How To Hold A Vote When Candidates Are Too Many ? A Scalable And Random-Based Election

Pierre-Louis Guhur,* David Guez,* Thibauld Favre,*
*LaPrimaire.org
pierre-louis.guhur@laposte.net, tfavre@gmail.com

***Abstract*—Strong non-partitions in recent democratic elections show that the panel of candidates do not convince everyone, and therefore the outcome is not fully representative of the population. In order to provide a better outcome, an election must be done with more candidates. On the other side, when the profiles or the programs of the candidates are diverse, it is unrealistic to demand from voters that they review all candidates. In such a case, medias or primaries filter out some candidates, in order to clarify the political landscape, and the situation does not get improvements. In this article, we address this specific issue and we provide a method to hold elections when the number of candidates is too big for human review. Our new election process is called *election with random sets*. It consists in selecting randomly a set of candidates for each voters. We experimentally tested the voting system, exhibited a minimal number of voters to validate it. Finally, this voting system was empirically approved in a presidential primary in France called LaPrimaire.org.**

***Index Terms*—**

I. Introduction

Election is a widely used decision-making process. The decision is made among a panel of candidates by voters according to a voting system. Such candidates could be one of the voters in the case of a Presidential election, but they can also be ideas or projects in the case of a tender process. Voters are often populations or experts, but are not necessary humans: in computer science, abstract entities (for example trees in the random forest algorithms) can vote.

A voting system measures which candidate is preferable for the voters. The most used method for electing is the majority rule: each voter gets a unique vote, and the winner is the candidate that receives the most votes. Because voting systems suffer from several issues, several authors [1], [2], [3] provided new ways to cast votes.

However, none of them focuses on the scalability of the number of candidates: how to deal with a large number of candidates. Scalability is paramount in an election, because having more candidates lead to more diverse profiles, and might provide a more suitable outcome of the election. Existing voting systems are not scalable for practical reasons: voters can not examine in details the programs and the profils of dozens or even hundreds of candidates.

In this article, we address a solution to this issue. Because a voting system can not ask voters to vote for all candidates, we designed a new election process, that we called *election with random sets*. In this process, a voter casts his/her vote(s) only for a set of candidates, that is randomly selected for each voter, according to the rule of a voting system qualified of *embedded*. In a second phase called *aggregation*, the votes obtained by all the sets are reunited, and the oucome of the election is decided.

Our *election with random sets* is intrinsically scalable, as the size of the set is chosen as the number of candidates that a voter is able to study (which depends on the context of the election). Moreover, its random initialization makes manipulation hard, according to Conitzer and Sanholm [3]. At the same time, one can wander if the outcomes are still representative of the voters' preferences. We simulated our election process, and exhibit parameters that guarantee the representativeness of the outcomes. Eventually, this voting system was employed in a Presidential primary in France organized by LaPrimaire.org.

In the Section II, we detail the vote with random sets. In the Section III, we show that the election is not biased with a minimal number of voters. We also lay out the perspective of the vote in the aftermath of LaPrimaire.org.

II. Vote With Random Sets

In this section, we detail our voting system for an election with an high number of candidates. Because the voters can not be asked to review all candidates, we design a system to ask them to review only a part of the candidates.

A. Principle

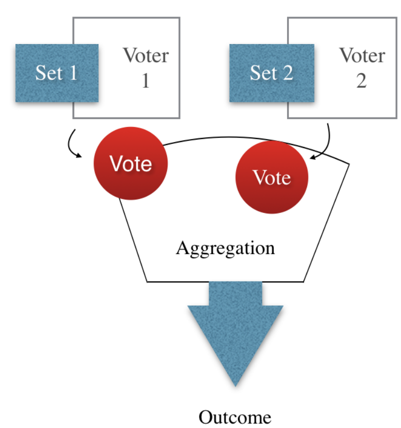

Figure 1. Scheme of the vote with random sets

The principle is explained in Figure 1. It consists in three phases:

- first at all, the voter receives a set of a fixed number of candidates;
- secondly, the voter gives his/her votes for the candidates in his/her set, according to the rules of a second voting system;
- lastly, all the votes are aggregated. The aggregation is obtained by scaling the votes by their frequency of apparition in the sets.

The number of candidates is determined by the maximum number of candidates that a regular voter can review. It all depends on the context of the vote. For example, the jury of a competition is able to examine more candidates than a citizen during a Presidential election. In the context of LaPrimaire.org, the number of candidates was restrained to 5, because most of the voters knew little about the candidates before the election.

B. Definition of the requirements

In our election process, we want to guarantee that the outcome of the vote follows voters' preferences, or in the opposite direction, that the vote is not biased. A measure to jauge the skewness of the election is difficult to choose. Eventually, we defined three requirements that need to be followed to obtain an impartial process.

1) Each candidate appears the same number of times in sets.
2) Each candidate is opposed the same number of times to any other candidate.
3) The aggregation process does not modify the outcome of the election realized in the ideal conditions.

To jauge if a requirement is followed or not, we measure it with a statistical function. This function has to remain below a certain tolerance. For the two first requirements, the *coefficient of variation* of the number of occurrences of each candidate was employed as the statistical function. The coefficient of variation (also called the relative standard deviation) is defined as the ratio between the biased standard deviation and the mean μ of a distribution:

$$Cv(x) = \frac{\sqrt{\frac{1}{N}\sum_{i=1}^{N}(x_i - \mu)^2}}{\mu}, \quad \mu = \frac{1}{N}\sum_{i=1}^{N} x_i$$

C. First requirement

Impartiality in candidates requires that each candidate appears the same number of times in sets. A slightly difference can though occur. Indeed, the aggregation phase compensate the difference in the frequency of apparition, because the frequency of apparition is known and results are scaled by the frequency of apparition. We impose a maximum value of $\epsilon_1 = 0.1$ for $requirement_1$, the coefficient of variation of the number of occurrences of each candidate.

D. Second requirement

Some voting systems suffer from the *Condorcet paradox*, and thus have no *Condorcet winner* [4]. A Condorcet winner is a winner of an election that would win a two-candidate election against any other candidate. The well-used majority rule can have no Condorcet winner. For example, Al Gore lost the 2002 US Presidential elections against George W. Bush, while he would have won in a two-candidate election against George W. Bush and against Ralph Nader.

When such voting systems are employed as the embedded voting system, the election might be unfair for some candidates when the second requirement is not followed. As an example, we consider an election using the majority rule, and three candidates, called A, B and C, who are the most likely candidates to win the election. If A and B are always in the same set, while C is always in a different set, A and B are less likely to receive some votes than C.

The second requirement is also paramount, when the voters' judgement are influenced by their last judgments. This cognitive biased is called the decoy effect [5]. For instance, if the embedded voting system is the majority judgement (presented in section III), voters would provide a better judgement for a candidate when s/he is in the same set than the least likely-to-win candidates, than when s/he is in find the same set than the most likely-to-win candidates.

Consequently, we consider $requirement_2$, the coefficient of variation of the number of times that a candidate is opposed to another candidate as the statistical function for the second requirement. We want it to remain below $\epsilon_2 = 0.1$.

E. Third requirement

We define the ideal conditions of the third requirements as an election where no issue impacts the judgment of the voters. Voters review each candidate similarly and are not affected by any kind of filter. An election might not end up with only one winner, but several (for example for two round elections or for a competitive examination). For an outcome with $Nwinners$, we measured how many candidates that are ranked in the top $Nwinners$ of the perfect election were not in the top $Nwinners$ of our voting system. This defines what we call the error of the election. As previously, we expect that the error is in average below a tolerance $\epsilon_3 = 0.1$.

The two first requirements will be fulfilled by the algorithm to build the set, whereas the last requirement will be translated into a requirement in the number of voters.

F. Building the sets

Having random sets prevent a cheater to corrupt voters to cast their votes for a special candidate, because the cheater does not know if this candidate will appear in their sets. But, if the random sets are not completely patternless, the cheater can be able to predict the next sets, within a certain confidence, by observing the last sets.

Moreover, the sets have to build *online*, namely when a voter presents him/herself to the election. If the sets are built *offline*, some events might unbalance the distribution of sets among the candidates and the two first requirements would not be satisfied. Examples of such events are: many voters that were supposed to receive a set with a certain candidate do not show up or invalidate their vote; servers crash and all first sets containing a certain candidate are lost; a certain candidate appears many times at the beginning of the election, polls reveal it during the elections, and finally voters, that wanted to vote for this candidate, might find useless to participate to the election.

We designed an algorithm that builds the sets online, follows the two first requirements, and is patternless. When a voter asks a set of candidates, we do the following steps:

1) counting the number of occurrences N_{occ} of each candidates in the previous sets;
2) building a table that attributes to each candidate the value $1/N_{occ}^{\alpha}$ if $N_{occ} \neq 0$ and 0 otherwise;
3) dividing this table by its sum;
4) drawing N_c different candidates from this table.

The algorithm depends on the exponent α. The higher α is, the more likely a candidate, that has not been chosen in previous sets, is to be chosen of the next set. Higher values of α thus helps to fulfill the first requirement. Moreover, by leveraging the number of occurrences, all candidates will appear as likely as each other to be chosen in the next set. That is why, higher values of α helps also to satisfy the second requirement. The drawback with too high values of α is that in case of a unbalance number of occurrences, the probabilities of selecting the candidate with the lower number of occurrences override the other candidates: the process is no more patternless. Basically, the smaller value of α that fulfills the first requirement is employed. By default, α is set at 1 for the sake of simplicity.

G. *Relevant parameters*

In order to satisfy the three requirements, five parameters need to be known.

- The number of candidates N_c increases the difficulties for the aggregation phase. For high values of N_c, a voter gives a small piece of information with respect to all candidates.
- At the opposite, the number of voters helps to reconstruct the votes, because it provides more information.
- Similarly, the number of candidates by set provides more information and helps to satisfy the requirements. In the example of LaPrimaire.org, we fix it to 5 with regard of previous elections as explain in Section II-A.
- As we saw in Section II-F, α is chosen as a trade-off between patternless and the two first requirements.
- We saw also that elections might have not only one winner, by also several winners. The number of winners affects the last requirements: in LaPrimaire.org, five candidates were chosen during the first round, and the results were the same if the third candidate was ranked before of after the fourth candidate.

III. Validation of the election process

In this Section, we show how to set the 5 parameters previously defined, such as the three requirements defined in Section II are followed. First, we will show how one can increase α in order to satisfy the two first requirements. Then, we will show the last requirement can be defined as a minimum number of voters.

Scripts for recomputing the experiments are available on a git repository [6].

A. *Validation of the two first requirements*

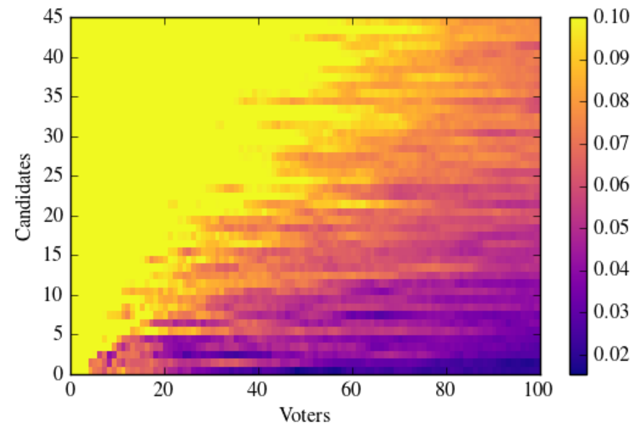

Figure 2. Representation of $requirement_1$ for $\alpha = 1$, and several number of candidates and voters. Values higher than $\epsilon_1 = 0.1$ are fixed at ϵ_1

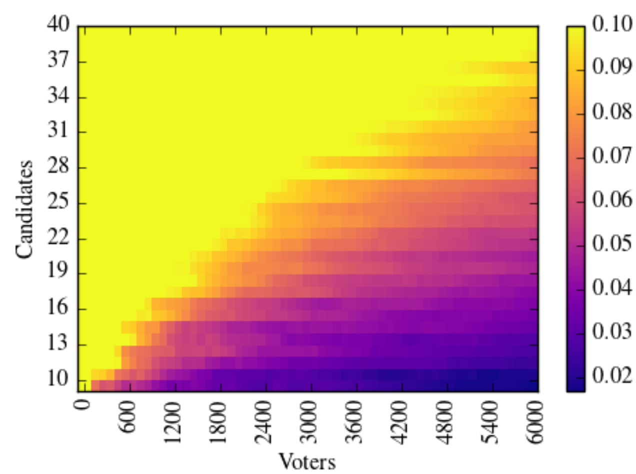

Figure 3. Representation of $requirement_2$ for $\alpha = 2$, and several number of candidates and voters. Values higher than $\epsilon_2 = 0.1$ are fixed at ϵ_2

In the Figures 2 and 3, $requirement_1$ and $requirement_2$ are varying with respect to the numbers of candidates and the numbers of voters. An high value of $requirement_1$ means that some candidates appear more often than other candidates, whereas an high value of $requirement_2$ means that some pairs of candidates appear more often in the same sets than other pairs of candidates. Results were expected in the Section II-G: the statistical functions are all the more low as many voters and few candidates participate to the election.

The organizers of an election may not be able to change the number of voters and the number of candidates. However, they can fulfill the first requirement by selecting the most suitable α. In the repository [6], the function `findMinAlpha` computes iteratively the value of α, until it finds a value that satisfies the two first requirements for a fixed number of candidates and of voters. The tolerances ϵ_1 and ϵ_2 can also be modified in `findMinAlpha`. Examples of computed values of α are presented in the Figure 4 for a variable number of voters and of candidates. α has its highest values in the corner top-left, because the requirements are all the more difficult to achieve there, according to figures 2 and 3.

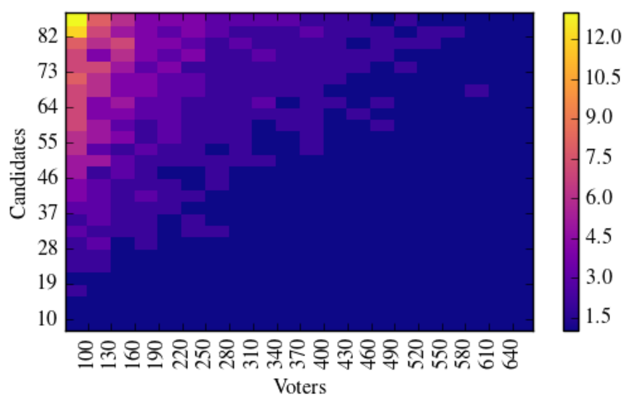

Figure 4. Value of the minimum value of α that fits the tolerances ϵ_1 and ϵ_2 for several numbers of candidates and of voters.

B. *Minimum number of voters*

To validate the third requirement, we want to compare an election with random sets and a perfect election. The perfect election is an election that is not affected by any kind of biases or manipulations. In order to simulate it, we used results from a real election made with majority judgment. That was achieved with data from OpinionWay, that organized a poll a few days before the first round of the French Presidential election in 2012, sponsored by the Parisian think-tank Terra Nova. The election gathered 10 candidates and 7 different grades and results give the percentage for each grade.

In order to simulate elections with more than 10 candidates, we measured the standard deviation σ and the mean μ for the 10 candidates, and then we drew new results from a normal law parametrized with σ and μ. Scripts for the simulation are also available on the repository [6]. We run them on a server Intel(R) Xeon(R) CPU E5-2650L v3 @ 1.80GHz with 4 cores.

The value of $requirement_3$ does not depend on α; the latter can thus be selected only with respect to the two first requirements. However, it depends on $Nwinners$, the number of selected candidates at the end of the round, on Nc the number of candidates that participate to the election, and Ne the number of voters. The organizers of the election may not be able to change $Nwinners$ and Nc, but can influence the number of voters that cast their vote. That is why, `findMinNvoters` computes the minimal number of voters that achieves the third requirement for given $Nwinners$ and Nc. In the Figure 5, the minimal number of voters evolves with N_c and $Nwinners$. It shows also that the minimum number of voters decreases with the number of winners, but increases with the number of candidates.

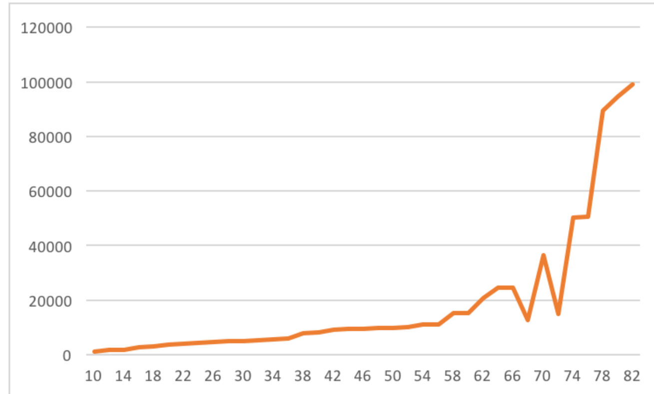

Figure 5. The minimal number of voters as a function of the number of candidates and the number of winners.

C. *Results of LaPrimaire.org*

LaPrimaire.org is the name of an online primary for the French Presidential election of 2017. It is organized by the association Democratech. Its goal is to arouse citizens' interest for public policy, to allowing them to choose a Presidential candidate and to cowrite political agenda. 215 candidates enrolled themselves, but only 16 of them gathered 500 supports required to participate. 4 of them gave up or merged before the election. The choice of applying the random sets was driven by the fact that voters knew little about the candidate before the election. From October 26 to November 1st, a first-round election occurred to select the 5 best candidates and $11,304$ voters participated. From December 16 to 31, the second round run to elect the best candidate with $32,685$ voters.

The embedded voting system was the majority judgment [7]. This voting system consists in attributing a grade between "Very good," "Good," "Fair," "Correct" and "Insufficient". The candidates are ranking with respect to their median grade. In case of equality, two candidates are separated with an algorithm called the *tie breaking*.

The function `findMinAlpha` shows that $\alpha = 1$ was enough to follow the two first requirements, while $requirement_3$ is below $epsilon_3$ for 11304 voters. As shown in Table I, results between the two rounds are similar. This can be seen as an empirical validation of the method.

Table I. Results from the election on LaPrimaire.org

Round	First round		Second round	
	Grade	Gauge	Grade	Gauge
C. Marchandise	Good	71.62%	Very good	50.64%
N. Bernabeu	Good	55.83%	Fair	72.26%
R. Revon	Good	53.65%	Good	50.24%
M. Bourgeois	Fair	66.26%	Fair	59.85%
M. Pettini	Fair	64.70%	Fair	57.94%

IV. Conclusion

We observed that elections are not scalable in their number of candidates, although having a large number of candidates can provide a better outcome. We offered a new voting system, in which voters judge only a set of few candidates. During the aggregation phase, all votes are gathered and the outcome of the election is computed. The process is based on randomness: it prevents the election from manipulations, but also helps to reconstruct the outcome.

We made simulations to verify that our voting system satisfy three requirements: 1. all candidates appear the same number of times in the sets, 2. candidates are as likely as possible to meet other candidates, 3. the outcome is almost the same than in the case of a election without any issue. Our voting system depend on a parameter α, but we provide a script to compute α from the number of candidates and the number of voters. Finally, we experimented that the third requirement is satisfy with a minimal number of voters which can easy be computed.

For a vote with strong constraints on the number of candidates and the number of voters, we observed that the size of the sets or the number of remaining candidates at the end of the election have to be increased. We think that this idea can be extended, and that for highly constraints elections, those variable can be adapted during the election, such as the three requirements are always satisfied. For example, a first round based on the first voters can select a large number of remaining candidates; then a second round reduce the number of remaining candidates; and so on, until we achieve the expected number of candidates.

REFERENCES

[1] K. J. Arrow, "A difficulty in the concept of social welfare," *The Journal of Political Economy*, pp. 328–346, 1950.

[2] E. Friedgut, G. Kalai, N. Nisan *et al.*, *Elections can be manipulated often*. Citeseer, 2008.

[3] V. Conitzer and T. Sandholm, "Complexity of manipulating elections with few candidates," in *AAAI/IAAI*, 2002, pp. 314–319.

[4] M. B. Garman and M. I. Kamien, "The paradox of voting: Probability calculations," *Behavioral Science*, vol. 13, no. 4, pp. 306–316, 1968. [Online]. Available: http://dx.doi.org/10.1002/bs.3830130405

[5] J. Huber, J. W. Payne, and C. Puto, "Adding asymmetrically dominated alternatives: Violations of regularity and the similarity hypothesis," *Journal of consumer research*, vol. 9, no. 1, pp. 90–98, 1982.

[6] "Repository for votes with random sets." [Online]. Available: https://github.com/plguhur/random-sets

[7] M. L. Balinski and R. Laraki, *Majority judgment: measuring, ranking, and electing*. MIT press, 2010.