomized cubic splines, we generate 10 random points that are equidistant in the x-axis and use SciPy's univariate spline interpolation function to create graphs that vary significantly between iterations. The complete code is available at `https://github.com/aviditacharya/RCV`.

```python
def runGeneralDistributionVoters(loc=0.5, scale=0.2, trials=500000,
                                numCandidates=NUM_CANDIDATES,
                                randomizeCandidates=RANDOMIZE_CANDIDATES,
                                leftCandidates=LEFT_CANDIDATES,
                                rightCandidates=RIGHT_CANDIDATES):

    CESPolarization = []
    RCVPolarization = []

    distribution = normalDistribution(dLoc=loc, dScale=scale)
    intervals = np.linspace(0, 1, num=NUM_GRAPH_SECTIONS)
    intervalHeights = distribution(intervals)

    prefixSum = np.append([0], np.cumsum(intervalHeights))
    medianLoc = np.searchsorted(prefixSum, prefixSum[-1] / 2)

    for trial in range(trials):
        candidates = []

        if RANDOMIZE_CANDIDATES:
            while len(candidates) < NUM_CANDIDATES:
                randomCandidate = random.randint(0, math.floor(prefixSum[-1]) *
                    ↪ 5000) / 5000.
                candidateLocation = np.searchsorted(prefixSum, randomCandidate)

                if candidateLocation != medianLoc:
                    candidates.append(candidateLocation)
    ...
```

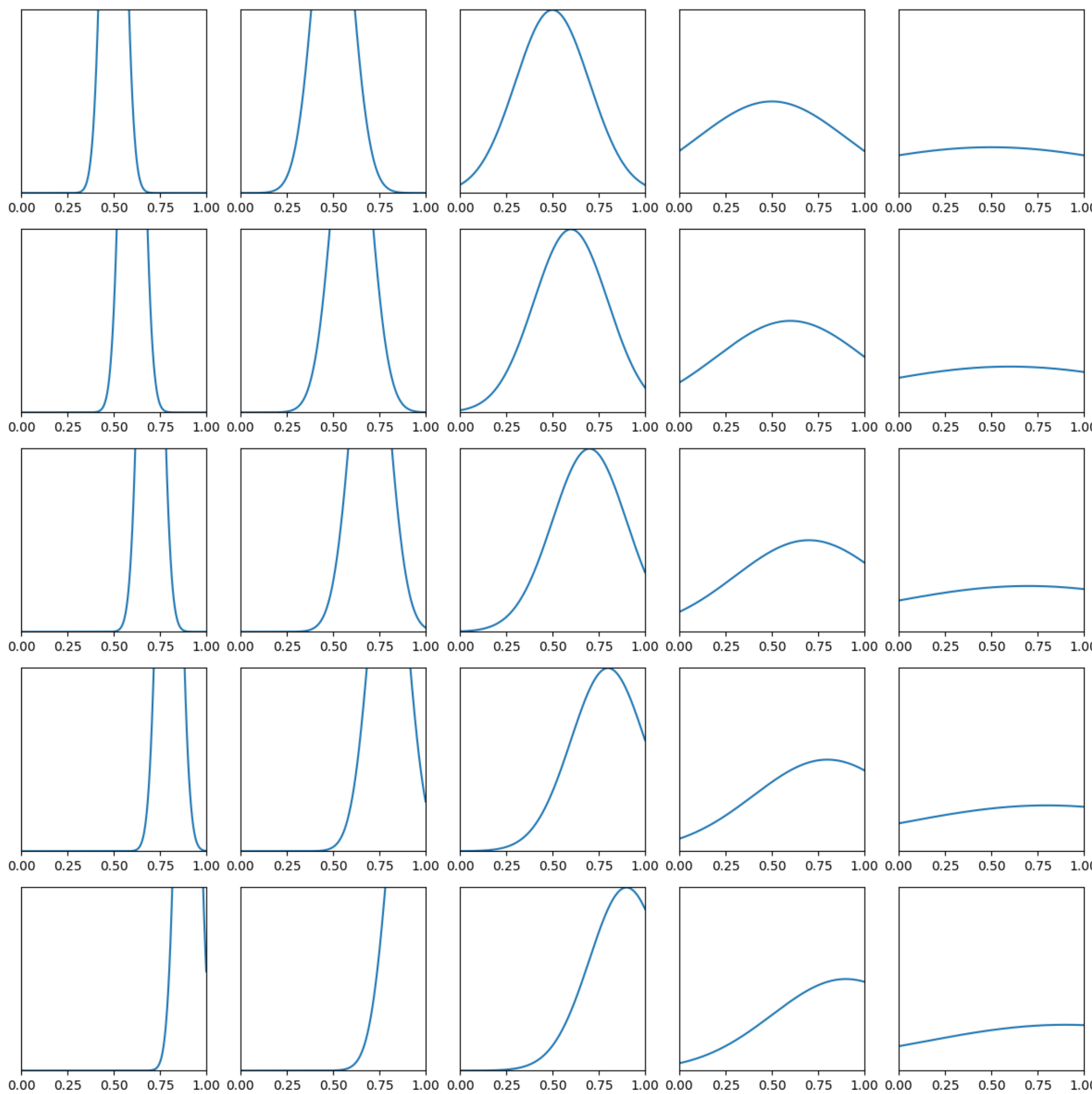


Figure B.1: Visualization of the 25 truncated normal distributions used in the simulations.

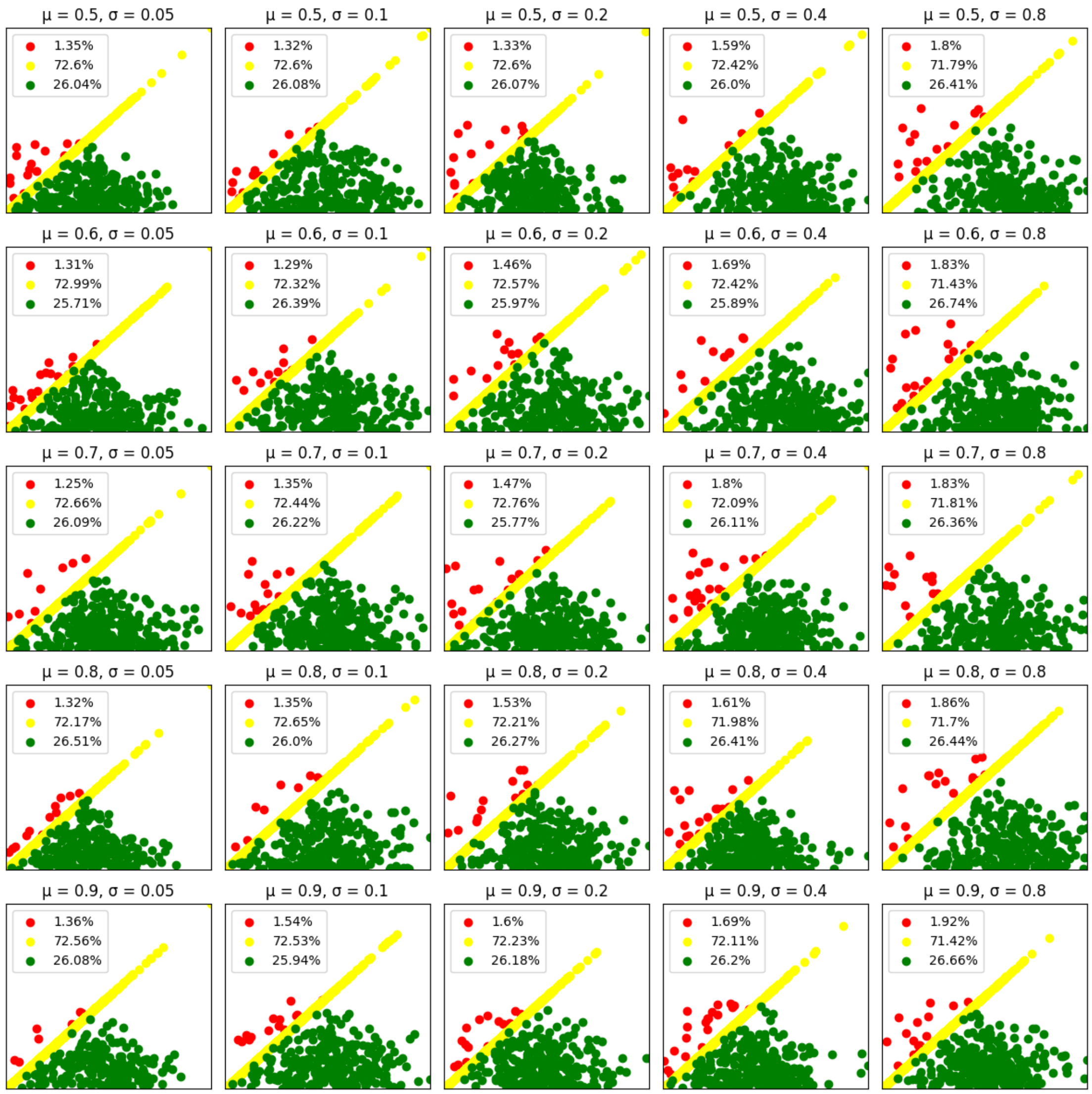


Figure B.2: Random sample of simulation results for 4-candidate elections and different choices of μ and σ, as described in the text. In each subplot, the vertical axis is our measure of extremism of the winning candidate under RCV and the horizontal axis is the extremism of the winning candidate under CES. Thus, red points correspond to elections in which RCV produced a more extreme winner than CES, green to those in which it produced a less extreme winner, and yellow to those in which the extremism of the winner was the same under both systems because the same candidate won under both. The percentages reported are based on the results from 50,000 trials for each subplot.

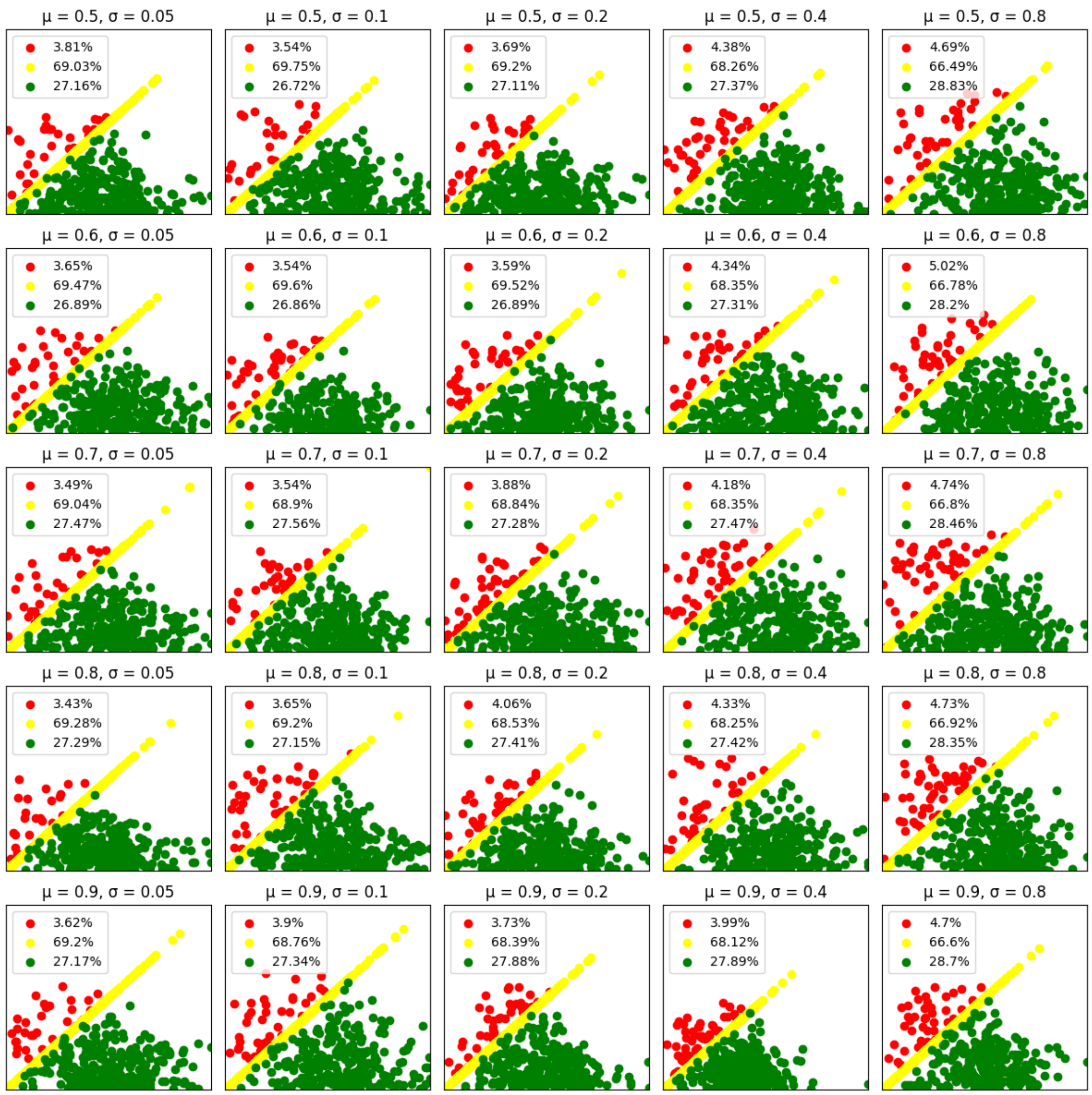


Figure B.3: Random sample of simulation results for 5-candidate elections and different choices of μ and σ, as described in the text. In each subplot, the vertical axis is our measure of extremism of the winning candidate under RCV and the horizontal axis is the extremism of the winning candidate under CES. Thus, red points correspond to elections in which RCV produced a more extreme winner than CES, green to those in which it produced a less extreme winner, and yellow to those in which the extremism of the winner was the same under both systems because the same candidate won under both. The percentages reported are based on the results from 50,000 trials for each subplot.

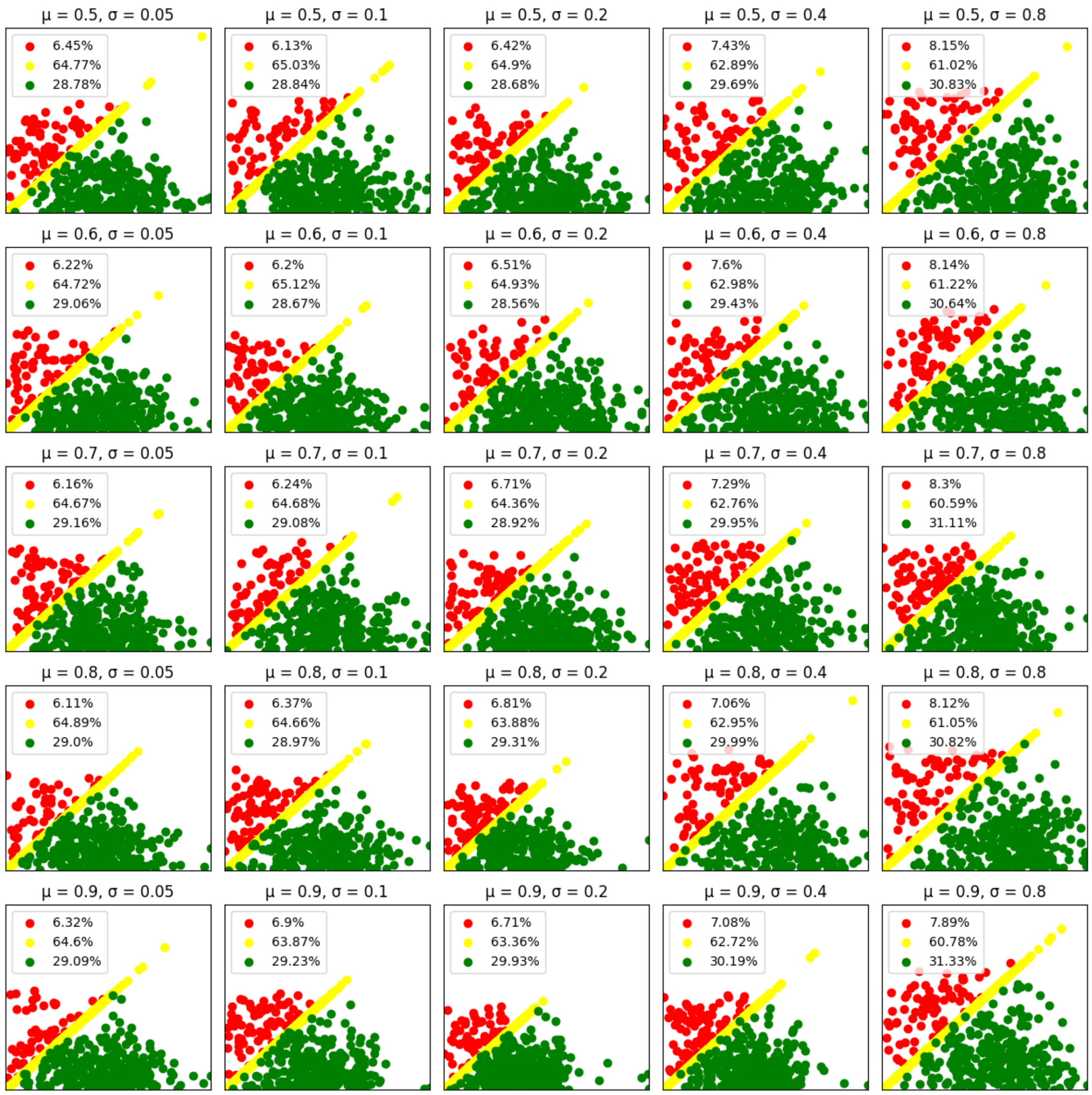


Figure B.4: Random sample of simulation results for 6-candidate elections and different choices of μ and σ, as described in the text. In each subplot, the vertical axis is our measure of extremism of the winning candidate under RCV and the horizontal axis is the extremism of the winning candidate under CES. Thus, red points correspond to elections in which RCV produced a more extreme winner than CES, green to those in which it produced a less extreme winner, and yellow to those in which the extremism of the winner was the same under both systems because the same candidate won under both. The percentages reported are based on the results from 50,000 trials for each subplot.

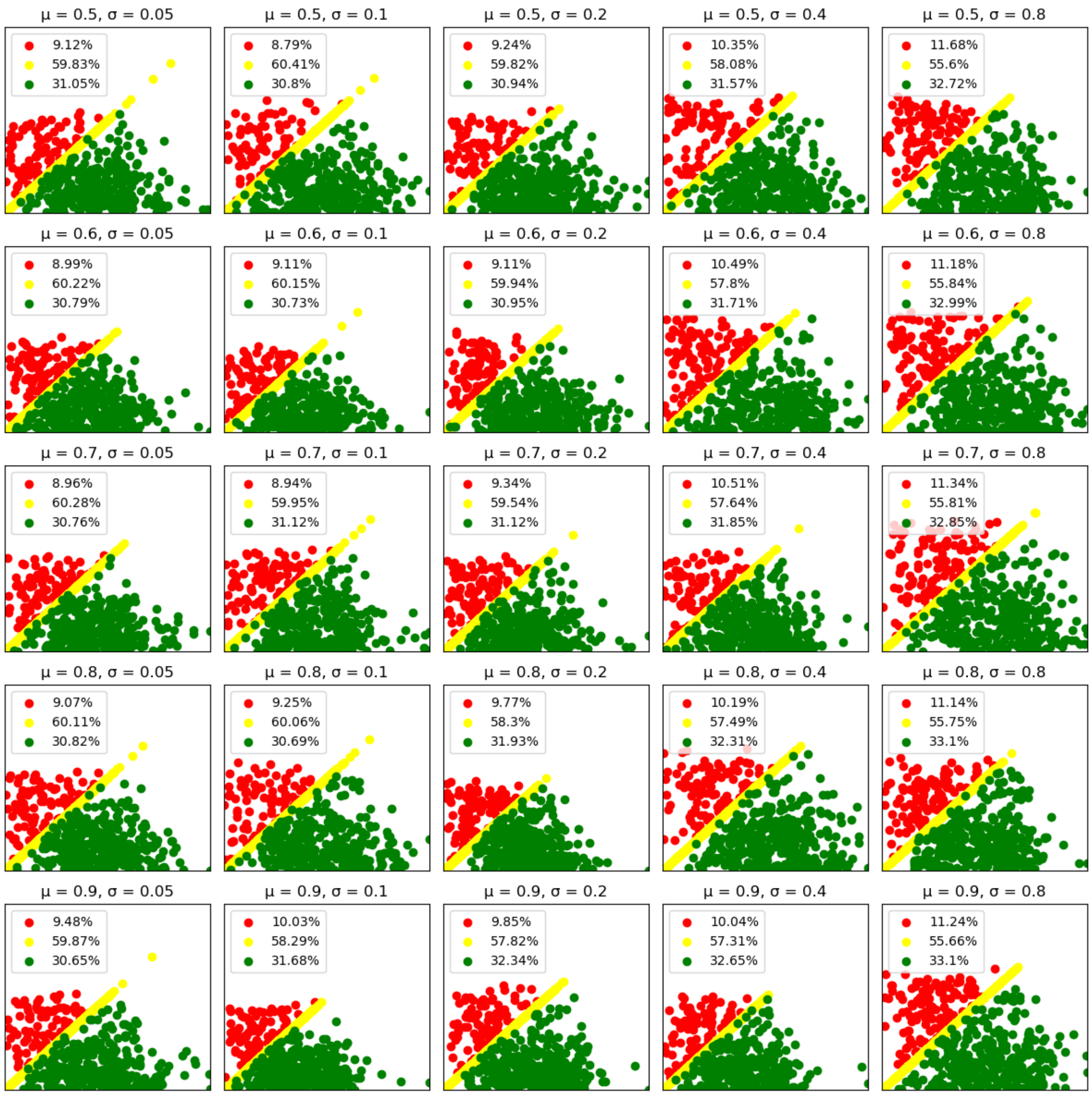


Figure B.5: Random sample of simulation results for 7-candidate elections and different choices of μ and σ, as described in the text. In each subplot, the vertical axis is our measure of extremism of the winning candidate under RCV and the horizontal axis is the extremism of the winning candidate under CES. Thus, red points correspond to elections in which RCV produced a more extreme winner than CES, green to those in which it produced a less extreme winner, and yellow to those in which the extremism of the winner was the same under both systems because the same candidate won under both. The percentages reported are based on the results from 50,000 trials for each subplot.

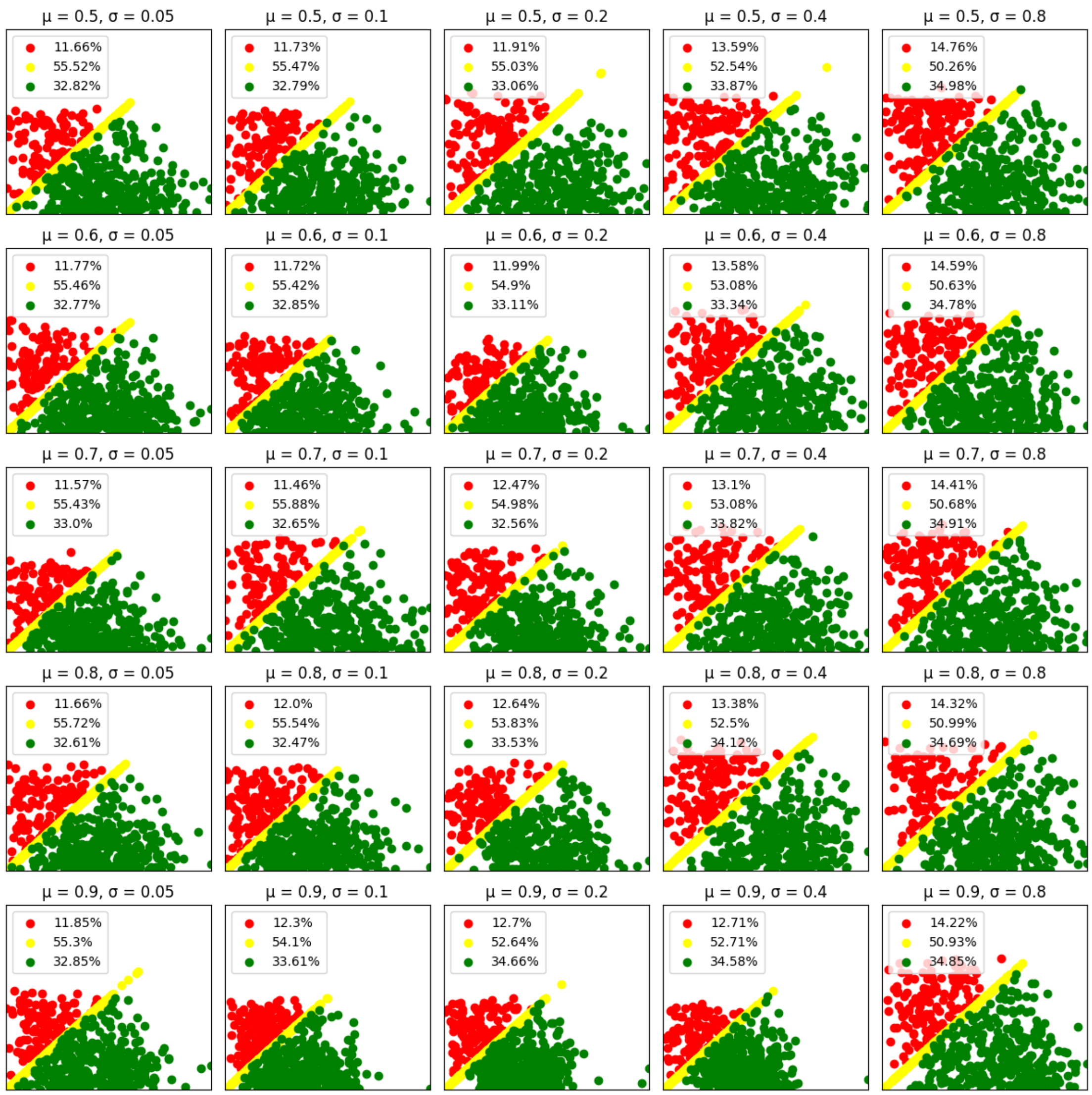


Figure B.6: Random sample of simulation results for 8-candidate elections and different choices of μ and σ, as described in the text. In each subplot, the vertical axis is our measure of extremism of the winning candidate under RCV and the horizontal axis is the extremism of the winning candidate under CES. Thus, red points correspond to elections in which RCV produced a more extreme winner than CES, green to those in which it produced a less extreme winner, and yellow to those in which the extremism of the winner was the same under both systems because the same candidate won under both. The percentages reported are based on the results from 50,000 trials for each subplot.

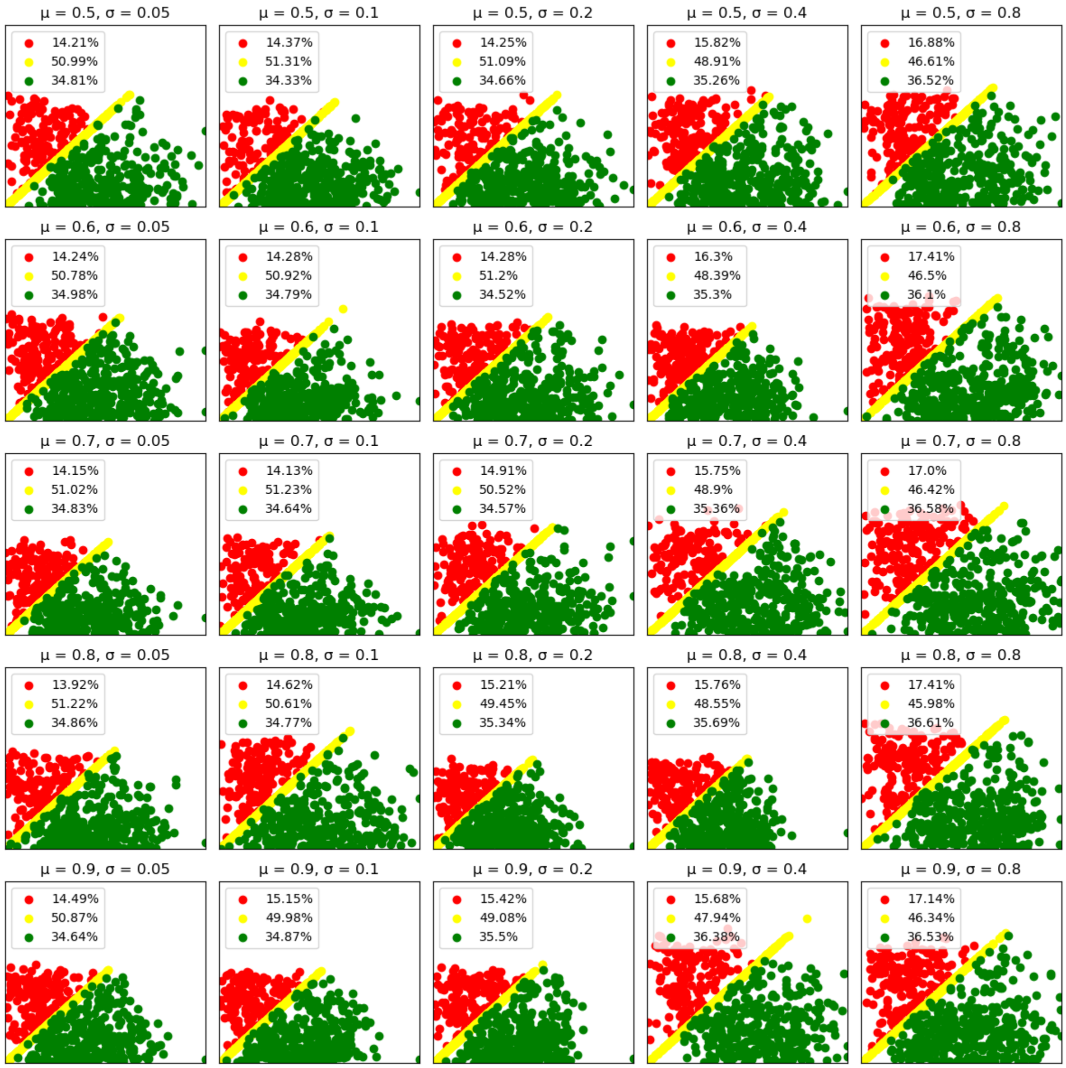


Figure B.7: Random sample of simulation results for 9-candidate elections and different choices of μ and σ, as described in the text. In each subplot, the vertical axis is our measure of extremism of the winning candidate under RCV and the horizontal axis is the extremism of the winning candidate under CES. Thus, red points correspond to elections in which RCV produced a more extreme winner than CES, green to those in which it produced a less extreme winner, and yellow to those in which the extremism of the winner was the same under both systems because the same candidate won under both. The percentages reported are based on the results from 50,000 trials for each subplot.

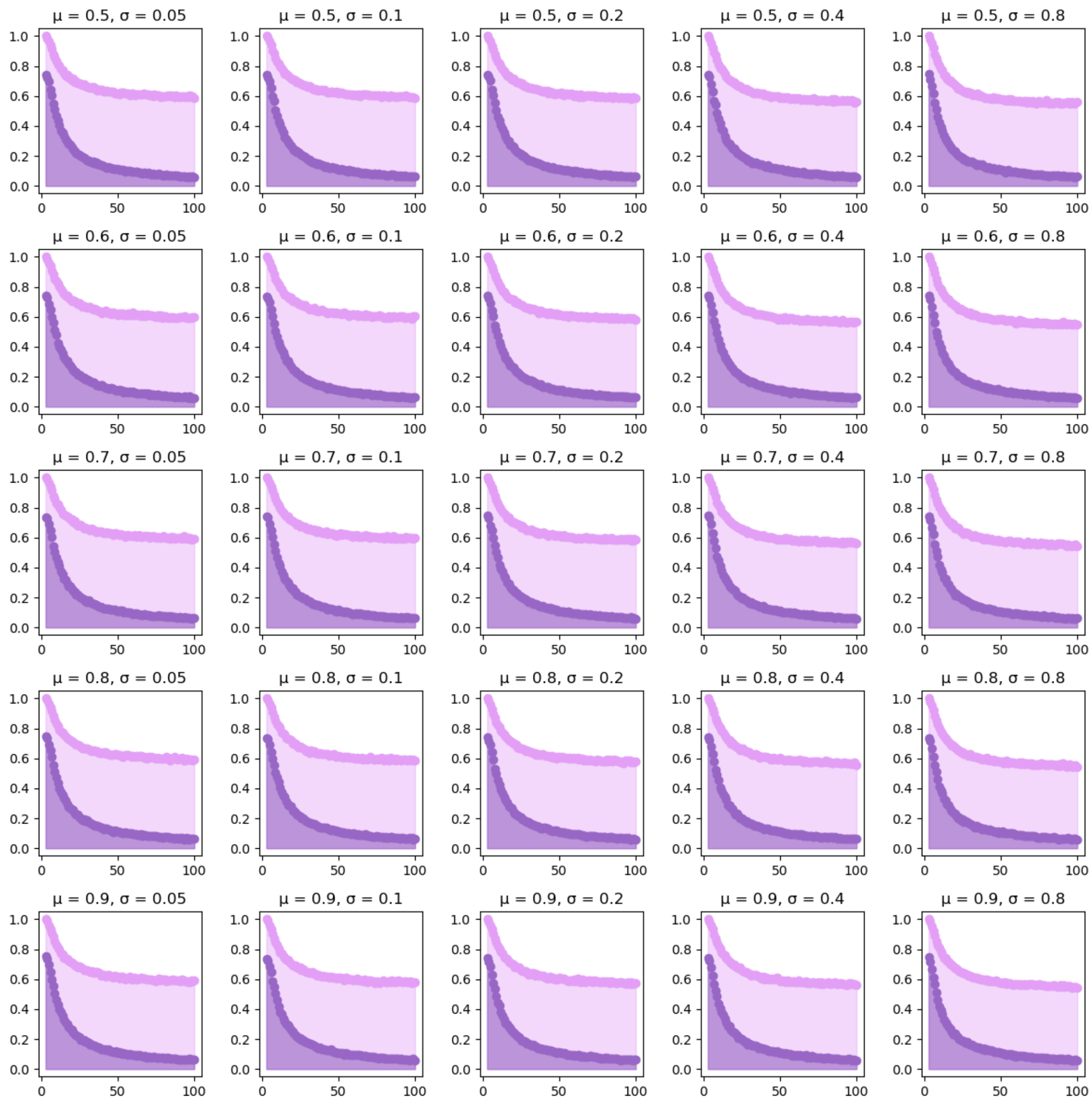


Figure B.8: Share of elections in which RCV produces more extreme and equally extreme winners as CES, plotted against the number of candidates from our simulations described in Section 3 below, across different normal distributions with mean μ and standard deviation σ.

C Exact Calculations for Uniform F

C.1 Summary of Results

In this section, we summarize the results for our exact calculations of RCV's and CES's relative performances based on polytope calculations for choice of F uniform distribution, as discussed in Section 3.3. We report results for $n \leq 7$ candidates. We also provide the code that we use to compute these results, which can also be used to compute the corresponding results for any number of candidates.

Proposition C.1. *Assuming that the distribution of voters F and candidates on $[0, 1]$ is uniform, the probabilities that RCV produces strictly less extreme candidates than CES, and that CES produces strictly less extreme candidates than RCV are given by the following table:*

n	*Ranked-Choice Voting*	*Current Electoral System*
2	$\frac{1}{2}$	0
3	$\frac{55}{96} \approx 57.3\%$	0
4	$\frac{12467}{20736} \approx 60.1\%$	$\frac{25}{2592} \approx 1.0\%$
5	$\frac{17837769667}{29059430400} \approx 61.4\%$	$\frac{41173813}{1634592960} \approx 2.5\%$
6	$\frac{657224197316733516244 9}{1059370685930395238400 0}$	$\frac{395815939018302369 1}{913250591319306240 00}$
7	$\frac{28882614014205940679827535017360562876908233 67}{46281886887158374013806521933812332560384000 00} \approx 62.4\%$	$\frac{1278539846988502331838532399784937495851918 69}{2067914094958140115510504171510763795251200 000} \approx 6.2\%$

Therefore, RCV overwhelmingly outperforms CES in terms of its likelihood of decreasing extremism, and secondly, when RCV does worse, it does worse more weakly than CES when *it* performs worse. In addition, the next proposition shows that in aggregate, there is a substantial decrease in extremism under RCV.

Proposition C.2. *Assuming that the distribution of voters and candidates on $[0, 1]$ is uniform, the expected decrease in extremism in going from CES to RCV with n candidates is given by the following table:*

n	*Decrease in Extremism*
2	$\frac{1}{12} \approx 0.083$
3	$\frac{47}{432} \approx 0.109$
4	$\frac{67129}{583200} \approx 0.115$
5	$\frac{26327232257147}{229496851584000} \approx 0.115$
6	$\frac{4071428778202291163536373 9}{3623828575137529722163200 00} \approx 0.112$

Finally, it is worth disaggregating this data slightly; that is, noting the difference in extremism in the case where ranked-choice voting performs better vs. the difference in extremism in the case where the current electoral system performs better:

Proposition C.3. *Assuming that the distribution of voters and candidates on $[0, 1]$ is uniform, the expected difference in extremism in ranked-choice voting vs. current electoral system with n candidates in the various cases for which each performs better is given by the following table:*

n	Δ *when RCV performs better*	Δ *when CES performs better*
2	$\frac{1}{6} \approx 0.167$	*N/A*
3	$\frac{94}{495} \approx 0.190$	*N/A*
4	$\frac{1440613}{7480200} \approx 0.193$	$\frac{25583}{360000} \approx 0.071$
5	$\frac{11008848973277989}{57852168094801080} \approx 0.190$	$\frac{591396779028863}{7121949721286400} \approx 0.083$
6	$\frac{43671326616076746892811314359139}{232937785407175708463577953719740} \approx 0.187$	$\frac{495588157889807957490440005457}{5424458747494321541657988857520} \approx 0.091$

The code that is used to determine these results is presented below. The code finds a complicated union of higher-dimensional polytopes representing each possible scenario, and either computes their volume or integrates over them.

C.2 Producing Polytopes

To produce these values, we used Sage's built-in models for polytope computation. First, we begin with the following straightforward utility functions:

```
# for creating sparse lists quickly
def sparse(n, *flips):
    l = [0]*n
    for f in flips:
        l[f[0]] = f[1]
    return l
```

```
# subtracts two lists elementwise; for inequalities relating vote shares
def subtract(list1, list2):
    if len(list1) != len(list2):
        raise Exception("Lists have different length.")
    return [list1[i] - list2[i] for i in range(len(list1))]
```

```
# gets permutations ending in fixed value: for enumerating all possible
    ↪ scenarios in ranked-choice voting
def perms_ending_in(k, n):
    P = Permutations(n).list()
    final = []
    for p in P:
        if p[-1] == k:
            final.append(p)
    return final
```

There are four categories of inequalities which need to be produced to define these polytopes. Firstly, there are the basic inequalities, which constrain $0 < C_1 < \cdots < C_n < 0$. These are produced by the following function:

```
# input: number of candidates
# output: inequalities for 0 < c_1 < c_2 < c_3 < ... < c_n < 1
def basic_inequalities(n):
    lower = [sparse(n + 1, [1, 1])]
    order = [
            sparse(n + 1, [i, -1], [i+1, 1]) for i in range(1, n)
            ]
    upper = [sparse(n + 1, [0, 1], [-1, -1])]
    return lower + order + upper
```

Next, there are the inequalities which determine if candidate i or candidate j is more "extreme"; these are produced by the following function:

```
# input: candidate to have lesser extremism, candidate to have more_extremism,
    ↪ number of candidates
# output: array of sets of inequalities for possibilities of less_ext < more_ext
def ext_inequalities(less_ext, more_ext, n):
    # error if necessary
    if less_ext not in [i for i in range(1, n+1)] or more_ext not in [i for i in
        ↪ range(1, n+1)]:
        raise Exception("Bad arguments for less_ext, more_ext, n")
    # throw nothing if they're equal
    elif less_ext == more_ext:
        return [[]]
    # less_ext is more left
    elif less_ext < more_ext:
        return [
                [ # same side
                    sparse(n + 1, [0, -1/2], [less_ext, 1]),
                 ],
                [ # opposite sides
                    sparse(n + 1, [0, 1/2], [less_ext, -1]),
                    sparse(n + 1, [0, -1/2], [more_ext, 1]),
                    sparse(n + 1, [0, -1], [less_ext, 1], [more_ext, 1]),
                 ]
                ]
    # more_ext is more left
    elif more_ext < less_ext:
        return [
                [ # same side
                    sparse(n + 1, [0, 1/2], [less_ext, -1]),
                 ],
                [ # opposite sides
                    sparse(n + 1, [0, -1/2], [less_ext, 1]),
```

```
            sparse(n + 1, [0, 1/2], [more_ext, -1]),
            sparse(n + 1, [0, 1], [less_ext, -1], [more_ext, -1]),
         ]
        ]
```

The final two categories of inequality relate to winning scenarios in ranked-choice voting and the current electoral system. Thus, they both rely on the following two functions, which are used to create inequalities asserting that a particular candidate wins over another candidate in a particular scenario:

```
# input: number of candidate, number of candidates, candidate set, and location
    # if location == '', then candidate set must be a list of entries (by
       ↪ default it is all candidates)
    # if location == 'l' or 'r', then candidate set must be a single number (it
       ↪  is then retrofitted into a list)
# output: set of coefficients giving the voting value
def voting_coefficients(k, n, candidate_set = None, location = ''):
    # sanitize data as described above
    if location == '':
        location = [0, 1]
        if candidate_set == None:
            candidate_set = list(range(1, n+1))
    elif location == 'l':
        location = [0, 1/2]
        candidate_set = list(range(1, candidate_set + 1))
    elif location == 'r':
        location = [1/2, 1]
        candidate_set = list(range(candidate_set + 1, n + 1))
    # error if candidate is not allowed to be tested
    if k not in candidate_set:
        raise Exception("Attempted to test candidate outside of allowed set.")
    # vacuous win
    if len(candidate_set) == 1:
        return 1
    # bottom end
    if k == candidate_set[0]:
        return sparse(n+1, [0, -location[0]], [candidate_set[0], 1/2], [
            ↪ candidate_set[1], 1/2])
    # top end
    if k == candidate_set[-1]:
        return sparse(n+1, [0, location[1]], [candidate_set[-1], -1/2], [
            ↪ candidate_set[-2], -1/2])
    # center
    else:
        ind = candidate_set.index(k)
```

```
        return sparse(n+1, [candidate_set[ind-1], -1/2], [candidate_set[ind+1],
            ↪ 1/2])

# input: number of candidate to win, number of candidate to lose, number of
    ↪ candidates, candidate set, and location
     # if location == '', then candidate set must be a list of entries (by
         ↪ default it is all candidates)
     # if location == 'l' or 'r', then candidate set must be a single number (it
         ↪  is then retrofitted into a list)
# output: set of coefficients giving the voting value
def voting_inequality(k1, k2, n, candidate_set = None, location = ''):
    # return the difference of the voting coefficients
    return subtract(
                    voting_coefficients(k1, n, candidate_set, location),
                    voting_coefficients(k2, n, candidate_set, location)
                    )
```

This allows us to define the final two groups of inequalities: those for voting scenarios in the current electoral system, and those for voting scenarios under ranked-choice voting. We begin with the former:

```
# input: number of candidates to be on the left, number of candidates
# output: list of inequalities fixing this number of candidates to be on the
    ↪ left
def left_primary_size_inequalities(left, n):
    if left == 0:
        return [sparse(n+1, [0, -1/2], [1, 1])]
    elif left == n:
        return [sparse(n+1, [0, 1/2], [n, -1])]
    else:
        return [
                sparse(n+1, [0, 1/2], [left, -1]),
                sparse(n+1, [0, -1/2], [left+1, 1])
                ]

# input: winner of general election, loser of general election, number of
    ↪ candidates on the left, number of candidates
# output: set of inequalities which force said scenario
def ces_win_scenario(winner, loser, num_cands_left, n):
    # size
    size_ieqs = left_primary_size_inequalities(num_cands_left, n)
    # if there are no left candidates
    if num_cands_left == 0:
        right = winner
        # inequalities for right winner
```

```
        right_ieqs = []
        for i in range(num_cands_left + 1, n + 1):
            if i != right:
                right_ieqs += [voting_inequality(right, i, n, num_cands_left, 'r
                    ↪ ')]
        # return all inequalities
        return size_ieqs + right_ieqs
    # if there are no right candidates
    if num_cands_left == n:
        left = winner
        # inequalities for left winner
        left_ieqs = []
        for i in range(1, num_cands_left + 1):
            if i != left:
                left_ieqs += [voting_inequality(left, i, n, num_cands_left, 'l')
                    ↪ ]
        # return all inequalities
        return size_ieqs + left_ieqs
    # if there are candidates on both sides
    else:
        # left/right winners
        left = min(winner, loser)
        right = max(winner, loser)
        # inequalities for left winner
        left_ieqs = []
        for i in range(1, num_cands_left + 1):
            if i != left:
                left_ieqs += [voting_inequality(left, i, n, num_cands_left, 'l')
                    ↪ ]
        # inequalities for right winner
        right_ieqs = []
        for i in range(num_cands_left + 1, n + 1):
            if i != right:
                right_ieqs += [voting_inequality(right, i, n, num_cands_left, 'r
                    ↪ ')]
        # inequalities for final winner
        final_ieq = [voting_inequality(winner, loser, n, [winner, loser])]
        # return all inequalities
        return size_ieqs + left_ieqs + right_ieqs + final_ieq

# input: number of candidate to win, number of candidates
# output: array of sets of inequalities which force said candidate to win
def ces_win_inequalities(c, n):
    possibilities = []
    # case of all on right
    possibilities += [ces_win_scenario(c, 0, 0, n)]
```

```
    # case of all on left
    possibilities += [ces_win_scenario(c, 0, n, n)]
    # case of some on each side
    for ncl in range(1, n):
        # case where c is on the left
        if c <= ncl:
            for l in range(ncl + 1, n + 1):
                possibilities += [ces_win_scenario(c, l, ncl, n)]
        # case where c is on the right
        else:
            for l in range(1, ncl + 1):
                possibilities += [ces_win_scenario(c, l, ncl, n)]
    return possibilities
```

Finally, we conclude with the inequalities for scenarios in ranked-choice voting:

```
# input: permutation which describes the order in which candidates fall out of
    ↪ the race, number of candidates
# output: set of inequalities which force said scenario
def rcv_win_scenario(perm, n):
    # initialize variables
    cands_remaining = list(range(1, n+1))
    ieqs = []
    # go through losers
    for loser in perm:
        # go through those not losing this round
        non_losers = [i for i in cands_remaining if i != loser]
        for winner in non_losers:
            # add an inequality
            ieqs += [voting_inequality(winner, loser, n, cands_remaining)]
        # update candidates remaining
        cands_remaining = non_losers
    # return
    return ieqs

# input: number of candidate to win, number of candidates
# output: array of sets of inequalities which force said candidate to win
def rcv_win_inequalities(r, n):
    return [rcv_win_scenario(perm, n) for perm in perms_ending_in(r, n)]
```

C.3 Computing Probabilities

Computing the probability in the uniform case is fairly straightforward. Indeed, one only needs to iterate over each combination of possible scenarios for the current electoral system and ranked-choice voting, splitting them by which winner they result in (and which winner is more extreme) and tally up the volume of the polytope associated with each scenario. In code,

```
# input: number of candidates
# output: probability between 0 and 1
def uniform_probability(n):
    # probabilities of all possibilities
    prob = [0, 0, 0]
    # basic inequalities
    basic = basic_inequalities(n)
    # iterate over all possible winners for rcv
    for r in range(1, n+1):
        # iterate over all possible winners for ces
        for c in range(1,n+1):
            # report
            print("Computing case of RCV winner " + str(r) + " and CES winner "
                ↪ + str(c))
            # obtain winner scenarios for rcv
            rcvs = rcv_win_inequalities(r, n)
            # obtain winner scenarios for ces
            cess = ces_win_inequalities(c, n)
            # case where r and c are identical
            if r == c:
                for rcv in rcvs:
                    for ces in cess:
                        prob[1] += Polyhedron(ieqs = basic + rcv + ces).volume()
            # case where r and c are distinct
            else:
                # r has less extremism
                rcv_wins = ext_inequalities(r, c, n)
                for rcv_win in rcv_wins:
                    for rcv in rcvs:
                        for ces in cess:
                            prob[0] += Polyhedron(ieqs = basic + rcv + ces +
                                ↪ rcv_win).volume()
                # c has less extremism
                ces_wins = ext_inequalities(c, r, n)
                for ces_win in ces_wins:
                    for rcv in rcvs:
                        for ces in cess:
                            prob[2] += Polyhedron(ieqs = basic + rcv + ces +
                                ↪ ces_win).volume()
    # return
    return [i * factorial(n) for i in prob]
```

C.4 Computing Expected Extremism

This is very similar to the case of computing probabilities: here, instead of finding the volume over a polytope, we integrate the multivariate polynomial corresponding to the difference in extremism. The only added difficulty is in computing these polynomials, for which an auxiliary function is used:

```
# input: number of candidates, candidate to have lesser extremism, candidate to
    ↪ have more_extremism
# output: polynomial functions for difference of extremism in case less_ext <
    ↪ more_ext
def ext_diff_funcs(less_ext, more_ext, n):
    # initialize variables
    x = polygen(QQ, ['x' + str(i) for i in range(n)])
    # error if necessary
    if less_ext not in [i for i in range(1, n+1)] or more_ext not in [i for i in
        ↪ range(1, n+1)]:
        raise Exception("Bad arguments for less_ext, more_ext, n")
    # zero out if equal
    elif less_ext == more_ext:
        return 0
    # less_ext is more left
    elif less_ext < more_ext:
        return [
                x[more_ext-1] - x[less_ext-1], # same side
                x[less_ext-1] + x[more_ext-1] - 1 # opposite sides
                ]
    # more_ext is more left
    elif more_ext < less_ext:
        return [
                x[less_ext-1] - x[more_ext-1], # same side
                1 - x[less_ext-1] - x[more_ext-1] # opposite sides
                ]

# input: number of candidates
# output: expected extremism difference between 0 and 1
def expected_extremism_difference(n):
    # expectation
    E = [0, 0]
    # basic inequalities
    basic = basic_inequalities(n)
    # iterate over all possible winners for rcv
    for r in range(1, n + 1):
        # iterate over all possible winners for ces
        for c in range(1, n + 1):
            if r != c:
```

```
            # report
            print("Computing case of RCV winner " + str(r) + " and CES
                ↪ winner " + str(c))
            # function
            # obtain winner scenarios for rcv
            rcvs = rcv_win_inequalities(r, n)
            # obtain winner scenarios for ces
            cess = ces_win_inequalities(c, n)
            # r has less extremism
            rcv_wins = ext_inequalities(r, c, n)
            rcv_diffs = ext_diff_funcs(r, c, n)
            for i in range(len(rcv_wins)):
                for rcv in rcvs:
                    for ces in cess:
                        Pr = Polyhedron(ieqs = basic + rcv + ces + rcv_wins[
                            ↪ i])
                        if Pr.volume() > 0:
                            I = Pr.integrate(rcv_diffs[i])
                            E[0] += I
            # c has less extremism
            ces_wins = ext_inequalities(c, r, n)
            ces_diffs = ext_diff_funcs(c, r, n)
            for i in range(len(ces_wins)):
                for rcv in rcvs:
                    for ces in cess:
                        Pc = Polyhedron(ieqs = basic + rcv + ces + ces_wins[
                            ↪ i])
                        if Pc.volume() > 0:
                            I = Pc.integrate(ces_diffs[i])
                            E[1] += I
    # return
    return [i*factorial(n) for i in E]
```