

WORKING PAPERS

N° 17-755

January 2017

“The Likelihood of a Condorcet Winner in the Logrolling Setting”

William Gehrlein, Michel Le Breton
and Dominique Lepelley

The Likelihood of a Condorcet Winner in the Logrolling Setting*

William Gehrlein[†] Michel Le Breton[‡] Dominique Lepelley[§]

December 2016

Abstract

The purpose of this note is to compute the probability of logrolling for three different probabilistic cultures. The primary finding is that the restriction of preferences to be in accord with the condition of separable preferences creates enough additional structure among voters' preference rankings to create an increase in the likelihood that a Condorcet winner will exist with both IC and IAC-based scenarios.

Classification JEL : D71, D72.

Key Words : Condorcet, Separable preferences, Logrolling, Vote Trading.

*We would like to express our deepest gratitude to Victor Le Breton for his generous assistance in preparing R programs used to compute random approximations of the values of some of the probabilities introduced in our paper.

[†]University of Delaware, USA, wvg@udel.edu

[‡]Toulouse School of Economics and Institut Universitaire de France, France, michel.lebreton@tse-fr.eu

[§]CEMOI, Université de la Réunion, France, dominique.lepelley@univ-reunion.fr

1 Introduction

Majority voting is one of the oldest pastimes of social choice theorists starting with Condorcet (1785). It is well known that majority voting may lead to intransivities in the social preference. This situation is often referred to as a Condorcet paradox. The evaluation of the likelihood of the Condorcet paradox under various probabilistic assumptions has been the subject of an important literature (Gehrlein (2006), Gehrlein and Lepelley (2010)).

The purpose of this note is to examine the likelihood of a Condorcet paradox in a specific setting. Precisely, the set of alternatives X is given by $\{0,1\}^k$ and the preferences of the voters are postulated to be separable linear orders. This setting underlies all the analysis of logrolling and vote trading in committees/legislatures (Miller (1994)). An alternative is a sequence of k distinct bills where each bill has two possible outcomes: “passage” (1) or “defeat” (0). Separability simply means that preferences on each bundle of issues are independent of what could be decided on the remaining issues.

In that setting, there is a Condorcet winner on each individual issue. In our note, the vector whose coordinates are these local Condorcet winners is called the componentwise Condorcet winner. Of course, the componentwise Condorcet winner does not need to be a Condorcet winner. If so, by construction, any alternative dominating the componentwise Condorcet winner must differ from it on at least two coordinates. In terms of interpretation, this means that among the majority coalition of voters who have this preference, some voters will vote against their preference on a specific bill in exchange of other(s) going against their preference on some other bill(s). This arrangement is referred to as logrolling or vote trading. The equivalence between logrolling and the non-existence of a Condorcet winner has been noted by many scholars including Bernholz (1973), Kadane (1972), Koehler (1975), Miller (1975,1977) and Schwartz (1975)¹.

In this note, we revisit the estimation of the likelihood of a Condorcet paradox in this logrolling setting. Since the domain of admissible preferences domain is restricted, the result cannot be obtained as a corollary of existing frequency results. Precisely, we compute, for three probabilistic models, the likelihood of a Condorcet paradox (or equivalently of logrolling) when two or three bills are under scrutiny. We compare the derived frequencies with the frequencies which are obtained in the unrestricted case.

¹Agreements are not binding. Building on early work by Riker and Brams (1973) and Ferejohn (1974), Casella and Palfrey (2015) challenge this equivalence. They look at a specific trading mechanism and exhibit examples where coalitional stability is not equivalent to the non-existence of a Condorcet winner.

2 The Setting

The set of voters/Committee/legislature is denoted by $N = \{1, \dots, n\}$ and the set of alternatives (Hypercube) by $X = \{0, 1\}^k$ where $k \geq 2$. Therefore the number m of alternatives is equal to 2^k . Here an alternative is a sequence of yes/no or accept/reject answers describing the decisions of the legislators/voters in the committee to a sequence of k bills/questions/pieces of legislation/issues on which they are deliberating.

We assume that the preferences of the members of the committee are restricted to belong to a subset of the set of linear orders over X . Precisely, we assume that for all $i \in N$, the preference P_i is a *separable* linear order. By separable, we mean that the following holds true. For all $a, b \in X$ and all $S \subset \{1, \dots, k\}$, if $aP_i b$ and $a^S = b^S$, then $(a^{N \setminus S}, c^S)P_i(b^{N \setminus S}, c^S)$ for all $c^S \in \{0, 1\}^S$. (Here d^T stands for the restriction of the vector $d \in X$ to the subset of coordinates T). We denote by LS_k the set of separable linear orders over X and by $\phi(k)$ their number².

A profile of preferences is a vector $\mathbf{P} = (P_1, P_2, \dots, P_n)$ in $(LS_k)^n$ where for all $i \in N$, P_i denotes the preference of voter i . The anonymous pattern attached to P is the vector $\mathbf{n}(\mathbf{P}) = (n_1(\mathbf{P}), n_2(\mathbf{P}), \dots, n_{\phi(k)}(\mathbf{P}))$ describing how many voters have each of the $\phi(k)$ conceivable separable preferences³. The anonymous pattern of a profile just keep track of the number of voters for each conceivable preference but forget their names. Of course:

$$\sum_{j=1}^{\phi(k)} n_j(\mathbf{P}) = n \text{ for all } \mathbf{P} \in (LS_k)^n$$

Given a profile \mathbf{P} in $(LS_k)^n$ and $x, y \in X$ with $x \neq y$, x majority dominates y for \mathbf{P} if:

$$\sum_{j=1}^{\phi(k)} n_j(\mathbf{P}, \mathbf{x}, \mathbf{y}) > \frac{n}{2}$$

where:

$$n_j(\mathbf{P}, \mathbf{u}, \mathbf{v}) = \begin{cases} n_j(\mathbf{P}, \mathbf{u}, \mathbf{v}) & \text{if } uP_j v \\ 0 & \text{if } vP_j u \end{cases} \text{ for all } u, v \text{ in } X, u \neq v \text{ and all } j = 1, \dots, \phi(k)$$

For every $\mathbf{P} \in (LS_k)^n$, *majority domination* defines a strict binary relation $MD(\mathbf{P})$ over X . If n is odd, then $MD(\mathbf{P})$ is complete. It is therefore a *tournament*. Note however, as first pointed out by Hollard and Le Breton (1996), that this tournament is separable in the following sense. For all $a, b \in X$ and all $S \subset \{1, \dots, k\}$, if $aMD(\mathbf{P})b$ and $a^S = b^S$,

²We note that $\phi(k+1) \geq 2(k+1)\phi(k)$.

³Each conceivable preference is identified by an index ranging from 1 to $\phi(k)$.

then $(a^{N \setminus S}, c^S)MD(P)(b^{N \setminus S}, c^S)$ for all $c^S \in \{0, 1\}^S$. We denote by TS_k the set of separable tournaments⁴ and by $\psi(k)$ the number of separable tournaments over X ⁵.

Given a profile \mathbf{P} in $(LS_k)^n$, $x \in X$ is a *Condorcet winner* for \mathbf{P} if for every $y \in X, y \neq x$, x majority dominates y . Condorcet winner does not always exist but note that for each of the k questions/issues, there is a Condorcet winner since we have only two possible answers (by separability, any voter can answer any question without needing to know what has been decided on the other questions). The vector x such that x^j is the Condorcet winner on the j^{th} component will be called the *componentwise Condorcet winner*. Note that if there is a Condorcet winner, it must be the componentwise Condorcet winner.

In what follows, we will make use of the following three probabilistic models over $(LS_k)^n$.

IC (Impartial Culture) will denote the model where each profile \mathbf{P} in $(LS_k)^n$ is drawn with equal probability. Since there are $\phi(k)^m$ such profiles, this means that each \mathbf{P} has a probability equal to $\frac{1}{\phi(k)^m}$.

IAC (Impartial Anonymous Culture) will denote the model where each anonymous pattern is drawn with equal probability. Since there are $\binom{n+\phi(k)-1}{\phi(k)-1}$ such patterns⁶, this means that each anonymous pattern has a probability equal to $\frac{1}{\binom{n+\phi(k)-1}{\phi(k)-1}}$. Note that strictly speaking IAC does not define a probability distribution over \mathbf{P} . In fact, any arbitrary assignment of probabilities to specific profiles will work in the computations that follow, as long as the overall likelihood for observing each of the anonymous patterns remains the same. For example, since for each anonymous pattern $\mathbf{n} = (n_1, n_2, \dots, n_{\phi(k)})$, there are $\frac{n!}{n_1!n_2!\dots n_{\phi(k)}!}$ attached profiles, we could assume for instance that they are all equally likely.

*ITC (Impartial Tournament Culture)*⁷ will denote the model where each separable tournament is drawn with equal probability. Since there are $\psi(k)$ such tournaments⁸, this means that each tournament has a probability equal to $\frac{1}{\psi(k)}$. As above, we can then split the probability mass attached to any tournament equally across the profiles compatible with that tournament.

⁴The set of separable tournaments is the superset of separable linear orders which is obtained when we delete the transitivity requirement while keeping the separability one

⁵Hollard and Le Breton (1996) have proved that every separable tournament can be obtained through majority aggregation of separable preferences if the number of voters is large enough.

⁶The number of ordered decompositions of the integer n into r integers is equal to $\binom{r+n-1}{r-1}$. In our case, r is the number of separable linear orderings.

⁷This wording is ours. To the best of our knowledge, there is no name for this probabilistic model. It has been used for instance by Bell (1981) and Fey (2008).

⁸To the best of our knowledge, nor the values of the function ψ have been tabulated, neither its asymptotic behavior has been studied.

3 The Case $k=2$

When $k = 2$, there are 8 separable linear orders over a total of 24 linear orders. Therefore, an anonymous pattern for n voters is a 8–dimensional vector $\mathbf{n} = (n_1, n_2, \dots, n_8)$ of integers:

n_1	n_2	n_3	n_4	n_5	n_6	n_7	n_8
(0, 0)	(0, 0)	(0, 1)	(0, 1)	(1, 0)	(1, 0)	(1, 1)	(1, 1)
(0, 1)	(1, 0)	(0, 0)	(1, 1)	(0, 0)	(1, 1)	(0, 1)	(1, 0)
(1, 0)	(0, 1)	(1, 1)	(0, 0)	(1, 1)	(0, 0)	(1, 0)	(0, 1)
(1, 1)	(1, 1)	(1, 0)	(1, 0)	(0, 1)	(0, 1)	(0, 0)	(0, 0)

such that: $n = n_1 + n_2 + \dots + n_8$

$(0, 0)$ is a Condorcet winner iff:

$$\begin{aligned}
 n_1 + n_2 + n_3 + n_5 &> \frac{n}{2} \\
 n_1 + n_2 + n_3 + n_4 &> \frac{n}{2} \\
 n_1 + n_2 + n_5 + n_6 &> \frac{n}{2}
 \end{aligned} \tag{1}$$

Probability that there exists a Condorcet winner for ITC

It is straightforward to show that there are 4 tournaments for which $(1, 1)$ is the componentwise Condorcet winner. This means that we have 16 separable tournaments (out of⁹ the $2^{4^2} = 64$ tournaments). Among those, only 8 have a Condorcet winner. So, we obtain the ITC probability of having a Condorcet winner is equal to 50%.

Probability that there exists a Condorcet winner for IC

Here IC means that for each voter, the 8 possible linear orders are drawn with probability $\frac{1}{8}$.

Therefore the distribution of the random vector \mathbf{n} is a multinomial distribution $\pi = (\frac{1}{8}, \frac{1}{8}, \dots, \frac{1}{8})$. From the multivariate central limit theorem, we deduce that when n is large $\frac{\mathbf{n} - n\pi}{\sqrt{n}}$ converges to the 8-dimensional gaussian $N(0, \Omega)$ where:

$$\Omega = \begin{pmatrix} \frac{7}{64} & \frac{-1}{64} & \cdot & \frac{-1}{64} \\ \frac{-1}{64} & \frac{7}{64} & \cdot & \frac{-1}{64} \\ \cdot & \cdot & \cdot & \cdot \\ \frac{-1}{64} & \frac{-1}{64} & \cdot & \frac{7}{64} \end{pmatrix}$$

Then, when n is large, the distribution of the 3–dimensional vector $(n_1 + n_2 + n_3 + n_5 - \frac{n}{2}, n_1 + n_2 + n_3 + n_4 - \frac{n}{2}, n_1 + n_2 + n_5 + n_6 - \frac{n}{2})$ is the 3–dimensional gaussian $Z \equiv N(0, \Delta)$ where:

⁹The number of tournaments over m vertices is equal to $2^{\binom{m}{2}}$.

$$\Delta = \begin{pmatrix} \frac{1}{4} & 0 & \frac{1}{8} \\ 0 & \frac{1}{4} & \frac{1}{8} \\ \frac{1}{8} & \frac{1}{8} & \frac{1}{4} \end{pmatrix}$$

We deduce that if n is large, the probability of the event described by inequalities (1) is the the probability that $Z \in \mathbb{R}_+^{\#}$. This probability is equal to:

$$\frac{1}{8} + \frac{1}{4\pi} \left(2 \arcsin \left(\frac{1}{2} \right) \right) \simeq 20.833\%$$

The probability of the existence of a Condorcet winner is 4 times this number. This number 83.332% is slightly larger than the number that we obtain when there are no separability restrictions (i.e. 24 linear orders instead of 8). In that case the probability is (Garman and Kamien (1968):

$$\frac{1}{2} \left[1 + \frac{6}{\pi} \arcsin \left(\frac{1}{3} \right) \right] \simeq 82.45\%$$

Probability that there exists a Condorcet winner for IAC

Here, the number of possible realizations of the integer valued random vector \mathbf{n} is:

$$\begin{aligned} & \sum_{i_1=0}^n \times \sum_{i_2=0}^{n-i_1} \times \sum_{i_3=0}^{n-i_1-i_2} \times \sum_{i_4=0}^{n-i_1-i_2-i_3} \\ & \times \sum_{i_5=0}^{n-i_1-i_2-i_3-i_4} \times \sum_{i_6=0}^{n-i_1-i_2-i_3-i_4-i_5} \times \sum_{i_7=0}^{n-i_1-i_2-i_3-i_4-i_5-i_6} \end{aligned}$$

The IAC model assumes that all these¹⁰ $\binom{n+7}{7} = \frac{1}{5040} (n+1)(n+2)(n+3)(n+4)(n+5)(n+6)(n+7)$ realizations are equally likely. Solving (1) amounts to counting the number of integer solutions to a system of linear inequalities. This can be done efficiently using the theory of Ehrhart's polynomials. After solving this combinatorial step, we obtain a number which has to be divided by the number above. The IAC probability of a Condorcet winner is 4 times this number.

We obtain

$$\frac{7(n+5)(n+3)}{8(n+2)(n+6)}$$

for n odd and

$$\frac{7n(2n+11)(n+4)(n+2)}{16(n+1)(n+3)(n+5)(n+7)}$$

¹⁰The number of ordered decompositions of the integer n into m integers is equal to $\binom{m+n-1}{m-1}$. Here m is the number of separable linear orderings.

for n even.

For n tending to infinity, we find that the IAC probability of a Condorcet winner is $\frac{7}{8} = 87.5\%$: this number is larger than the IAC number in the unrestricted case which is 83.84% (see Gehrlein, 2006).

4 The Case $k=3$

The number of alternatives is equal to $2^3 = 8$. Therefore, the total number of linear orders is $8! = 40320$. Let us enumerate the number of separable linear orders. Without loss of generality by symmetry, it is enough to count the number of separable orders with $(1, 1, 1)$ on top. Note that separability implies immediately that in such case, $(0, 0, 0)$ is the bottom alternative. We are left with counting the number of orderings of the 6 intermediate alternatives. Note first that there are 6 ways to order the vectors $(1, 0, 0)$, $(0, 1, 0)$ and $(0, 0, 1)$. Consider a separable linear order P such that $(1, 0, 0)P(0, 1, 0)P(0, 0, 1)$. By separability, we deduce that $(1, 1, 0)P(1, 0, 1)P(0, 1, 1)$. Indeed from separability $(1, 0, 0)P(0, 1, 0)$ implies that $(1, 0, 1)P(0, 1, 1)$ and $(0, 1, 0)P(0, 0, 1)$ implies $(1, 1, 0)P(1, 0, 1)$. Further, from separability again: $(1, 1, 0)P(1, 0, 0)$ and $(1, 0, 1)P(1, 0, 0)$. Therefore the only degree of freedom concerns the two necessarily adjacent alternatives $(0, 1, 1)$ and $(1, 0, 0)$. The two choices are compatible with separability. We conclude that there are $6 \times 2 = 12$ separable linear orders with $(1, 1, 1)$ on top. Since there are 8 possibilities to pick the top element, we conclude that there are $8 \times 12 = 96$ separable linear orders.

For subsequent use, it is useful to index these 96 linear orders. An anonymous pattern is now a 96-dimensional integer valued vector

Suppose that we want to look at those for which $(1, 1, 1)$ is a Condorcet winner. We will have 7 linear inequalities and the unique question is to write them in a regular way. One possibility for a nice coding would be the following.

Start with $(1, 1, 1)$ and then $(1, 1, 0)$, $(1, 0, 1)$, $(0, 1, 1)$, $(1, 0, 0)$, $(0, 1, 0)$, $(0, 0, 1)$, $(0, 0, 0)$ i.e. rank the blocks from low index to high index according to the number of 1 in the top vector and in case of ties, order lexicographically from the first coordinate to the third.

In the first block, rank the six orderings of the three vectors $(1, 0, 0)$, $(0, 1, 0)$ and $(0, 0, 1)$ lexicographically with respect to the order of components from first to sixth according and for each ordering rank the ordering where the top vector with a single 1 dominates a vector with two 1 just before the ordering where the top vector with a single 1 is dominated by a vector with two 1.

Doing so, we obtain the following indexation of any anonymous pattern $\mathbf{n} = (n_1, n_2, \dots, n_{96})$:

n_1	n_2	n_3	n_4	n_5	n_6	n_7	n_8	n_9	n_{10}	n_{11}	n_{12}
(1, 1, 1)	(1, 1, 1)	(1, 1, 1)	(1, 1, 1)	(1, 1, 1)	(1, 1, 1)	(1, 1, 1)	(1, 1, 1)	(1, 1, 1)	(1, 1, 1)	(1, 1, 1)	(1, 1, 1)
(1, 1, 0)	(1, 1, 0)	(1, 0, 1)	(1, 0, 1)	(1, 1, 0)	(1, 1, 0)	(0, 1, 1)	(0, 1, 1)	(1, 0, 1)	(1, 0, 1)	(0, 1, 1)	(0, 1, 1)
(1, 0, 1)	(1, 0, 1)	(1, 1, 0)	(1, 1, 0)	(0, 1, 1)	(0, 1, 1)	(1, 1, 0)	(1, 1, 0)	(0, 1, 1)	(0, 1, 1)	(1, 0, 1)	(1, 0, 1)
(1, 0, 0)	(0, 1, 1)	(1, 0, 0)	(0, 1, 1)	(0, 1, 0)	(1, 0, 1)	(0, 1, 0)	(1, 0, 1)	(0, 0, 1)	(1, 1, 0)	(0, 0, 1)	(1, 1, 0)
(0, 1, 1)	(1, 0, 0)	(0, 1, 1)	(1, 0, 0)	(1, 0, 1)	(0, 1, 0)	(1, 0, 1)	(0, 1, 0)	(1, 1, 0)	(0, 0, 1)	(1, 1, 0)	(0, 0, 1)
(0, 1, 0)	(0, 1, 0)	(0, 0, 1)	(0, 0, 1)	(1, 0, 0)	(1, 0, 0)	(0, 0, 1)	(0, 0, 1)	(1, 0, 0)	(1, 0, 0)	(0, 1, 0)	(0, 1, 0)
(0, 0, 1)	(0, 0, 1)	(0, 1, 0)	(0, 1, 0)	(0, 0, 1)	(0, 0, 1)	(1, 0, 0)	(1, 0, 0)	(0, 1, 0)	(0, 1, 0)	(1, 0, 0)	(1, 0, 0)
(0, 0, 0)	(0, 0, 0)	(0, 0, 0)	(0, 0, 0)	(0, 0, 0)	(0, 0, 0)	(0, 0, 0)	(0, 0, 0)	(0, 0, 0)	(0, 0, 0)	(0, 0, 0)	(0, 0, 0)

(1, 1, 1) on top

n_{13}	n_{14}	n_{15}	n_{16}	n_{17}	n_{18}	n_{19}	n_{20}	n_{21}	n_{22}	n_{23}	n_{24}
(1, 1, 0)	(1, 1, 0)	(1, 1, 0)	(1, 1, 0)	(1, 1, 0)	(1, 1, 0)	(1, 1, 0)	(1, 1, 0)	(1, 1, 0)	(1, 1, 0)	(1, 1, 0)	(1, 1, 0)
(1, 1, 1)	(1, 1, 1)	(1, 0, 0)	(1, 0, 0)	(1, 1, 1)	(1, 1, 1)	(0, 1, 0)	(0, 1, 0)	(1, 0, 0)	(1, 0, 0)	(0, 1, 0)	(0, 1, 0)
(1, 0, 0)	(1, 0, 0)	(1, 1, 1)	(1, 1, 1)	(0, 1, 0)	(0, 1, 0)	(1, 1, 1)	(1, 1, 1)	(0, 1, 0)	(0, 1, 0)	(1, 0, 0)	(1, 0, 0)
(1, 0, 1)	(0, 1, 0)	(1, 0, 1)	(0, 1, 0)	(0, 1, 1)	(1, 0, 0)	(0, 1, 1)	(1, 0, 0)	(0, 0, 0)	(1, 1, 1)	(0, 0, 0)	(1, 1, 1)
(0, 1, 0)	(1, 0, 1)	(0, 1, 0)	(1, 0, 1)	(1, 0, 0)	(0, 1, 1)	(1, 0, 0)	(0, 1, 1)	(1, 1, 1)	(0, 0, 0)	(1, 1, 1)	(0, 0, 0)
(0, 1, 1)	(0, 1, 1)	(0, 0, 0)	(0, 0, 0)	(1, 0, 1)	(1, 0, 1)	(0, 0, 0)	(0, 0, 0)	(1, 0, 1)	(1, 0, 1)	(0, 1, 1)	(0, 1, 1)
(0, 0, 0)	(0, 0, 0)	(0, 1, 1)	(0, 1, 1)	(0, 0, 0)	(0, 0, 0)	(1, 0, 1)	(1, 0, 1)	(0, 1, 1)	(0, 1, 1)	(1, 0, 1)	(1, 0, 1)
(0, 0, 1)	(0, 0, 1)	(0, 0, 1)	(0, 0, 1)	(0, 0, 1)	(0, 0, 1)	(0, 0, 1)	(0, 0, 1)	(0, 0, 1)	(0, 0, 1)	(0, 0, 1)	(0, 0, 1)

(1, 1, 0) on top

n_{25}	n_{26}	n_{27}	n_{28}	n_{29}	n_{30}	n_{31}	n_{32}	n_{33}	n_{34}	n_{35}	n_{36}
(1, 0, 1)	(1, 0, 1)	(1, 0, 1)	(1, 0, 1)	(1, 0, 1)	(1, 0, 1)	(1, 0, 1)	(1, 0, 1)	(1, 0, 1)	(1, 0, 1)	(1, 0, 1)	(1, 0, 1)
(1, 0, 0)	(1, 0, 0)	(1, 1, 1)	(1, 1, 1)	(1, 0, 0)	(1, 0, 0)	(0, 0, 1)	(0, 0, 1)	(1, 1, 1)	(1, 1, 1)	(0, 0, 1)	(0, 0, 1)
(1, 1, 1)	(1, 1, 1)	(1, 0, 0)	(1, 0, 0)	(0, 0, 1)	(0, 0, 1)	(1, 0, 0)	(1, 0, 0)	(0, 0, 1)	(0, 0, 1)	(1, 1, 1)	(1, 1, 1)
(1, 1, 0)	(0, 0, 1)	(1, 1, 0)	(0, 0, 1)	(0, 0, 0)	(1, 1, 1)	(0, 0, 0)	(1, 1, 1)	(0, 1, 1)	(1, 0, 0)	(0, 1, 1)	(1, 0, 0)
(0, 0, 1)	(1, 1, 0)	(0, 0, 1)	(1, 1, 0)	(1, 1, 1)	(0, 0, 0)	(1, 1, 1)	(0, 0, 0)	(1, 0, 0)	(0, 1, 1)	(1, 0, 0)	(0, 1, 1)
(0, 0, 0)	(0, 0, 0)	(0, 1, 1)	(0, 1, 1)	(1, 1, 0)	(1, 1, 0)	(0, 1, 1)	(0, 1, 1)	(1, 1, 0)	(1, 1, 0)	(0, 0, 0)	(0, 0, 0)
(0, 1, 1)	(0, 1, 1)	(0, 0, 0)	(0, 0, 0)	(0, 1, 1)	(0, 1, 1)	(1, 1, 0)	(1, 1, 0)	(0, 0, 0)	(0, 0, 0)	(1, 1, 0)	(1, 1, 0)
(0, 1, 0)	(0, 1, 0)	(0, 1, 0)	(0, 1, 0)	(0, 1, 0)	(0, 1, 0)	(0, 1, 0)	(0, 1, 0)	(0, 1, 0)	(0, 1, 0)	(0, 1, 0)	(0, 1, 0)

(1, 0, 1) on top

n_{73}	n_{74}	n_{75}	n_{76}	n_{77}	n_{78}	n_{79}	n_{80}	n_{81}	n_{82}	n_{83}	n_{84}
(0, 0, 1)	(0, 0, 1)	(0, 0, 1)	(0, 0, 1)	(0, 0, 1)	(0, 0, 1)	(0, 0, 1)	(0, 0, 1)	(0, 0, 1)	(0, 0, 1)	(0, 0, 1)	(0, 0, 1)
(0, 0, 0)	(0, 0, 0)	(0, 1, 1)	(0, 1, 1)	(0, 0, 0)	(0, 0, 0)	(1, 0, 1)	(1, 0, 1)	(0, 1, 1)	(0, 1, 1)	(1, 0, 1)	(1, 0, 1)
(0, 1, 1)	(0, 1, 1)	(0, 0, 0)	(0, 0, 0)	(1, 0, 1)	(1, 0, 1)	(0, 0, 0)	(0, 0, 0)	(1, 0, 1)	(1, 0, 1)	(0, 1, 1)	(0, 1, 1)
(0, 1, 0)	(1, 0, 1)	(0, 1, 0)	(1, 0, 1)	(1, 0, 0)	(0, 1, 1)	(1, 0, 0)	(0, 1, 1)	(1, 1, 1)	(0, 0, 0)	(1, 1, 1)	(0, 0, 0)
(1, 0, 1)	(0, 1, 0)	(1, 0, 1)	(0, 1, 0)	(0, 1, 1)	(1, 0, 0)	(0, 1, 1)	(1, 0, 0)	(0, 0, 0)	(1, 1, 1)	(0, 0, 0)	(1, 1, 1)
(1, 0, 0)	(1, 0, 0)	(1, 1, 1)	(1, 1, 1)	(0, 1, 0)	(0, 1, 0)	(1, 1, 1)	(1, 1, 1)	(0, 1, 0)	(0, 1, 0)	(1, 0, 0)	(1, 0, 0)
(1, 1, 1)	(1, 1, 1)	(1, 0, 0)	(1, 0, 0)	(1, 1, 1)	(1, 1, 1)	(0, 1, 0)	(0, 1, 0)	(1, 0, 0)	(1, 0, 0)	(0, 1, 0)	(0, 1, 0)
(1, 1, 0)	(1, 1, 0)	(1, 1, 0)	(1, 1, 0)	(1, 1, 0)	(1, 1, 0)	(1, 1, 0)	(1, 1, 0)	(1, 1, 0)	(1, 1, 0)	(1, 1, 0)	(1, 1, 0)

(0, 0, 1) on top

n_{85}	n_{86}	n_{87}	n_{88}	n_{89}	n_{90}	n_{91}	n_{92}	n_{93}	n_{94}	n_{95}	n_{96}
(0, 0, 0)	(0, 0, 0)	(0, 0, 0)	(0, 0, 0)	(0, 0, 0)	(0, 0, 0)	(0, 0, 0)	(0, 0, 0)	(0, 0, 0)	(0, 0, 0)	(0, 0, 0)	(0, 0, 0)
(0, 0, 1)	(0, 0, 1)	(0, 1, 0)	(0, 1, 0)	(0, 0, 1)	(0, 0, 1)	(1, 0, 0)	(1, 0, 0)	(0, 1, 0)	(0, 1, 0)	(1, 0, 0)	(1, 0, 0)
(0, 1, 0)	(0, 1, 0)	(0, 0, 1)	(0, 0, 1)	(1, 0, 0)	(1, 0, 0)	(0, 0, 1)	(0, 0, 1)	(1, 0, 0)	(1, 0, 0)	(0, 1, 0)	(0, 1, 0)
(0, 1, 1)	(1, 0, 0)	(0, 1, 1)	(1, 0, 0)	(1, 0, 1)	(0, 1, 0)	(1, 0, 1)	(0, 1, 0)	(1, 1, 0)	(0, 0, 1)	(1, 1, 0)	(0, 0, 1)
(1, 0, 0)	(0, 1, 1)	(1, 0, 0)	(0, 1, 1)	(0, 1, 0)	(1, 0, 1)	(0, 1, 0)	(1, 0, 1)	(0, 0, 1)	(1, 1, 0)	(0, 0, 1)	(1, 1, 0)
(1, 0, 1)	(1, 0, 1)	(1, 1, 0)	(1, 1, 0)	(0, 1, 1)	(0, 1, 1)	(1, 1, 0)	(1, 1, 0)	(0, 1, 1)	(0, 1, 1)	(1, 0, 1)	(1, 0, 1)
(1, 1, 0)	(1, 1, 0)	(1, 0, 1)	(1, 0, 1)	(1, 1, 0)	(1, 1, 0)	(0, 1, 1)	(0, 1, 1)	(1, 0, 1)	(1, 0, 1)	(0, 1, 1)	(0, 1, 1)
(1, 1, 1)	(1, 1, 1)	(1, 1, 1)	(1, 1, 1)	(1, 1, 1)	(1, 1, 1)	(1, 1, 1)	(1, 1, 1)	(1, 1, 1)	(1, 1, 1)	(1, 1, 1)	(1, 1, 1)

(0, 0, 0) on top

(1, 1, 1) is a Condorcet winner iff the following seven inequalities hold true

$$\begin{aligned}
& n_1 + n_2 + n_3 + n_4 + n_5 + n_6 + n_7 + n_8 + n_9 + n_{10} + n_{11} + n_{12} \\
& + n_{25} + n_{26} + n_{27} + n_{28} + n_{29} + n_{30} + n_{31} + n_{32} + n_{33} + n_{34} + n_{35} + n_{36} \\
& + n_{37} + n_{38} + n_{39} + n_{40} + n_{41} + n_{42} + n_{43} + n_{44} + n_{45} + n_{46} + n_{47} + n_{48} \\
& + n_{73} + n_{74} + n_{75} + n_{76} + n_{77} + n_{78} + n_{79} + n_{80} + n_{81} + n_{82} + n_{83} + n_{84} \\
& > \frac{n}{2};
\end{aligned}$$

$$\begin{aligned}
& n_1 + n_2 + n_3 + n_4 + n_5 + n_6 + n_7 + n_8 + n_9 + n_{10} + n_{11} + n_{12} \\
& + n_{13} + n_{14} + n_{15} + n_{16} + n_{17} + n_{18} + n_{19} + n_{20} + n_{21} + n_{22} + n_{23} + n_{24} \\
& + n_{37} + n_{38} + n_{39} + n_{40} + n_{41} + n_{42} + n_{43} + n_{44} + n_{45} + n_{46} + n_{47} + n_{48}
\end{aligned}$$

$$\begin{aligned}
& +n_{61} + n_{62} + n_{63} + n_{64} + n_{65} + n_{66} + n_{67} + n_{68} + n_{69} + n_{70} + n_{71} + n_{72} \\
& > \frac{n}{2};
\end{aligned}$$

$$\begin{aligned}
& n_1 + n_2 + n_3 + n_4 + n_5 + n_6 + n_7 + n_8 + n_9 + n_{10} + n_{11} + n_{12} \\
& +n_{13} + n_{14} + n_{15} + n_{16} + n_{17} + n_{18} + n_{19} + n_{20} + n_{21} + n_{22} + n_{23} + n_{24} \\
& +n_{25} + n_{26} + n_{27} + n_{28} + n_{29} + n_{30} + n_{31} + n_{32} + n_{33} + n_{34} + n_{35} + n_{36} \\
& +n_{49} + n_{50} + n_{51} + n_{52} + n_{53} + n_{54} + n_{55} + n_{56} + n_{57} + n_{58} + n_{59} + n_{60} \\
& > \frac{n}{2};
\end{aligned}$$

$$\begin{aligned}
& n_1 + n_2 + n_3 + n_4 + n_5 + n_6 + n_7 + n_8 + n_9 + n_{10} + n_{11} + n_{12} \\
& \quad +n_{13} + n_{14} + n_{17} + n_{18} + n_{19} + n_{20} \\
& \quad +n_{27} + n_{28} + n_{33} + n_{34} + n_{35} + n_{36} \\
& +n_{37} + n_{38} + n_{39} + n_{40} + n_{41} + n_{42} + n_{43} + n_{44} + n_{45} + n_{46} + n_{47} + n_{48} \\
& \quad +n_{61} + n_{62} + n_{65} + n_{66} + n_{67} + n_{68} \\
& \quad n_{75} + n_{76} + n_{81} + n_{82} + n_{83} + n_{84} \\
& > \frac{n}{2};
\end{aligned}$$

$$\begin{aligned}
& n_1 + n_2 + n_3 + n_4 + n_5 + n_6 + n_7 + n_8 + n_9 + n_{10} + n_{11} + n_{12} \\
& \quad +n_{13} + n_{14} + n_{15} + n_{16} + n_{17} + n_{18} \\
& +n_{25} + n_{26} + n_{27} + n_{28} + n_{29} + n_{30} + n_{31} + n_{32} + n_{33} + n_{34} + n_{35} + n_{36} \\
& \quad +n_{49} + n_{50} + n_{51} + n_{52} + n_{53} + n_{54} \\
& \quad +n_{79} + n_{80} + n_{81} + n_{82} + n_{83} + n_{84} \\
& > \frac{n}{2};
\end{aligned}$$

(2)

$$\begin{aligned}
& n_1 + n_2 + n_3 + n_4 + n_5 + n_6 + n_7 + n_8 + n_9 + n_{10} + n_{11} + n_{12} \\
& \quad +n_{13} + n_{14} + n_{15} + n_{16} + n_{17} + n_{18} + n_{19} + n_{20} + n_{23} + n_{24} \\
& \quad +n_{25} + n_{26} + n_{27} + n_{28} + n_{33} + n_{34} \\
& \quad +n_{41} + n_{42} + n_{43} + n_{44} + n_{47} + n_{48} \\
& \quad +n_{49} + n_{50} + n_{51} + n_{52} + n_{57} + n_{58}
\end{aligned}$$

$$\begin{aligned}
& +n_{65} + n_{66} + n_{67} + n_{68} + n_{71} + n_{72} \\
& > \frac{n}{2};
\end{aligned}$$

and

$$\begin{aligned}
& n_1 + n_2 + n_3 + n_4 + n_5 + n_6 + n_7 + n_8 + n_9 + n_{10} + n_{11} + n_{12} \\
& +n_{13} + n_{14} + n_{15} + n_{16} + n_{17} + n_{18} + n_{19} + n_{20} + n_{23} + n_{24} \\
& +n_{25} + n_{26} + n_{27} + n_{28} + n_{30} + n_{32} + n_{33} + n_{34} + n_{35} + n_{36} \\
& +n_{38} + n_{40} + n_{41} + n_{42} + n_{43} + n_{44} + n_{45} + n_{46} + n_{47} + n_{48} \\
& +n_{49} + n_{51} \\
& +n_{65} + n_{67} \\
& +n_{81} + n_{83} \\
& > \frac{n}{2}.
\end{aligned}$$

Probability that there exists a Condorcet winner for ITC

It is easy to show that the number of separable tournaments with $(1, 1, 1)$ as a componentwise Condorcet winner is equal to $4^5 = 1024$. This implies that we have $8 \times 1024 = 8192$ separable tournaments (out of the $2^{82} = 268\,435\,456$ tournaments!). Further among those, only $8 \times 4^3 = 512$ have a Condorcet winner. This implies that the ITC probability that there exists a Condorcet winner is equal to $\frac{512}{8192} = \frac{1}{16} = 6.25\%$.

Probability that there exists a Condorcet winner for IC

Here IC means that for each voter, the 96 possible linear orders are drawn with probability $\frac{1}{96}$.

Therefore, the distribution of the random vector \mathbf{n} is a multinomial distribution $\pi = (\frac{1}{96}, \frac{1}{96}, \dots, \frac{1}{96})$. From the multivariate central limit theorem, we deduce that when n is large $\frac{\mathbf{n} - n\pi}{\sqrt{n}}$ converges to the 96-dimensional gaussian $N(0, \Omega)$ where:

$$\Omega = \begin{pmatrix} \frac{95}{9216} & \frac{-1}{9216} & \cdot & \frac{-1}{9216} \\ \frac{-1}{9216} & \frac{95}{9216} & \cdot & \frac{-1}{9216} \\ \cdot & \cdot & \cdot & \cdot \\ \frac{-1}{9216} & \frac{-1}{9216} & \cdot & \frac{95}{9216} \end{pmatrix}$$

Then, when n is large, the distribution of the 7-dimensional vector

$$\left(\frac{N^1 - \frac{n}{2}}{\sqrt{n}}, \frac{N^2 - \frac{n}{2}}{\sqrt{n}}, \frac{N^3 - \frac{n}{2}}{\sqrt{n}}, \frac{N^4 - \frac{n}{2}}{\sqrt{n}}, \frac{N^5 - \frac{n}{92}}{\sqrt{n}}, \frac{N^6 - \frac{n}{2}}{\sqrt{n}}, \frac{N^7 - \frac{n}{2}}{\sqrt{n}} \right)$$

where:

$$\begin{aligned}
N^1 = & n_1 + n_2 + n_3 + n_4 + n_5 + n_6 + n_7 + n_8 + n_9 + n_{10} + n_{11} + n_{12} + \\
& n_{25} + n_{26} + n_{27} + n_{28} + n_{29} + n_{30} + n_{31} + n_{32} + n_{33} + n_{34} + n_{35} + n_{36} + \\
& n_{37} + n_{38} + n_{39} + n_{40} + n_{41} + n_{42} + n_{43} + n_{44} + n_{45} + n_{46} + n_{47} + n_{48} + \\
& n_{73} + n_{74} + n_{75} + n_{76} + n_{77} + n_{78} + n_{79} + n_{80} + n_{81} + n_{82} + n_{83} + n_{84}
\end{aligned}$$

$$\begin{aligned}
N^2 = & n_1 + n_2 + n_3 + n_4 + n_5 + n_6 + n_7 + n_8 + n_9 + n_{10} + n_{11} + n_{12} + \\
& n_{13} + n_{14} + n_{15} + n_{16} + n_{17} + n_{18} + n_{19} + n_{20} + n_{21} + n_{22} + n_{23} + n_{24} + \\
& n_{37} + n_{38} + n_{39} + n_{40} + n_{41} + n_{42} + n_{43} + n_{44} + n_{45} + n_{46} + n_{47} + n_{48} + \\
& n_{61} + n_{62} + n_{63} + n_{64} + n_{65} + n_{66} + n_{67} + n_{68} + n_{69} + n_{70} + n_{71} + n_{72}
\end{aligned}$$

$$\begin{aligned}
N^3 = & n_1 + n_2 + n_3 + n_4 + n_5 + n_6 + n_7 + n_8 + n_9 + n_{10} + n_{11} + n_{12} + \\
& n_{13} + n_{14} + n_{15} + n_{16} + n_{17} + n_{18} + n_{19} + n_{20} + n_{21} + n_{22} + n_{23} + n_{24} + \\
& n_{25} + n_{26} + n_{27} + n_{28} + n_{29} + n_{30} + n_{31} + n_{32} + n_{33} + n_{34} + n_{35} + n_{36} + \\
& n_{49} + n_{50} + n_{51} + n_{52} + n_{53} + n_{54} + n_{56} + n_{56} + n_{57} + n_{58} + n_{59} + n_{60}
\end{aligned}$$

$$\begin{aligned}
N^4 = & n_1 + n_2 + n_3 + n_4 + n_5 + n_6 + n_7 + n_8 + n_9 + n_{10} + n_{11} + n_{12} + \\
& n_{13} + n_{14} + n_{17} + n_{18} + n_{19} + n_{20} + \\
& n_{27} + n_{28} + n_{33} + n_{34} + n_{35} + n_{36} + \\
& n_{37} + n_{38} + n_{39} + n_{40} + n_{41} + n_{42} + n_{43} + n_{44} + n_{45} + n_{46} + n_{47} + n_{48} + \\
& n_{61} + n_{62} + n_{65} + n_{66} + n_{67} + n_{68} + \\
& n_{75} + n_{76} + n_{81} + n_{82} + n_{83} + n_{84}
\end{aligned}$$

$$\begin{aligned}
N^5 = & n_1 + n_2 + n_3 + n_4 + n_5 + n_6 + n_7 + n_8 + n_9 + n_{10} + n_{11} + n_{12} + \\
& n_{13} + n_{14} + n_{15} + n_{16} + n_{17} + n_{18} + \\
& n_{25} + n_{26} + n_{27} + n_{28} + n_{29} + n_{30} + n_{31} + n_{32} + n_{33} + n_{34} + n_{35} + n_{36} + \\
& n_{43} + n_{44} + n_{45} + n_{46} + n_{47} + n_{48} + \\
& n_{49} + n_{50} + n_{51} + n_{52} + n_{53} + n_{54} + \\
& n_{79} + n_{80} + n_{81} + n_{82} + n_{83} + n_{84}
\end{aligned}$$

$$\begin{aligned}
N^6 = & n_1 + n_2 + n_3 + n_4 + n_5 + n_6 + n_7 + n_8 + n_9 + n_{10} + n_{11} + n_{12} + \\
& n_{13} + n_{14} + n_{15} + n_{16} + n_{17} + n_{18} + n_{19} + n_{20} + n_{21} + n_{22} + n_{23} + n_{24} + \\
& n_{25} + n_{26} + n_{27} + n_{28} + n_{33} + n_{34} + \\
& n_{41} + n_{42} + n_{43} + n_{44} + n_{47} + n_{48} + \\
& n_{49} + n_{50} + n_{51} + n_{52} + n_{57} + n_{58} + \\
& n_{65} + n_{66} + n_{67} + n_{68} + n_{71} + n_{72}
\end{aligned}$$

$$\begin{aligned}
N^7 = & n_1 + n_2 + n_3 + n_4 + n_5 + n_6 + n_7 + n_8 + n_9 + n_{10} + n_{11} + n_{12} + \\
& n_{13} + n_{14} + n_{15} + n_{16} + n_{17} + n_{18} + n_{19} + n_{20} + n_{22} + n_{24} + \\
& n_{25} + n_{26} + n_{27} + n_{28} + n_{30} + n_{32} + n_{33} + n_{34} + n_{35} + n_{36} + \\
& n_{38} + n_{40} + n_{41} + n_{42} + n_{43} + n_{44} + n_{45} + n_{46} + n_{47} + n_{48} + \\
& n_{49} + n_{51} + \\
& n_{65} + n_{67} + \\
& n_{81} + n_{83}
\end{aligned}$$

is approximatively the 7-dimensional gaussian $Z \equiv N(0, \Delta)$ where:

$$\Delta = \begin{pmatrix} \frac{1}{4} & 0 & 0 & \frac{1}{8} & \frac{1}{8} & 0 & \frac{5}{48} \\ 0 & \frac{1}{4} & 0 & \frac{1}{8} & 0 & \frac{1}{8} & \frac{5}{48} \\ 0 & 0 & \frac{1}{4} & 0 & \frac{1}{8} & \frac{1}{8} & \frac{5}{48} \\ \frac{1}{8} & \frac{1}{8} & 0 & \frac{1}{4} & \frac{8}{48} & \frac{8}{48} & \frac{7}{48} \\ \frac{1}{8} & 0 & 0 & \frac{1}{4} & \frac{48}{48} & \frac{48}{48} & \frac{7}{48} \\ \frac{1}{8} & \frac{1}{8} & \frac{1}{8} & \frac{48}{48} & \frac{1}{4} & \frac{48}{48} & \frac{7}{48} \\ 0 & \frac{1}{8} & \frac{1}{8} & \frac{48}{48} & \frac{48}{48} & \frac{1}{4} & \frac{7}{48} \\ \frac{5}{48} & \frac{5}{48} & \frac{5}{48} & \frac{7}{48} & \frac{7}{48} & \frac{7}{48} & \frac{1}{4} \end{pmatrix}$$

According to Niemi and Weisberg (1968), the probability of having a Condorcet winner when there are 8 alternatives is equal to 58.49%. Here, by using the R program the codes of which are reproduced together with one simulation output in appendix 1, we obtain with 10^6 simulations, a bootstrap of 2000 and 1000 draws, an average empirical probability equal to 0.07382304, a variance equal to $3.20618927327327 \times 10^{-7}$ and a 95% empirical confidence interval equal to [0.072654875, 0.074935125].

We deduce that the IC probability of having a Condorcet winner is here around $0.07382304 \times 8 = 0.59058$ with a 95% chance of being in the interval [0.58124, 0.59948]. It seems that the

probability is higher than in the unrestricted case but given the random estimate, we cannot claim that it holds true with probability 1.

Another approach could consist in the computation of the multiple integral expressing the orthant probability. We just provide a sketch in appendix 2.

Probability that there exists a Condorcet winner for IAC

Here, all $\binom{n+95}{95}$ possible realizations of the integer valued random vector P_n are assumed to be equally likely. Solving (2) amounts to count integer solutions to a system of linear inequalities. This can be done efficiently using the theory of Ehrhart’s polynomials. After solving this combinatorial step, we obtain a number which has to be divided by the number above. The IAC probability of a Condorcet winner is 8 times this number.

When n tends to infinity, the probability 96–dimensional vector describing the proportions (after division by n) of voters in each of the 96 groups is the uniform law on the 96–dimensional unitary simplex. Further, the 7 inequalities describing the constraints on the proportions $p_i = \frac{n_i}{n}$ (instead of the integers n_i) for $(1, 1, 1)$ to be a Condorcet winner are linear cuts. Of course, $\mathbf{p} = (p_1, p_2, \dots, p_{96})$ is a vector in the 96–dimensional unitary simplex S_{96} i.e. \mathbf{p} is such that:

$$p_i \geq 0 \text{ for all } i = 1, 2, \dots, 96 \quad \text{and} \quad \sum_{i=1}^{96} p_i = 1.$$

The probability that $(1, 1, 1)$ is a Condorcet winner with an infinitely large population is the volume of a convex polytope, denoted H , in S_{96} described by the following 7 linear inequalities

$$\begin{aligned} & p_1 + p_2 + p_3 + p_4 + p_5 + p_6 + p_7 + p_8 + p_9 + p_{10} + p_{11} + p_{12} \\ & + p_{25} + p_{26} + p_{27} + p_{28} + p_{29} + p_{30} + p_{31} + p_{32} + p_{33} + p_{34} + p_{35} + p_{36} \\ & + p_{37} + p_{38} + p_{39} + p_{40} + p_{41} + p_{42} + p_{43} + p_{44} + p_{45} + p_{46} + p_{47} + p_{48} \\ & + p_{73} + p_{74} + p_{75} + p_{76} + p_{77} + p_{78} + p_{79} + p_{80} + p_{81} + p_{82} + p_{83} + p_{84} \\ & \geq \frac{1}{2}; \end{aligned}$$

$$\begin{aligned} & p_1 + p_2 + p_3 + p_4 + p_5 + p_6 + p_7 + p_8 + p_9 + p_{10} + p_{11} + p_{12} \\ & + p_{13} + p_{14} + p_{15} + p_{16} + p_{17} + p_{18} + p_{19} + p_{20} + p_{21} + p_{22} + p_{23} + p_{24} \\ & + p_{37} + p_{38} + p_{39} + p_{40} + p_{41} + p_{42} + p_{43} + p_{44} + p_{45} + p_{46} + p_{47} + p_{48} \\ & + p_{61} + p_{62} + p_{63} + p_{64} + p_{65} + p_{66} + p_{67} + p_{68} + p_{69} + p_{70} + p_{71} + p_{72} \\ & \geq \frac{1}{2}; \end{aligned}$$

$$\begin{aligned}
& p_1 + p_2 + p_3 + p_4 + p_5 + p_6 + p_7 + p_8 + p_9 + p_{10} + p_{11} + p_{12} \\
& + p_{13} + p_{14} + p_{15} + p_{16} + p_{17} + p_{18} + p_{19} + p_{20} + p_{21} + p_{22} + p_{23} + p_{24} \\
& + p_{25} + p_{26} + p_{27} + p_{28} + p_{29} + p_{30} + p_{31} + p_{32} + p_{33} + p_{34} + p_{35} + p_{36} \\
& + p_{49} + p_{50} + p_{51} + p_{52} + p_{53} + p_{54} + p_{55} + p_{56} + p_{57} + p_{58} + p_{59} + p_{60} \\
& \geq \frac{1}{2};
\end{aligned}$$

$$\begin{aligned}
& p_1 + p_2 + p_3 + p_4 + p_5 + p_6 + p_7 + p_8 + p_9 + p_{10} + p_{11} + p_{12} \\
& \quad + p_{13} + p_{14} + p_{17} + p_{18} + p_{19} + p_{20} \\
& \quad \quad p_{27} + p_{28} + p_{33} + p_{34} + p_{35} + p_{36} \\
& + p_{37} + p_{38} + p_{39} + p_{40} + p_{41} + p_{42} + p_{43} + p_{44} + p_{45} + p_{46} + p_{47} + p_{48} \\
& \quad + p_{61} + p_{62} + p_{65} + p_{66} + p_{67} + p_{68} \\
& \quad \quad p_{75} + p_{76} + p_{81} + p_{82} + p_{83} + p_{84} \\
& \geq \frac{1}{2};
\end{aligned}$$

$$\begin{aligned}
& p_1 + p_2 + p_3 + p_4 + p_5 + p_6 + p_7 + p_8 + p_9 + p_{10} + p_{11} + p_{12} \\
& \quad + p_{13} + p_{14} + p_{15} + p_{16} + p_{17} + p_{18} \\
& + p_{25} + p_{26} + p_{27} + p_{28} + p_{29} + p_{30} + p_{31} + p_{32} + p_{33} + p_{34} + p_{35} + p_{36} \\
& \quad \quad p_{43} + p_{44} + p_{45} + p_{46} + p_{47} + p_{48} \\
& \quad + p_{49} + p_{50} + p_{51} + p_{52} + p_{53} + p_{54} \\
& \quad + p_{79} + p_{80} + p_{81} + p_{82} + p_{83} + p_{84} \\
& \geq \frac{1}{2};
\end{aligned}$$

$$\begin{aligned}
& p_1 + p_2 + p_3 + p_4 + p_5 + p_6 + p_7 + p_8 + p_9 + p_{10} + p_{11} + p_{12} \\
& + p_{13} + p_{14} + p_{15} + p_{16} + p_{17} + p_{18} + p_{19} + p_{20} + p_{21} + p_{22} + p_{23} + p_{24} \\
& \quad + p_{25} + p_{26} + p_{27} + p_{28} + p_{33} + p_{34} \\
& \quad + p_{41} + p_{42} + p_{43} + p_{44} + p_{47} + p_{48} \\
& \quad + p_{49} + p_{50} + p_{51} + p_{52} + p_{57} + p_{58} \\
& \quad + p_{65} + p_{66} + p_{67} + p_{68} + p_{71} + p_{72}
\end{aligned}$$

$$\geq \frac{1}{2};$$

and

$$\begin{aligned} & p_1 + p_2 + p_3 + p_4 + p_5 + p_6 + p_7 + p_8 + p_9 + p_{10} + p_{11} + p_{12} \\ & + p_{13} + p_{14} + p_{15} + p_{16} + p_{17} + p_{18} + p_{19} + p_{20} + p_{22} + p_{24} \\ & + p_{25} + p_{26} + p_{27} + p_{28} + p_{30} + p_{32} + p_{33} + p_{34} + p_{35} + p_{36} \\ & + p_{38} + p_{40} + p_{41} + p_{42} + p_{43} + p_{44} + p_{45} + p_{46} + p_{47} + p_{48} \\ & + p_{49} + p_{51} \\ & + p_{65} + p_{67} \\ & + p_{81} + p_{83} \\ & \geq \frac{1}{2}. \end{aligned}$$

The codes of the R program to compute the volume of an arbitrary polytope are described in appendix 3. Having to deal with S_{96} raises a curse of dimensionality. To deal with it, we may want to exploit the symmetries of the polytope¹¹. Indeed a careful inspection of these 7 inequalities show that many variables come together and can be aggregated to form new variables. The process of aggregation as well as the new variables are described in appendix 4 for the cases $k = 3$ and $k = 2$. By doing so, we move from 96 to 32 variables in the case $k = 3$ and from 8 to 6 variables in the case $k = 2$. Of course, the probability law on the lower dimensional vector is not anymore uniform. However, we show in appendix 4 that it belongs to the Dirichlet family. This parametric class of probabilities can be emulated in R programs.¹² We have performed several "bootstrap" simulations¹³. Unsurprisingly, they do not deliver really different results. For one of them, we have obtained a mean estimated volume equal to 0.074301, a variance equal to $6.99583573573574 \times 10^{-5}$, and a 95% empirical confidence interval equal to [0.058975, 0.092].

We deduce that the IAC probability of having a Condorcet winner is here around $0.074301 \times 8 = 59.44\%$ (quite close to the IC one) with a 95% chance of being in the interval [47.18%, 73.6%].¹⁴

¹¹The idea of exploiting symmetries already appear in Schürmann (2013).

¹²The codes of the R program to compute the probability of a polytope for a Dirichlet distribution is available from the authors upon request.

¹³i.e. `volume_estimator_bootstrap(N = ..., n = 96, n_par_bootstrap = ..., A = A, n_simu = ...)`.

¹⁴As compared to IC, the bootstrap has been performed for $N = 10^5$ (instead of $N = 10^6$ simulations) with a bootstrap of 1000 (instead of 2000) and 1000 draws.

5 Conclusion

The computed probability values that are of primary interest from this study are summarized in Table 1.¹⁵ Results are most evident for the case of two issues where the degree of dependence that IAC-based assumptions are known to introduce beyond the complete independence of IC-based assumptions causes a predictable increased likelihood for observing a Condorcet winner for both the unrestricted case and when separable preferences are introduced.

Table 1: Probability of a Condorcet Winner with Separable Preferences

Distribution	Issues	
	$k = 2$	$k = 3$
IC	.8245	.58
IC-Separable	.8333	.59
IAC	.8384	.58
IAC-Separable	.8750	.59

It is particularly of interest to note that the introduction of the condition of Separable Preferences also increases the probability of observing a Condorcet winner from the unrestricted cases for both IC and IAC. So, the addition of this simple model to restrict preferences to be in accord with the logically consistent behavior of separable preferences creates enough additional structure among voters' preferences rankings to increase the likelihood that a Condorcet winner will exist with both IC and IAC-based scenarios. These probabilities all predictably decrease and trend toward a common value for the case of three issues, but the critical point is that the same patterns consistently emerge.

6 Appendices

6.1 Appendix: 1 The R Program to Compute an Orthant Probability in the Gaussian Case

```

1 #We try to install the packages required to simulate multivariate
   gaussians and to do parallel computing
2 # If they are installed, we load them
3 neededPkgs i-c("foreach", "doParallel", mvtnorm")
4 for(pkg in neededPkgs) {
5   if (require (pkg, character. only=TRUE)) {
6     print (paste(pkg, "is loaded correctly"))

```

¹⁵This Table displays exact values for $k = 2$ and rounded estimates for $k = 3$. The IAC value for $k = 3$ comes from Feix and Rouet (1999).

```

7 } else {
8   print (paste("trying to install", pkg))
9   install. packages (pkg)
10  library (pkg)
11 }
12 }
13 rm (neededPkgs, pkg)
14
15
16 Simulation_all j-function (n,k, mu=rep(0,k), sigma= diag(k)) {
17 # Simulates and returns a named table of length 2^k with the empirical
      probabilities of every case
18 # n is the number of gaussian r.v. to simulate
19 # k is the size of one gaussian r.v.
20 # mu is the vector of means for the gaussian r.v., of length k, default
      set to 0
21 # sigma is the varaince-covariance matrix for the gaussian r.v., default
      is identity matrix (unitary independent gaussians)
22 if (length (mu) !=k) {stop("k different from the length of mu")}
23 # We initialize the counting vector with every case
24 Y j- expand.grid(rep(list(0:1),k))
25 # We create a column saying to which case each row corresponds, we set it
      as the index of the table
26 Y[,"hash"] j-apply (Y,1,function(x){return(paste(x,sep=" ",collapse=""))
      })
27 row.names(Y) j- Y$hash
28 Y$hash j- NULL
29 # We initialize the counting column in our table by setting it to 0 fpr
      each row
30 Y[,"total"] j-0
31 # We simulate all the gaussians first
32 input j- rmvnorm (n, mean=mu, sigma=sigma)
33
34 # sub-function that updates the counting table by adding +1 to the value
      of a case
35 'updt j-' j- function (x, ...,value){

```

```

36  x[value,"total"] i-x[value, "total"]+1
37  x
38  }
39  # sub-function that reads a gaussian r.v; observation and returns the
      corresponding case
40  one_pass i- function (value){
41    y i- sapply (value, function (i) {return (max(0, sign(i)))})
42    code i- paste (y,sep="",collapse="")
43    return (code)
44  }
45  # We read all gaussians and update the table each time, job's done !
46  for (i in 1:n) {
47    updt (Y) i- one_pass (input [i,])
48  }
49  Y$total i- Y$total/n
50
51  return (Y)
52 }
53
54 simulation_one i- function (n,k,mu=rep(0,k), sigma=diag(k), case=
      paste (rep(1,k), sep="", collapse="")){
55  # Simulates and returns the empirical probability of one particular case
56  # n is the number of gaussians r.v. to simulate
57  # k is the size of the gaussian r.v.
58  # mu is the vector of means for the gaussian r.v., of length k, default
      set to 0
59  # sigma is the variance-covariance matrix for the gaussian r.v., default
      is identity matrix (unitary independent gaussians)
60  # case is the case that we want to study ( default is "all positive"),
61  # defined as a charcater string of 0s and 1s corresponding to the
      components of the gaussian r.v. which are negative/positive
62  # ex : for k=4, case ="0001" means that every component except the last
      one is negative
63  # so default is coded to be "11...1"
64  if (length(mu) !=k) {stop (k different from the length of mu")}
65

```

```

66 # We simulate all the gaussian r.v.
67 input j- rmvnorm (n,mean=mu, sigma=sigma)
68
69 # Sub-function that reads a gaussian r.v. observation and returns
    corresponding case
70 one_pass j- function (value){
71   y j- sapply (value, function (i),{return (max(0,sign(i)))})
72   code j- paste (y,sep=" ",collapse=" ")
73   return (code)
74 }
75 # We convert all our simulated gaussians r.v. into a vector of the
    corresponding cases
76 codes j- apply(input, 1,function(i){one_pass(i)})
77
78 # We are only interested in one case, we get the empirical mean of this
    case appearing and the job's done !
79 y j- mean(codes == case)
80 print (paste ("empirical probability of case", case, "is :", y))
81 return (y)
82 }
83
84 simulation_parallel j- function (n_par_simu, k, n_simu =100, mimax = c(0,
    1), mu=rep(0,k), sigma = diag(k),
85                                     case = paste (rep(1,k), sep=" ", collapse=" ")){
86 # simulation_one can get pretty slow if n is too big
87 # This function overcomes this by computing in parallel several instances
    of simulation_one with a smaller n
88 # n_simu is the number of instances of simulation that we compute
89 # n_par_simu is the number of gaussians that we simulate in each instance
90 # In total, we simulate n_par_simu*n_simu gaussians
91 # The fact that we get an estimate of our probability for each instance
    means that we can build empirical confidence intervals
92 #and compute the varaince of our estimate
93 # also the argument minmax corresponds to a range we want to know the
    probability for our estimate to be into
94 # default is (0,1) so the empirical probability will be 1, you can change

```

it to get different probabilities

```
95
96 # Setup parallel backend to use many pocessors
97 cores=detectCores()
98 cl j- makeCluster (cores [1]-1) # We eave one idle core to not overload
      your computer
99 registerDoParallel (cl)
100
101 final j- foreach (i=1:n_simu, .combine= 'c', .export =c("simulation_one",
      "rmvnorm")) %dopar% {
102   y = simulation_one (n_par_simu, k,mu=mu, sigma=sigma, case=case)
103
104   y
105 }
106 stopCluster (cl)
107
108 print(paste("Empirical probability of case", case, "is : ", mean(final)))
109 print (paste ("The variance of our estimate is", var(final)))
110 q j- quantile (final, probs = c(0.025, 0.975))
111 print (paste (95% Empirical confidence interval is [" , q[1], " , " ,q[2],"]")
      )
112 print (paste ("Empirical probability of being in range [" ,minmax [1], " ,",
      minmax [2],"] is :",
      mean ((final j minmax[1] &(final j minmax[2])))
113
114 plot . new ()
115 hist (final, prob=TRUE, col="grey", xlab ="probability of case", ylab ="
      frequency",
116       main ="Distribution of the estimates")
117 lines (density (final), col="blue", lwd=2)
118
119 return (final)
120 }
121
122 simulation_bootstrap j- function (n,k,n_par_bootstrap = min (c(1, floor(0.5*
      n))) , n_simu =100, minmax =c(0,1), mu = rep (o,k), sigma = diag (k),
123       case = paste (rep(1,k), sep=" ",collapse="")){
```

```

124 # simulation_one can get pretty slow if n is too big
125 # This function overcomes this by first simulating a lot of gaussian r.v.
126 # and then computing several estimates of our probbaility by bootstrapping
      several samples (bootstrap is sampling with replacement)
127 # This avoids having to simulate a lot of gaussian r.v.
128 # n is the number of gaussian r.v. to simulate
129 # n_simu is the number of bootstraped samples that we'll use
130 # n_par_bootstraped is the size of each bootstraped sample
131 # The fact that we get an estimate of our probbaility for each instance
      means that we can build empirical confidence intervals
132 # and compute the variance of our estimate
133 # also the argument minmax corresponds to a range we want to know the
      probability of our estimate to be into
134 # default is (0,1) so the empirical probability will be 1, you can change
      it to get different probabilities

135
136 if (length (mu) !=k){stop ("k different from the length of mu")}
137
138 # Setup parallel backend to use many processors
139 cores=detectCores()
140 cl j-makeCluster (cores [1]-1) # We leave one idle core to not overload
      your computer
141 registerDoParallel (cl)
142
143 # We fist simulate all the gaussians r.v.
144 input j- rmvnorm (n,mean=mu, sigma =sigma)
145
146 #Sub-function that reads a gaussian r.v. observation and returns the
      corresponding case
147 one_pass j- function (value){
148   y j- sapply(value, function (i){return (max (0, sign (i)))})
149   code j- paste (y,sep=" ",collapse=" ")
150   return (code)
151 }
152
153 final j- foreach (i=1:n_simu, . combine= 'c') %dopar% { {

```



```

154 x j- input[sample (nrow(input), size=n_par_bootstrap, replace=T,]
155 # We convert all our simulated gaussian r.v. into a vector of this
      corresponding cases
156 codes j- apply (x, 1, function (i) {one_pass(i)})
157
158 # We are only interested in one case , we get the empirical mean of this
      case appearing and the job's done !
159 y j- mean (codes == case)
160
161 y
162 }
163 stopCluster (cl)
164
165 print (paste ("Empirical probability of case", case, "is : ", mean(final)))
166 print (paste ("The variance of our estimate is", var(final)))
167 q j- quantile (final, probs= c(0.025, 0.975))
168 print (paste ("95% Empirical confidence interval is [", q[1], ", ", q[2], "]")
169 print (paste("Empirical probability of being in range [", minmax [1], ", ",
      minmax [2], "] is : ",
170           mean (( final j minmax[1] & (final j minmax [2])))
171 plot.new()
172 hist(final,prob=TRUE, col="grey", xlab ="Probbaility of case ", ylab ="
      Frequency",
173       main ="Distribution of the estimates")
174 lines (density (final), col="blue", lwd=2)
175
176 return (final)
177 }
178
179 # NB 1: simulation_all is slower than simulation_one because it does more
      operations on a 2^k-sized table
180 # If you are only interested in a few values of if k is too big, use
      simulation_one
181
182 # NB 2/ simulation_parallel and simulation-bootstrap both use parallel
      computing to speed up computations

```

```

183 # simulation_bootstrap should be faster than simulation_parallel in general
      because we tend to simulate less gaussian r.v. for the bootstrap
184 # Also bootstrapping reduces variance for the same bias (0 here) in theory
185 # However, bootstrapping in practice can be tricky when you want to estimate
      a case where probability is low :
186 # The chance of getting the case when simulating the r.v. is low and the
      the chance of getting that case when bootstrapping is also low
187 # Therefore you should use simulation_bootstrap when you're estimating a
      probability which you think is high, simulation_parallel otherwise
188
189
190 # # # # EXAMPLES # # #
191 sigma_sept2 j-matrix (c(12,0,0,6,6,0,5,
192                        0,12,0,6,0,6,5,
193                        0,0,12,0,6,6,5,
193                        6,6,0,12,4,4,7,
195                        6,0,6,4,12,4,7,
196                        0,6,6,4,4,12,7,
197                        5,5,5,7,7,7,12) , nrow=7, ncol=7)
198 test j- simulation_all (1000,3)
199 test j- simulation_one (100000,3)
200
201 sigma_test j- matrix (c(1,0.5,0,0.5, 1,0,0,0,1), nrow=3, ncol=3)
202 test_all j- simulation_all (n=100, k=3,sigma=sigma_test)
203
204 test_one j- simulation_one (n=1000, k=7, sigma=sigma_sept, case ="
      1111111")
205
206 test j- simulation_bootstrap (n=100000, 7, n_par_bootstrap =10000, n_simu
      =100, sigma =sigma_sept2, minmax=c(0.07, 0.08))
207
208 test j- simulation_parallel (n_par_simu =10000, 7, n_simu =10000, sigma=
      sigma_sept2)

```

6.2 Appendix 2: The Exact Orthant Probability for IC and $k = 3$

Consider a 7-dimensional gaussian vector X with zero mean and matrix of variance-covariance Δ defined as follows:

$$\Delta = \begin{pmatrix} \frac{1}{4} & 0 & 0 & \frac{1}{8} & \frac{1}{8} & 0 & \frac{5}{48} \\ 0 & \frac{1}{4} & 0 & \frac{1}{8} & 0 & \frac{1}{8} & \frac{5}{48} \\ 0 & 0 & \frac{1}{4} & 0 & \frac{1}{8} & \frac{1}{8} & \frac{5}{48} \\ \frac{1}{8} & \frac{1}{8} & 0 & \frac{1}{4} & \frac{1}{4} & \frac{1}{4} & \frac{7}{48} \\ \frac{1}{8} & 0 & \frac{1}{8} & \frac{1}{4} & \frac{1}{4} & \frac{1}{4} & \frac{7}{48} \\ 0 & \frac{1}{8} & \frac{1}{8} & \frac{1}{4} & \frac{1}{4} & \frac{1}{4} & \frac{7}{48} \\ \frac{5}{48} & \frac{5}{48} & \frac{5}{48} & \frac{7}{48} & \frac{7}{48} & \frac{7}{48} & \frac{1}{4} \end{pmatrix}$$

We derive:

$$\Delta^{-1} = \begin{pmatrix} \frac{2018}{255} & \frac{386}{255} & \frac{386}{255} & -\frac{266}{85} & -\frac{266}{85} & \frac{142}{85} & -\frac{32}{17} \\ \frac{386}{255} & \frac{2018}{255} & \frac{386}{255} & -\frac{266}{85} & \frac{142}{85} & -\frac{266}{85} & -\frac{32}{17} \\ \frac{386}{255} & \frac{386}{255} & \frac{2018}{255} & \frac{142}{85} & -\frac{266}{85} & -\frac{266}{85} & -\frac{32}{17} \\ -\frac{266}{85} & -\frac{266}{85} & \frac{142}{85} & \frac{774}{85} & -\frac{42}{85} & -\frac{42}{85} & -\frac{48}{17} \\ -\frac{266}{85} & \frac{142}{85} & -\frac{266}{85} & -\frac{42}{85} & \frac{774}{85} & -\frac{42}{85} & -\frac{48}{17} \\ \frac{142}{85} & -\frac{266}{85} & -\frac{266}{85} & -\frac{42}{85} & -\frac{42}{85} & \frac{774}{85} & -\frac{48}{17} \\ -\frac{32}{17} & -\frac{32}{17} & -\frac{32}{17} & -\frac{48}{17} & -\frac{48}{17} & -\frac{48}{17} & \frac{192}{17} \end{pmatrix} \text{ and determinant } \Delta = \frac{425}{169869312}$$

Therefore the density of X is:

$$f(x_1, x_2, x_3, x_4, x_5, x_6, x_7) = \frac{1}{\sqrt{\frac{425}{169869312}} 2\pi^{\frac{7}{2}}} \exp - \frac{x^t \Delta^{-1} x}{2}$$

Since

$$\begin{pmatrix} x_1 & x_2 & x_3 & x_4 & x_5 & x_6 & x_7 \end{pmatrix} \begin{pmatrix} \frac{2018}{255} & \frac{386}{255} & \frac{386}{255} & -\frac{266}{85} & -\frac{266}{85} & \frac{142}{85} & -\frac{32}{17} \\ \frac{386}{255} & \frac{2018}{255} & \frac{386}{255} & -\frac{266}{85} & \frac{142}{85} & -\frac{266}{85} & -\frac{32}{17} \\ \frac{386}{255} & \frac{386}{255} & \frac{2018}{255} & \frac{142}{85} & -\frac{266}{85} & -\frac{266}{85} & -\frac{32}{17} \\ -\frac{266}{85} & -\frac{266}{85} & \frac{142}{85} & \frac{774}{85} & -\frac{42}{85} & -\frac{42}{85} & -\frac{48}{17} \\ -\frac{266}{85} & \frac{142}{85} & -\frac{266}{85} & -\frac{42}{85} & \frac{774}{85} & -\frac{42}{85} & -\frac{48}{17} \\ \frac{142}{85} & -\frac{266}{85} & -\frac{266}{85} & -\frac{42}{85} & -\frac{42}{85} & \frac{774}{85} & -\frac{48}{17} \\ -\frac{32}{17} & -\frac{32}{17} & -\frac{32}{17} & -\frac{48}{17} & -\frac{48}{17} & -\frac{48}{17} & \frac{192}{17} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \\ x_7 \end{pmatrix}$$

is equal to:

$$\begin{aligned} & x_1 \left(\frac{2018}{255} x_1 + \frac{386}{255} x_2 + \frac{386}{255} x_3 - \frac{266}{85} x_4 - \frac{266}{85} x_5 + \frac{142}{85} x_6 - \frac{32}{17} x_7 \right) - \\ & x_4 \left(\frac{266}{85} x_1 + \frac{266}{85} x_2 - \frac{142}{85} x_3 - \frac{774}{85} x_4 + \frac{42}{85} x_5 + \frac{42}{85} x_6 + \frac{48}{17} x_7 \right) - \end{aligned}$$

$$\begin{aligned}
& x_5 \left(\frac{266}{85}x_1 - \frac{142}{85}x_2 + \frac{266}{85}x_3 + \frac{42}{85}x_4 - \frac{774}{85}x_5 + \frac{42}{85}x_6 + \frac{48}{17}x_7 \right) - \\
& x_6 \left(\frac{266}{85}x_2 - \frac{142}{85}x_1 + \frac{266}{85}x_3 + \frac{42}{85}x_4 + \frac{42}{85}x_5 - \frac{774}{85}x_6 + \frac{48}{17}x_7 \right) - \\
& x_7 \left(\frac{32}{17}x_1 + \frac{32}{17}x_2 + \frac{32}{17}x_3 + \frac{48}{17}x_4 + \frac{48}{17}x_5 + \frac{48}{17}x_6 - \frac{192}{17}x_7 \right) + \\
& x_2 \left(\frac{386}{255}x_1 + \frac{2018}{255}x_2 + \frac{386}{255}x_3 - \frac{266}{85}x_4 + \frac{142}{85}x_5 - \frac{266}{85}x_6 - \frac{32}{17}x_7 \right) + \\
& x_3 \left(\frac{386}{255}x_1 + \frac{386}{255}x_2 + \frac{2018}{255}x_3 + \frac{142}{85}x_4 - \frac{266}{85}x_5 - \frac{266}{85}x_6 - \frac{32}{17}x_7 \right)
\end{aligned}$$

We deduce that the probability that $(1, 1, 1)$ is a Condorcet winner is equal to:

$$\begin{aligned}
& \int_0^1 \int_0^1 \int_0^1 \int_0^1 \int_0^1 \int_0^1 \int_0^1 \frac{1}{\sqrt{\frac{425}{169869312} 2\pi^{\frac{7}{2}}}} \times \\
& \exp - \frac{x_1 \left(\frac{2018}{255}x_1 + \frac{386}{255}x_2 + \frac{386}{255}x_3 - \frac{266}{85}x_4 - \frac{266}{85}x_5 + \frac{142}{85}x_6 - \frac{32}{17}x_7 \right)}{2} \times \\
& \exp - \frac{-x_4 \left(\frac{266}{85}x_1 + \frac{266}{85}x_2 - \frac{142}{85}x_3 - \frac{774}{85}x_4 + \frac{42}{85}x_5 + \frac{42}{85}x_6 + \frac{48}{17}x_7 \right)}{2} \times \\
& \exp - \frac{-x_5 \left(\frac{266}{85}x_1 - \frac{142}{85}x_2 + \frac{266}{85}x_3 + \frac{42}{85}x_4 - \frac{774}{85}x_5 + \frac{42}{85}x_6 + \frac{48}{17}x_7 \right)}{2} \times \\
& \exp - \frac{-x_6 \left(\frac{266}{85}x_2 - \frac{142}{85}x_1 + \frac{266}{85}x_3 + \frac{42}{85}x_4 + \frac{42}{85}x_5 - \frac{774}{85}x_6 + \frac{48}{17}x_7 \right)}{2} \times \\
& \exp - \frac{-x_7 \left(\frac{32}{17}x_1 + \frac{32}{17}x_2 + \frac{32}{17}x_3 + \frac{48}{17}x_4 + \frac{48}{17}x_5 + \frac{48}{17}x_6 - \frac{192}{17}x_7 \right)}{2} \times \\
& \exp - \frac{x_2 \left(\frac{386}{255}x_1 + \frac{2018}{255}x_2 + \frac{386}{255}x_3 - \frac{266}{85}x_4 + \frac{142}{85}x_5 - \frac{266}{85}x_6 - \frac{32}{17}x_7 \right)}{2} \times \\
& \exp - \frac{x_3 \left(\frac{386}{255}x_1 + \frac{386}{255}x_2 + \frac{2018}{255}x_3 + \frac{142}{85}x_4 - \frac{266}{85}x_5 - \frac{266}{85}x_6 - \frac{32}{17}x_7 \right)}{2} \times \\
& dx_1 dx_2 dx_3 dx_4 dx_5 dx_6 dx_7
\end{aligned}$$

6.3 Appendix 3 : The R Program to Compute the Volume of a Polytope

```
1# We try to install the packages required to do parallel computing
2 # If there are installed, we load them
3 neededPkgs i- c("foreach","doParallel", "hitandrun")
4 for (pkg in neededPkgs) {
5   if (require (pkg, character.only=TRUE)) {
6     print (paste(pkg, "is loaded correctly"))
7   } else {
8     print (paste("trying to install", pkg))
9     install . packages (pkg)
10    library (pkg)
11  }
12 }
13 rm (neededPkgs , pkg)
14
15 volume_estimator i-function (N,n,A=diag(n),b=c(1/2,rep(nrow(A)/2, (
16   nrow(A)-1))), data=NULL){
17   gen_simplex i- function(n){
18     #This function generates a random point in the positive simplex of  $\mathbb{R}^n$ ,
19     that is,  $(x_1, \dots, x_n)$  such that
20     # $0 \leq x_i \leq 1$  for all  $i$ 
21     # $\sum(x_i) = 1$ 
22     if (n<2) {stop("n must be higher than 2 !") }
23     xj - runif (n)
24     while (sum (x) > 1) {
25       x i- runif (n)
26     }
27     return (x)
28   }
29   is_in_h i- function (x, A, b){
30     #Tests if a point of the simplex is in the subspace defined by  $Ax \leq b$ 
31     y i- as.vector (A %*% x)
32     return ( as.integer (all(y <= as.vector (b))))
```

```

32 }
33
34 if (is.null(data)){
35   X j- sapply(rep(n,N), gen.simplex) # simulated simplex points
36 }else{
37   X j- data # If we've already simulated the points, we just import them
           from the argument
38   }
39
40
41 Y j- apply (X, 2, function (x){return (is.in.h(x, A,b))})
42
43 return (mean(Y))
44 }
45
46 volume_estimator_bootstrap j-function (N,n,n_par_bootstrap = min(c(1,
           floor (0.5*N))), n_simu =100, minmax = c(0,1) ,
47                                     A, b =c(1/2, rep(n/2, (n-1))))}
48 #volume_estimator ca get pretty slow if N is too big
49 # This function overcomes this by first simulating a lot simplex points
50 # and then computing several estimates of our volume by bootstrapping
           several samples (bootstrap is sampling with replacement )
51 # This avoids having to simulate a lot of simplex points
52 # N is the number of points to simulate
53 # n is the dimension of the space
54 # n_simu is the number of bootstrapped samples that we'll use
55 # n_par_bootstrapped is the size of each bootstrapped sample
56 # The fact that we get an estimate of our volume for each instance means
           that we can build empirical confidence intervals
57 # and compute the variance of our estimate
58 # also the argument minmax corresponds to a range we want to know the
           probability of our estimate to be into
59 # default is (0,1) so the empirical probability will be 1 , you can change
           it to get different probabilities
60 # A and b are the same as before
61

```

```

62 # Setup parallel backend to use many processors
63 cores=detextCores()
64 cl|- makeCluster (cores(1)-1) # We leave one idle core to not overload
      your computer
65 registerDoParallel (cl)
66
67 input j-sapply (rep(n,N), gen_simplex) # Simulated simplex points
68
69 final j- foreach (i=1:n_simu, .combine='c', .export=c("volume_estimator
      ")) %dopar% {
70   x j- input [, sample(ncol(input), size=n_par_bootstrap, replace=T)]
71   # We sample with replacement from our simulated simplex points
72   volume j- volume_estimator (n_par_bootstrap, n, A, b, data=x)
73
74   volume
75 }
76 stopCluster (cl)
77
78 print (paste ("Mean estimated volume is /", mean (final)))
79 print (paste("The variance of our estimate is", var(final)))
80 qi-quantile(final, probs=c(0.025, 0.975))
81 print(paste("95% Empirical confidence interval is [ ",q[1]," ",q[2],"]")
      )
82 print(paste(" Empirical probability of the volume being in range [",
      minmax[1]," ",minmax[2],"] is : ",
83           mean((final<minmax[1]&(final >minmax[2])))
84 plot.new()
85 hist(final, prob=TRUE, col="grey", xlab="Estimated volume", ylab="
      Frequency",
86       main = " Distribution of the estimates")
87 lines(density(final), col="blue", lwd=2)
88
89 return (final)
90 }
91
92

```

```

93 # # # # EXAMPLES # # #
94
95 A j-matrix (0,nrow=7, ncol=96) #We'll build the matrix in the paper
    to get a working example
96 for (i in c(1 :12, 25:48, 73:84)) {A[1,i] j-1}
97 for (i in c(1 :24, 37:48, 61:72)) {A[2,i] j-1}
98 for (i in c(1 :36, 49:60,)) {A[3,i] j-1}
99 for (i in c(1 :14, 17:20, 27:28,33:48,61:62,65:68,75:76,81:84)) {A[4,i] j-1}
100 for (i in c(1 :18, 25:36, 49:54,79:84)) {A[5,i] j-1}
101 for (i in c(1 :28, 33:34, 41:44, 47:52,57:58,65:68,71:72)) {A[6,i] j-1}
102 for (i in c(1 :20, 23:28,30, 32:36,38, 40:49,51,65,67,81,83)) {A[7,i] j-1}
103
104 #test_volume j-volume_estimator (1000, 96,A) # Too long, rejection metod
    is useless for such a high dimension !
105
106 gen.simplexes j- function (N,N){
107 # This function generates N random points in the positive simplex of  $R^n$ ,
    that is,  $(x_1, \dots, x_n)$  such that
108 #  $0_i = x_i = 1$  for all i
109 #  $\sum(x_i) = 1$ 
110 if (n<2){stop("n must be higher than 2 !")}
111 M j- matrix (0,nrow=n+1, ncol=n)
112 M[n+1,] j-1
113 M[row(M) ==col(M)] j- -1
114 constraints j- list (constr =M,
115                       dir =rep("j",n+1),
116                       rhs = c(rep(0,n),1))
117
118 y j- hitandrum (constraints,
119                n.samples =N ,
120                x0 = rep(1/(n+1),n))
121
122 return(y)
123 }
124
125

```


6.4 Appendix 4 : Moving from Uniform to Dirichlet. How to deal with the Curse of Dimensionality

Let S_m be the m -dimensional unit simplex i.e. the set of vectors p in \mathbb{R}^m such that:

$$p_i \geq 0 \text{ for all } i = 1, 2, \dots, m \text{ and } \sum_{i=1}^m p_i = 1$$

The uniform distribution on S_m is a special case of a Dirichlet distribution. The Dirichlet distribution of order m with parameters $\alpha_1, \dots, \alpha_m > 0$ has a probability density function with respect to Lebesgue measure defined by :

$$f(x; \alpha) = \frac{1}{B(\alpha)} \prod_{k=1}^m x_k^{\alpha_k - 1} \text{ where } \alpha \equiv (\alpha_1, \dots, \alpha_m)$$

with¹⁶:

$$B(\alpha) = \frac{\prod_{k=1}^m \Gamma(\alpha_k)}{\Gamma(\sum_{k=1}^m \alpha_k)}$$

When $\alpha \equiv (1, \dots, 1)$, we obtain the uniform distribution. We know a lot of things on this parametric family of distributions. For our purpose, we only need this property called *aggregation property*. It asserts that if p follows a Dirichlet distribution of order m with parameters α , then if we sum the coordinates i and j leaving the others the same, the new vector $(p_1, \dots, p_i + p_j, \dots, p_m)$ follows a Dirichlet distribution of order $m - 1$ with parameters $(\alpha_1, \dots, \alpha_i + \alpha_j, \dots, \alpha_m)$.

How to use that in our case ? Here is the trick. We observe that in the seven inequalities many variables always come together in each inequality. If we proceed to a complete description of these symmetries, we create 32 new variables by adding old ones (either 2, 4 or 12) of them as follows.

$$q_1 = p_1 + p_2 + p_3 + p_4 + p_5 + p_6 + p_7 + p_8 + p_9 + p_{10} + p_{11} + p_{12}$$

$$q_2 = p_{19} + p_{20}$$

$$q_3 = p_{15} + p_{16}$$

¹⁶ Γ denotes the Gamma function. In particular if x is an integer $\Gamma(x) = (x - 1)!$

$$q_4 = p_{21} + p_{23}$$

$$q_5 = p_{22} + p_{24}$$

$$q_6 = p_{13} + p_{14} + p_{17} + p_{18}$$

$$q_7 = p_{25} + p_{26}$$

$$q_8 = p_{35} + p_{36}$$

$$q_9 = p_{29} + p_{31}$$

$$q_{10} = p_{30} + p_{32}$$

$$q_{11} = p_{27} + p_{28} + p_{33} + p_{34}$$

$$q_{12} = p_{37} + p_{39}$$

$$q_{13} = p_{38} + p_{40}$$

$$q_{14} = p_{45} + p_{46}$$

$$q_{15} = p_{41} + p_{42}$$

$$q_{16} = p_{43} + p_{44} + p_{47} + p_{48}$$

$$q_{17} = p_{49} + p_{51}$$

$$q_{18} = p_{50} + p_{52}$$

$$q_{19} = p_{53} + p_{54}$$

$$q_{20} = p_{57} + p_{58}$$

$$q_{21} = p_{55} + p_{56} + p_{59} + p_{60}$$

$$q_{22} = p_{61} + p_{62}$$

$$q_{23} = p_{65} + p_{67}$$

$$q_{24} = p_{66} + p_{68}$$

$$q_{25} = p_{71} + p_{72}$$

$$q_{26} = p_{63} + p_{64} + p_{69} + p_{70}$$

$$q_{27} = p_{75} + p_{76}$$

$$q_{28} = p_{79} + p_{80}$$

$$q_{29} = p_{81} + p_{83}$$

$$q_{30} = p_{82} + p_{84}$$

$$q_{31} = p_{73} + p_{74} + p_{77} + p_{78}$$

$$q_{32} = p_{85} + p_{86} + p_{87} + p_{88} + p_{89} + p_{90} + p_{91} + p_{92} + p_{93} + p_{94} + p_{95} + p_{96}$$

From the aggregation property, since the vector p follows a Dirichlet distribution of order 96 with parameter $(1, 1, 1, \dots, 1)$, we deduce that the vector q follows a Dirichlet distribution of order 32 with parameters:

$$\alpha = (12, 2, 2, 2, 2, 4, 2, 2, 2, 2, 4, 2, 2, 2, 2, 4, 2, 2, 2, 2, 4, 2, 2, 2, 2, 4, 2, 2, 2, 2, 4, 12)$$

So, we have moved from 96 dimensions to 32 dimensions but at the cost of moving from a simple Dirichlet (the uniform) to a more sophisticated one. The 7 inequalities with the 32 variables write as follows.

$$q_1 + q_7 + q_8 + q_9 + q_{10} + q_{11} + q_{12} + q_{13} + q_{14} + q_{15} + q_{16} + q_{27} + q_{28} + q_{29} + q_{30} + q_{31} \geq \frac{1}{2};$$

$$q_1 + q_2 + q_3 + q_4 + q_5 + q_6 + q_{12} + q_{13} + q_{14} + q_{15} + q_{16} + q_{22} + q_{23} + q_{24} + q_{25} + q_{26} \geq \frac{1}{2};$$

$$q_1 + q_2 + q_3 + q_4 + q_5 + q_6 + q_7 + q_8 + q_9 + q_{10} + q_{11} + q_{17} + q_{18} + q_{19} + q_{20} + q_{21} \geq \frac{1}{2};$$

$$q_1 + q_2 + q_6 + q_8 + q_{11} + q_{12} + q_{13} + q_{14} + q_{15} + q_{16} + q_{22} + q_{23} + q_{24} + q_{27} + q_{29} + q_{30} \geq \frac{1}{2};$$

$$q_1 + q_3 + q_6 + q_7 + q_8 + q_9 + q_{10} + q_{11} + q_{14} + q_{16} + q_{17} + q_{18} + q_{19} + q_{28} + q_{29} + q_{30} \geq \frac{1}{2};$$

$$q_1 + q_2 + q_3 + q_4 + q_5 + q_6 + q_7 + q_{11} + q_{15} + q_{16} + q_{17} + q_{18} + q_{20} + q_{23} + q_{24} + q_{25} \geq \frac{1}{2};$$

$$q_1 + q_2 + q_3 + q_5 + q_6 + q_7 + q_8 + q_{10} + q_{11} + q_{13} + q_{14} + q_{15} + q_{16} + q_{17} + q_{23} + q_{29} \geq \frac{1}{2}.$$

The matrix 7×32 and the column vector of this polytope, denoted H^D , are respectively:

$$\begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 1 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 1 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 1 & 1 & 1 & 1 & 0 & 1 & 1 & 0 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \end{pmatrix}$$

$$\begin{pmatrix} \frac{1}{2} \\ \frac{1}{2} \\ \frac{1}{2} \\ \frac{1}{2} \\ \frac{1}{2} \\ \frac{1}{2} \\ \frac{1}{2} \\ \frac{1}{2} \\ \frac{1}{2} \end{pmatrix}$$

The probability of being in H^D when q is drawn over S_{32} according to the Dirichlet $(12, 2, 2, 2, 2, 4, 2, 2, 2, 2, 4, 2, 2, 2, 2, 4, 2, 2, 2, 2, 4, 2, 2, 2, 2, 4, 2, 2, 2, 2, 4, 2, 2, 2, 2, 4, 12)$ is the same as the probability of being in H when p is drawn uniformly over S_{96} .

When $k = 2$, we could have done the same thing. With the original proportions, we would have the following 3 inequalities where p is a vector in S_8 :

$$\begin{aligned} p_1 + p_2 + p_3 + p_5 &> \frac{1}{2} \\ p_1 + p_2 + p_3 + p_4 &> \frac{1}{2} \\ p_1 + p_2 + p_5 + p_6 &> \frac{1}{2} \end{aligned}$$

This polytope is described by the following matrix and right hand-side vector:

$$\begin{pmatrix} 1 & 1 & 1 & 0 & 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 \end{pmatrix}$$

$$\begin{pmatrix} \frac{1}{2} \\ \frac{1}{2} \\ \frac{1}{2} \end{pmatrix}$$

We can move to 6 new variables:

$$q_1 = p_1 + p_2$$

$$q_2 = p_3$$

$$q_3 = p_4$$

$$q_4 = p_5$$

$$q_5 = p_6$$

$$q_6 = p_7 + p_8$$

The system of equations writes now:

$$\begin{aligned} q_1 + q_2 + q_4 &> \frac{1}{2} \\ q_1 + q_2 + q_3 &> \frac{1}{2} \\ q_1 + q_4 + q_5 &> \frac{1}{2} \end{aligned}$$

The matrix and right-hand side are now:

$$\begin{pmatrix} 1 & 1 & 0 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 1 & 0 \end{pmatrix}$$

$$\begin{pmatrix} \frac{1}{2} \\ \frac{1}{2} \\ \frac{1}{2} \end{pmatrix}$$

and the vector q displays a Dirichlet distribution with parameters $(2, 1, 1, 1, 1, 2)$ over S_6 . The probability of being in H^D when q is drawn over S_6 according to the Dirichlet $(2, 1, 1, 1, 1, 2)$ is the same as the probability of being in H when p is drawn uniformly over S_8 .

References

- [1] Bell, C.E. (1981) "A Random Voting Graph Almost Surely has a Hamiltonian Cycle when the Number of Alternatives is Large", *Econometrica*, 49, 1597–1603.
- [2] Casella, A. and T. Palfrey (2015) "Trading Votes for Votes: A Decentralized Matching Algorithm", Mimeo, Department of Economics, Columbia University.

- [3] Feix, M.R and J.L. Rouet (1999) "L'Espace des Phases Electoral et les Statistiques Quantiques. Application à la Simulation Numérique", Université d'Orléans, Mimeo.
- [4] Ferejohn, J. (1974) "Sour Notes on the Theory of Vote Trading", Caltech, Social Science Working Paper No.41.
- [5] Fey, M. (2008) "Choosing from a Large Tournament", *Social Choice and Welfare*, 31, 301–309.
- [6] Garman, M. and M. Kamien (1968) "Paradox of Voting: Probability Calculations", *Behavioral Science*, 13, 306-316.
- [7] Gerhlein, W.V. (2006) *Condorcet's Paradox*, Springer.
- [8] Gehrlein, V.W. and P. C. Fishburn (1976) The Probability of the Paradox of Voting: A Computable Solution", *Journal of Economic Theory*, 13, 14–25,
- [9] Gerhlein, W.V. and D. Lepelley (2010) *Voting Paradoxes and Group Coherence: The Condorcet Efficiency of Voting Rules*. Springer.
- [10] Gupta, S.S. (1963) "Bibliography on the Multivariate Normal Integrals and Related Topics, *Annals of Mathematical Statistics*, 34, 829-838.
- [11] Hollard, G. and M. Le Breton (1996) "Logrolling and a McGarvey Theorem for Separable Tournaments", *Social Choice and Welfare*, 13, 451-455.
- [12] Kadane, J., 1972, "On the Division of the Question", *Public Choice*, 13, 47-54.
- [13] Koehler, D. (1975) "Vote Trading and the Voting Paradox: A Proof of Logical Equivalence", *American Political Science Review*, 69, 954-960.
- [14] Krishnamoorthy, M.S. and M. Raghavachari (2005) "Condorcet Winner Probabilities: A Statistical Perspective", Mimeo, Department of Computer Science, Rensselaer Polytechnic Institute.
- [15] May, R.M. (1971) "Some Mathematical Remarks on the Paradox of Voting", *Behavioral Science*, 16, 143-151.
- [16] Miller, N.R. (1975) "Logrolling and the Arrow Paradox: A Note", *Public Choice*, 21, 107-110.
- [17] Miller, N.R. (1977) "Logrolling, Vote Trading and the Paradox of Voting: A Game Theoretical Overview", *Public Choice*, 30, 51-75.

- [18] Miller, N.R. (1994) *Committees, Agendas and Voting*, in *Fundamentals of Pure and Applied Economics*, Economics and Political Science Section, Ferejohn, J (Ed), Harwood Academic Publishers.
- [19] Niemi, R. G. and H.F. Weisberg (1968) "A Mathematical Solution for the Probability of the Paradox of Voting", *Behavioral Science*, 13, 317-323.
- [20] Riker, W. and S. Brams (1973) "The Paradox of Vote Trading", *American Political Science Review*, 67, 1235-1247.
- [21] Schürmann, A. (2013) "Exploiting Polyhedral Symmetries in Social Choice", *Social Choice and Welfare*, 40, 1097-1110.
- [22] Schwartz, T. (1975) "Vote Trading and Pareto Efficiency", *Public Choice*, 24, 101-110.