Pure and Applied Mathematics Journal
2023; 12(3): 49-58
http://www.sciencepublishinggroup.com/j/pamj
doi: 10.11648/j.pamj.20231203.12
ISSN: 2326-9790 (Print); ISSN: 2326-9812 (Online)

Implementation of the VMAVA Method in Order to Make Applications with a Large Number of Candidates and Voters

Koumbèbarè Kambiré, Zoïnabo Savadogo, Frédéric Nikiéma

Department of Mathematics, Laboratory of Numerical Analysis, Computer Science and Biomathematics, Joseph Ki-Zerbo University, Ouagadougou, Burkina Faso

Email address:
kambirekhamed@gmail.com (Koumbèbarè Kambiré), serezenab@yahoo.fr (Zoïnabo Savadogo), fredericnikiema0@gmail.com (Frédéric Nikiéma)

To cite this article:
Koumbèbarè Kambiré, Zoïnabo Savadogo, Frédéric Nikiéma. Implementation of the VMAVA Method in Order to Make Applications with a Large Number of Candidates and Voters. *Pure and Applied Mathematics Journal*. Vol. 12, No. 3, 2023, pp. 49-58.
doi: 10.11648/j.pamj.20231203.12

Received: June 29, 2023; **Accepted**: August 1, 2023; **Published**: August 31, 2023

Abstract: Nowadays, we see everywhere in the world and particularly in Africa, revolts following elections. It is therefore important to find a voting method that represents consensus. It should also be noted that despite the votes, there are candidates who do not agree to recognize the results after their defeat. Faced with this situation, the ideal would be to find a good method that can result in less contrast. This is how the VMAVA method was developed. We notice that it is a good method because it fulfills good properties. However in the VMAVA method, we notice that the numerical applications have been made on voting situations where there are four candidates and five voters, sometimes four candidates and four voters, at most five candidates and seven voters. In our work, we are therefore interested in the implementation of the VMAVA method to facilitate calculations in voting situations where there are for example ten, fifteen candidates and ten thousand, twenty thousand voters. To do this, we have built two main functions, one which is responsible for choosing the elected candidate (s) on the basis of the total number of approvals and the other which makes it possible to decide between possible ties using the arithmetic averages of the candidates. Despite some difficulties encountered in this task, we have achieved quite interesting and concordant results.

Keywords: Vote, Voter, Candidate, Approval, Arithmetic Mean, Implementation

1. Introduction

The development of the computer science sector has led to many changes in our so-called modern societies. Thus many changes are observable in several fields, such as economics, health science, social choice theory, etc. The interaction between computer science and social choice theory is the key point of our work. First of all, it is important to elucidate social choice theory in order to clearly situate its context of interaction with computing. The object of social choice is the selection of options by a group of individuals (in an almost equivalent way, one can also interpret social choice as an individual choice in the case of multiple criteria, the criteria then corresponding to individuals and the individual to society) [13]. In simpler terms, social choice theory is concerned with the analysis of voting procedures and their properties. Voting occurs in various everyday situations. Indeed, in high schools, colleges, trade unions and universities, the various delegates are most often appointed by vote. But later with the advent of democracy in our societies, demanding more participation, transparency and credibility in collective decision-making, we better discover the notion of voting through political elections. According to Patrick Blanchenay [12], the purpose of an election is to gather the opinion of a population and use it to make a decision that concerns this population. There are different situations in the literature in which voting makes it possible to aggregate individual preferences into a collective decision in order to produce a result. These situations vary due to the size of the election, the number of voters and candidates for selection. In terms of procedures and methods of aggregation, face-to-face, legislative and municipal elections cannot be compared to elections for class delegates or union structures. Since the voting contexts are many and varied, then the aggregation mechanisms also depend on them. Thus in large-scale elections, the number of voters is so large that it is not easy to manually calculate the outcome of the vote. It is in this context that we considered it useful to implement the VMAVA method to meet the need for large-scale elections in order to facilitate the calculation and save time. In, we will first present the VMAVA method, then propose the principle of its implementation and finally make applications.

2. State of the Art

2.1. A Few Reminders on Voting by Approval

In this section, we introduce some fundamental notions of social choice theory. Thus we recall that social choice is concerned with the procedures for aggregating individual preferences into collective preferences. Voting is therefore a means of meeting this need. In this context, group members are called voters and the objects they prefer are called candidates. The system of approval voting is an area of social choice, which has long been debated. Voting by approval or assent vote is a simple voting system studied and defended by theoreticians since the 1970s [4]. It was proposed in the 1970s, in particular by the research [11] in this voting system, voters can vote for one or more candidates, if they wish. Each voter draws up a list of all the candidates he wishes to support with his vote. The candidate who receives the most votes is elected. To do this, the voter classifies all the candidates into two groups: Candidates not receive then he gives his votes to each of the candidates of the subset to be approved. Approval voting can be used for both single-winner and multiple-winner elections. The first question that can be asked is the question of the representation of individual preferences. For some time, social choice specialists have been studying and characterizing different voting methods, trying to select the best one and discard the bad ones. But so far, none of these voting methods is used only for major political elections. The two most popular methods are first-past-the-post and two-round majority voting. But at some point, some countries have resorted to other voting methods that supposedly better reflect the reality of the people. According to the research [3] the majority system requires very little information from the voter: he only has the right to nominate a single candidate. On the other hand, for the vote by approval, voters are asked to rate each candidate. The elected candidate is the one who receives the highest sum of scores. Voting by approval amounts to a vote by rating where voters can only give a rating of 0 or 1 to each candidate [2]. However in some situations, the scoring is done on a scale. According to Catherine Petillon [15], the voter evaluates all the candidates according to a predefined scale (for example between 0 and 20, or else with a score among (-1, 0, 1)). The type of approval vote that is the subject of our study proceeds by scoring in accordance with a predefined ordinal appreciation scale.

1) Définitions:

A voter is anyone who has preferences over a set of candidates. In this work, we denote by:

$V = \{1, \dots \dots \dots \dots, n\}$ All voters and $C = \{c_1, \dots \dots \dots \dots, c_m\}$ The set of candidates Note that the set of candidates can be a set of individuals or actions. Voters' preferences can be ordinal, dichotomous or cardinal

2) Les différents types de préférences

a) - Préférences ordinals

We assume that the preferences of a voter i are given by a preference relation. A preference relation is a binary and complete relation denoted $\mathcal{R} \subseteq C \times C$ such as for any pair of candidates $(c_1, c_2) \in \mathcal{R}$, we will note $c_1 \mathcal{R}_i c_2$ if c_1 is preferred to c_2 by voter i. In other words, a binary relation is complete if each voter i is able to compare any pair of candidates (c_1, c_2). Thus, we have the following situations:

For any couple $(c_1, c_2) \in \mathcal{R}$, soit $c_1 \mathcal{R}_i c_2$ ou $c_2 \mathcal{R}_i c_1$ or both if the voter is indifferent between the two candidates c_1 et c_2. Moreover, the preference relation is often required to be transitive and antisymmetric. Transitivity means that for any $c_1, c_2, c_3 \in C$ such as $c_1 \mathcal{R}_i c_2$ and $c_2 \mathcal{R}_i c_3$, that we have $c_1 \mathcal{R}_i c_3$

Antisymmetry means that for any $c_1, c_2 \in C$ if $c_1 \mathcal{R}_i c_2$ so no $(c_2 \mathcal{R}_i c_1)$, which has the main consequence of not allowing indifference. In this context, a preference profile is defined by:

P={P1,..., Pn} which is then a vector containing a preference relation per voter. If a preference relation is both transitive and antisymmetric, then it is called a preference order.

Exemple 3.2.1: Consider a set of five voters {1, 2, 3, 4, 5}, a set of three candidates $\{c_1, c_2, c_3\}$, and the following preference profile P:

P1: $c_1 \succ c_2 \succ c_3$;
P2: $c_2 \succ c_1 \succ c_3$;
P3: $c_1 \succ c_3 \succ c_2$
P4: $c_1 \succ c_2 \succ c_3$;
P5: $c_3 \succ c_2 \succ c_1$

The preferences of voter 1 expresses the fact that he prefers c_1 à c_2, qu'il préfère c_2 à c_3 and so c_1 à c_3 by transitivity.

b) - Cardinal Preferences

According to Nathanaël BARROT [14], modeling preferences by a binary relationship does not make it possible to express an intensity of preference or even to compare the well-being of two voters. To overcome this problem, the cardinal preferences seem the best indicated. It is assumed in this case that a voter can express the satisfaction that a candidate gives him by positioning him on a predefined cardinal scale, in general a scale of scores. A voter's preferences are given by a score vector P_i of size m, containing one score per candidate. A preference profile is defined by

P = $\{P_1, \dots\dots\dots\dots\dots, P_n\}$ is then a set of score vectors.

Exemple 3.2.2. Consider a set of five voters $\{1, 2, 3, 4\}$, a set of three candidates $\{c_1, c_2, c_3\}$ and a scale of scores ranging from 1 to 4. The profile is as follows: P1: (4, 2, 2); P2: (2, 1, 3); P3: (2, 3, 4) P4: (1, 1, 2); P5: (3, 3, 1) Voter 1 awards four points to the candidate c_1 and two points to each candidate c_2 et c_3

c) Préférences dichotomique

With respect to dichotomous preferences, each voter separates the set of candidates into two subsets: the set of approved candidates and the set of those they reject. Voters are indifferent between two candidates from the same subset. The preferences of a voter i are therefore expressed by the data of one of the two subsets of candidates. A dichotomous preference profile P is then a set of candidate subsets

Exemple 3.2.3: Consider a set of five voters {1, 2, 3, 4, 5, 6} and a set of three candidates $\{c_1, c_2, c_3\}$. The profile P of the preferences is the following: P1: $\{c_1, c_2\}$; P2: $\{c_1\}$; P3: $\{c_2, c_3,\}$;

P4: $\{c_1, c_2, c_3\}$; P5: $\{c_3\}$; P6: $\{c_3, c_1,\}$ Voter 5 only approves the candidate c_3 while voter 4 approves all candidates c_1, c_2, c_3 In the context of computational voting [14] defines the dichotomous preferences of a voter i using a binary vector $P_i \in \{0, 1\}^m$ with the meaning that the coordinate of the binary vector P_i is equal to 1 if voter i approves of the candidate $c_{j, j=1, \ldots\ldots\ldots, m}$ and 0 otherwise. A preference profile $P = \{P_{1, \ldots\ldots\ldots\ldots\ldots,} P_n\}$ is then a vector containing a binary vector given by a voter. He illustrates this example as follows:

Exemple 3.2.4: Consider a set of five voters {1, 2, 3, 4, 5}, a set of three candidates $\{c_1, c_2, c_3\}$ and the same profile P as in the example 3.2.3 previous: P1: (1 1 0); P2: (1 0 0); P3: (0 1 1); P4: (1 1 1) P5: (0 0 1) Similarly, voter 2 approves only the candidate c_1 while voter 4 approves all three candidates Note that dichotomous preferences can be viewed as a special case of incomplete ordinal preferences with two equivalence classes, or as a special case of cardinal preferences with a two-level scoring scale (0 and 1). This is the example from which we have drawn a great deal of inspiration in our work.

2.2. Description of the VMAVA Method

This description is inspired by the research [1] The VMAVA method is a method based on the approval vote and the arithmetic mean. Its principle is as follows: We denote by C a set of m candidates for an election and V a set of n voters. Each of the voters uses parts of C, denoted P (C) which are disjoint and whose union gives C according to the following order: 1st choice, 2nd choice, 3rd choice, 4th choice. For example, each element of each of these subsets is assigned the score of 4, 3, 2, 1 respectively. The method is structured according to the following steps:

The steps of the VMAVA method

Step 1: We first calculate the arithmetic mean of the points awarded by each voter to the candidates he approves.

Step 2: The arithmetic mean of the points awarded to a candidate by all the voters is also calculated.

Step 3: We define three classes of candidates as follows: The class of candidates with a score higher than the average of the voter who approved them This class is noted GSup.[1]

1 GSup means upper group

Thus we build an approval profile for each voter The class noted GMed[2] which is made up of candidates with a score exactly equal to the average of the voter who approved them. The class noted GInf[3] which is composed of the candidates having a score lower than the average of the voter who approved them

Step 4: We retain the candidates who are in first class for each voter.

Step 5: Then we make the intersection of all the profiles of the voters. If only one candidate is obtained, he is the winner, if several candidates are obtained, their arithmetic averages are used and the one with the best average is selected. Considering n voters, if n is even, then the candidate who belongs to at least $\left(\frac{n}{2}+1\right)^{th}$ GSup is the winner. If n is odd, then we consider the integer part of $\frac{n}{2}$ In the case where all the GSups of the voters are empty (or the intersection of all the GSups is empty), the procedure is repeated with the subsets GMed. The candidates of the candidate GMed subsets are only moderately appreciated by the voters. If after all that, there are no winning candidates, it is better to review the elections otherwise use the GInf subsets, and in this case the winning candidate is the one who is not appreciated by the majority or who does not is appreciated by anyone.

2.3. Presentation of the Applications of the VMAVA Method

The examples listed in this section are taken from [1]. They are used to test our program.

Some examples

a) Example 1 of 4 Candidates and 5 Voters

Table 1. *Matrix of Choices.*

$V_1 =$	1st choice $\{c_2, c_4\}$	2nd choice $\{c_1\}$	3rd choice $\{c_3\}$	4th choice
$V_2 =$	1st choice $\{c_1\}$	2nd choice $\{c_3\}$	3rd choice $\{c_4\}$	4th choice $\{c_2\}$
$V_3 =$	1st choice $\{c_3\}$	2nd choice $\{c_1\}$	3rd choice $\{c_2, c_4\}$	4th choice
$V_4 =$	1st choice $\{c_1\}$	2nd choice $\{c_1\}$	3rd choice $\{c_2, c_4\}$	4th choice $\{c_3\}$
$V_5 =$	1st choice $\{c_1\}$	2nd choice $\{c_3\}$	3rd choice	4th choice $\{c_2, c_4\}$

Which corresponds to the following voting matrix

Table 2. *Score Matrix.*

	c_1	c_2	c_3	c_4	means
V_1	3	4	2	4	3, 25
V_2	4	1	3	2	2, 5
V_3	3	2	4	2	2, 75
V_4	3	2	1	2	2
V_5	4	1	3	1	2, 25

For $V_1 : GSup = \{c_2, c_4\} \quad GMed = \{\ \} \quad GInf = \{c_1, c_3\}.$

For $V_2 : GSup = \{c_1, c_3\} \quad GMed = \{\ \} \quad GInf = \{c_2, c_4\}.$

2 GMed means middle group

3 GInf means lower group

For $V_3 : GSup = \{c_1, c_3\}$ $GMed = \{\ \}$ $GInf = \{c_2, c_4\}$.

For $V_4 : GSup = \{c_1\}$ $GMed = \{c_2, c_4\}$ $GInf = \{c_3\}$.

For $V_5 : GSup = \{c_1, c_3\}$ $GMed = \{\ \}$ $GInf = \{c_2, c_4\}$

the winner is c_1

b) Example 2 of 4 Candidates and 4 Voters

Table 3. *Matrix of Choices.*

$V_1 =$	1st choice $\{c_1\}$	2nd choice $\{c_4\}$	3rd choice$\{c_3\}$	4th choice $\{c_2\}$
$V_2 =$	1st choice$\{c_2\}$	2nd choice$\{c_4\}$	3rd choice$\{c_3\}$	4th choice $\{c_1\}$
$V_3 =$	1st choice$\{c_3\}$	2nd choice$\{c_2\}$	3rd choice$\{c_1\}$	4th choice $\{c_4\}$
$V_4 =$	1st choice	2^{nd} choice$\{c_1, c_2, c_4\}$	3rd choice	4th choice $\{c_3\}$

Which corresponds to the following voting matrix:

Table 4. *Score Matrix.*

	c_1	c_2	c_3	c_4	**means**
V_1	4	1	2	3	2, 5
V_2	1	4	2	3	2, 5
V_3	2	3	4	1	2, 5
V_4	3	3	1	3	2, 5

For $V_1 : GSup = \{c_1, c_4\}$ $GMed = \{\ \}$ $GInf = \{c_2, c_3\}$.

For $V_2 : GSup = \{c_2, c_4\}$ $GMed = \{\ \}$ $GInf = \{c_1, c_3\}$.

For $V_3 : GSup = \{c_2, c_3\}$ $GMed = \{\ \}$ $GInf = \{c_1, c_4\}$.

For $V_4 : GSup = \{c_1, c_2, c_4\}$ $GMed = \{\ \}$ $GInf = \{c_3\}$.

The winning candidates are c_2 and c_4 who are tied so we calculate the arithmetic mean of the marks obtained by each of the two candidates; c_2 has an average of $\frac{11}{4}$ and c_4 has an average of $\frac{10}{4}$ ultimately c_2 is the winner.

c) Example 3 of 5 Candidates and 7 Voters

Table 5. *Matrix of Choices.*

$V_1 =$	1^{st}choice $\{c_2, c_5\}$	2nd choice $\{c_4\}$	3rd choice $\{c_1, c_3\}$	4th choice
$V_2 =$	1st choice $\{c_5\}$	2nd choice $\{c_4\}$	3rd choice	4th choice $\{c_1, c_2, c_3\}$
$V_3 =$	1st choice	2nd choice $\{c_4\}$	3rd choice $\{c_1, c_3\}$	4th choice $\{c_2, c_5\}$
$V_4 =$	1st choice $\{c_2\}$	2nd choice $\{c_4, c_5\}$	3rd choice	4th choice $\{c_1, c_3\}$
$V_5 =$	1st choice $\{c_3\}$	2nd choice $\{c_1, c_2, c_3, c_4\}$	3rd choice	4th choice
$V_6 =$	1st choice $\{c_5\}$	2nd choice $\{c_1, c_2, c_3, c_4\}$	3rd choice	4th choice
$V_7 =$	1st choice $\{c_1\}$	2nd choice $\{c_3, c_4\}$	3rd choice	4th choice $\{c_2, c_5\}$

Which corresponds to the following voting matrix:

Table 6. *Score Matrix.*

	V_1	V_2	V_3	V_4	V_5	V_6	V_7
c_1	2	1	2	1	3	3	4
c_2	4	1	1	4	3	3	1
c_3	2	1	2	1	4	3	3
c_4	3	3	3	3	3	3	3
c_5	4	4	1	3	3	4	1
Mean	3	2	1, 8	2, 4	3, 2	3, 2	2, 4

For $V_1 : GSup = \{c_2, c_5\}$ $GMed = \{c_4\}$ $GInf = \{c_1, c_3\}$.

For $V_2 : GSup = \{c_4, c_5\}$ $GMed = \{\ \}$ $GInf = \{c_1, c_2, c_3\}$.

For $V_3 : GSup = \{c_1, c_3, c_4\}$ $GMed = \{\ \}$ $GInf = \{c_2, c_5\}$.

For $V_4 : GSup = \{c_2, c_4, c_5\}$ $GMed = \{\ \}$ $GInf = \{c_1, c_3\}$.

For $V_5 : GSup = \{c_3\}$ $GMed = \{\ \}$ $GInf = \{c_1, c_2, c_4, c_5\}$.

For $V_6 : GSup = \{c_5\}$ $\text{GMed} = \{\ \}$ $GInf = \{c_1, c_2, c_3, c_4\}$.

For $V_7 : GSup = \{c_1, c_3, c_4\}$ GMed $= \{ \}$ $GInf = \{c_2, c_5\}$.

The winning candidates are c_4 and c_5 who are tied so we calculate the arithmetic mean of the marks obtained by each of the two candidates; c_4 has an average of $\frac{21}{7}$ and c_5 has an average $\frac{20}{7}$, ultimately c_4 is the winner. Implementation of the VMAVA Method.

3. Principle of Implementation

Denote by $m \geq 2$ and $n \geq 2$, respectively, the number of candidates and the number of voters. We adopt the following notations.

$C_m = \{C_j, 1 \leq j \leq m\}$, all the candidates for the election.

$V_n = \{V_i, 1 \leq i \leq n\}$, all the voters for the election.

Let us also denote by $T(n, m)$ the candidate score matrix.

$T(i, j)$ means the approval rating given to the applicant j by the voter i. This is the element at the intersection of row i and column j of the matrix $T(i, j)$. The approval matrix of all applicants can be presented as follows:

$$T(i, j) = \begin{pmatrix} & C_1 & C_2 & C_3 & \dots & \dots & \dots & C_j & \dots & C_m \\ V_1 & T_{11} & T_{12} & T_{13} & \dots & \dots & \dots & T_{1j} & \dots & T_{1m} \\ V_2 & T_{21} & T_{22} & T_{23} & \dots & \dots & \dots & T_{2j} & \dots & T_{2m} \\ V_3 & T_{31} & T_{32} & T_{33} & \dots & \dots & \dots & T_{3j} & \dots & T_{3m} \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ V_i & T_{i1} & \dots & \dots & \dots & \dots & \dots & T_{ij} & \dots & T_{im} \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ V_n & T_{n1} & \dots & \dots & \dots & \dots & \dots & T_{nj} & \dots & T_{nm} \end{pmatrix}.$$

Furthermore, we could define the approval matrix $T(i, j)$ by its transpose ${}^tT(i, j)$. A candidate c_i is preferred to a candidate c_k if and only if the number of voters having ranked c_i ahead of c_k is strictly greater than the number of voters having ranked c_k ahead of c_i (in the event of a tie the two candidates are considered indifferent) [9].

If $MD(i, j) > 0$, then $GSup(i, j) = 1$ which means that the candidate j belongs to ith $GSup$. Otherwise $GSup(i, j) = 0$ and the candidate j does not belong to ith $GSup$.

If $MD\ (i, j) = 0$, then Gmed (i, j) = 1 which means that the candidate j belongs to ith *GMed*. Otherwise $GSup(i, j) = 0$ and the candidate j does not belong to the ith *GMed*.

If $MD(i, j) < 0$, then $GInf(i, j) = 1$ which means that the candidate j belongs to ith *GInf*. Otherwise $GInf(i, j) = 0$ and the candidate j does not belong to the ith *GInf*. Each column of *GSup*, *GMed* or *GInf* can be called an approval vector. The sum of the elements of this vector gives us what we will also call the total number of approvals of a candidate that we note *N*. We have observed that the candidate receiving the maximum number of approvals that we note (A*N*) also belongs to the median *GSup* and therefore is the winner. Two candidates totaling the same number of maximum approvals are tied. To facilitate the choice of the winning candidate we totaled the number of approval of each candidate by summing the columns of $GSup(i, j)$. Thus, we obtain an approval vector *Us* such as $\max(\max(Us)) <= n$ (number of voters). This vector *Us* concatenated with the vector of the averages *MC* of the candidates constitutes a matrix noted *Gs*. If the winning candidate is not found in the *GSup*, the method is repeated in the *GMed* and *GInf*. In this case, the matrices *Gm* and *GI* are thus formed, which are synthesis matrices constructed in the same way as *Gs*. The *Gs* matrices, *Gm* and *GI* are synthesis matrices that allow us to give the winning candidate.

$Gs = [Us, MC]$ with Us, also a vector = sum of columns of GSup(i, j) = 1

$Gm = [Um, MC]$ with Um, also a vector = sum of the columns of

$GI = [UI, MC]$ with UI also a vector = sum of the columns of $GInf(i, j)$.

3.1. A Bit of Algorithmic

function G = Election()
inputs:
$T(n, m)$: matrix of natural numbers
n, m, i, j: whole
$i = 1$ to n; $j = 1$ to m
outputs:
$MC(1, m)$:\\C̅andidates Averages (row vector)
$MV(n, 1)$:\\V̅oters Averages (column vector)
$M = \text{repmat}(MV, 1, m)$; the replicated matrix of $MV(n, 1)$, of size $[n, m]$
$D = A - M$; % D is The matrix of comparisons by pair (candidate, voter)
GSup, *GMoy*, *GInf*: % the different classes of candidates, are binary matrices of the same dimensions as the matrix $T(n, m)$ [10]
$G =$ elected candidate, a singleton note that

$$GSup(i, j) = \begin{cases} 1 & \text{if the candidate } j \text{ is approved acute by the voter } i \\ 0 & \text{if the voter } i \text{ does not approve the candidate } j. \end{cases}$$

The same is true for $GMed(i, j)$ and $GInf(i, j)$.

3.2. Matlab Program

Le programme suivant est inspire de [6-8].

```
function = FonctElection()
A = input('Enter the voting matrix =');
[n, m] = size(A);
n = size(A, 1); m = size(A, 2);
k = 1 : n; l = 1 : m;
MV = sum(A(k, :), 2)/m; % average voters
MC = sum(A(:, 1), 1)/n; % candidate average;
disp('L" set of voter means is');
MV;
disp('L" set of candidate means is');
MC;
M = repmat(MV, 1, m);
D = A - M; % D comparison matrix
disp('The comparison matrix is'); D;
% Calcul des des matrices d'approbation des classes GSup; GMed et GInf
disp('The candidate classes are as follows:');
GSup = MD > 0;
GMoy = MD == 0;
GInf = MD < 0;
Gs = sum(GSup, 1);
Gm = [sum(GMoy, 1); MC];
GI = [sum(GInf, 1); MC];
Gs = [sum(GSup, 1); MC];
g = find(Gs(1,:)>=y0); % All candidates eligible by approval
G = find(Gs(2,:)>=x0); % All the candidates definitively elected
r=length(G); % reflects the uniqueness of the elected candidate
y0 = max(max(Gs(1, :))); % maximum approving voters in GSup
y1 = max(max(Gs(1, :))); % maximum approving voters in GMed
y2 = max(max(Gs(1, :))); % maximum approving voters in GInf
x0 = max(max(Gs(2, :))); % maximum of the arithmetic average of the candidates g = find(Gs(1, :) >= y0);
G = find(Gs(2, :) >= x0); % number of candidates to be elected if ( y0 ≠ 0 and r == 1)
    g = find(Gs(1, :) >= y0);
    G = find(Gs(2, :) >= x0);
    g = G;
sprintf('the elected candidate is the number %d.', G)
elseif (y1 ≠ 0 and r == 1)
    g = find(Gm(1, :) >= y1);
    G = find(Gs(2, :) >= x0 and Gs(1, :) >= y1);
    g = G;
    sprintf('the elected candidate is the number %d.', G)
elseif ( y2 ≠ 0 and r == 1)
    g = find(GInf(1, :) >= y2);
    G = find(Gs(2, :) >= x0 and Gs(1, :) >= y2);
    g = G;
    sprintf('the elected candidate is the number %d.', G)
else
    sprintf('the elections must be reconsidered')
    end
end
end
```

3.3. Applications and Results

a) Example 1: Example of 4 Candidates and 5 voters

Table 7. *Score Matrix.*

	c_1	c_2	c_3	c_4
V_1	3	4	2	4
V_2	4	1	3	2
V_3	3	2	4	2
V_4	3	2	1	2
V_5	4	1	3	1

By entering the matrix of votes by the syntax: matrix = xlsread ('C:\Users\USER\Desktop\my project\data1.xls'), we have the following results:

Voters averages

Table 8. *Averages of voters and candidates.*

V_1	3.2500					
V_2	2.5000					
V_3	2.7500	Candidates:	c_1	c_2	c_3	c_4
V_4	2.0000	MC	3.4000	2.0000	2.6000	2.2000
V_5	2.2500					

Table 9. *Matrix of comparisons.*

		c_1	c_2	c_3	c_4
	V_1	−0.2500	0.7500	−1.2500	0.7500
	V_2	1.5000	−1.5000	0.5000	−0.5000
$MD =$	V_3	0.2500	−0.7500	1.2500	−0.7500
	V_4	1.0000	0	−1.0000	0
	V_5	1.7500	−1.2500	0.7500	−1.2500

The different classes of candidates

Table 10. *Matrix of approval in the upper class.*

		c_1	c_2	c_3	c_4
	V_1	0	1	0	1
	V_2	1	0	1	0
GSup =	V_3	1	0	1	0
	V_4	1	0	0	0
	V_5	1	0	1	0

Table 11. *Matrix of approval in the middle class.*

		c_1	c_2	c_3	c_4
	V_1	0	0	0	0
	V_2	0	0	0	0
GMoy =	V_3	0	0	0	0
	V_4	0	1	0	1
	V_5	0	0	0	0

Table 12. *Approval matrices in the lower class.*

$GInf$ =		c_1	c_2	c_3	c_4
	V_1	1	0	1	0
	V_2	0	1	0	1
	V_3	0	1	0	1
	V_4	0	0	1	0
	V_5	0	1	0	1

Table 13. *Upper Class Results Matrix.*

		c_1	c_2	c_3	c_4
Gs =	*AN*	4	1	3	1
	MC	3.4	2	2	2.2

Table 14. *Matrix of results in the lower class.*

		c_1	c_2	c_3	c_4
Gm =	*AN*	0	1	0	1
	MC	3.4	2	2	2.2

Table 15. *Matrix of results in the lower class.*

		c_1	c_2	c_3	c_4
GI =	*AN*	1	3	2	3
	MC	3.4	2	2	2.2

Based on the approval totals in *GSup*, candidate c_1 is elected.

b) Example 2: Example of 4 Candidates and 4 voters

Table 16. *Score matrix.*

	c_1	c_2	c_3	c_4
$V_5 = V_1$	4	1	2	3
V_2	1	4	2	3
V_3	2	3	4	1
V_4	3	3	1	3

Executing the previous MALTAB program gives the following results:

Voters: Averages

Table 17. *Arithmetic averages of voters and candidates.*

V_1	2.5000					
V_2	2.5000	Candidates:	c_1	c_2	c_3	c_4
V_3	2.5000	MC:	2.5000	2.7500	2.2500	2.5000
V_4	2.5000					

The matrix of comparisons is:

Table 18. *Matrix of comparisons.*

		c_1	c_2	c_3	c_4
	V_1	1.5000	−1.5000	−0.5000	0.5000
MD =	V_2	−1.5000	1.5000	−0.5000	0.5000
	V_3	−0.5000	0.5000	1.5000	−1.5000
	V_4	0.5000	0.5000	−1.5000	0.5000

The different classes of candidates:

Table 19. *Approval matrices in classes GSup, GMed and Ginf.*

		c_1	c_2	c_3	c_4
	V_1	1	0	0	1
GSup =	V_2	0	1	0	1
	V_3	0	1	1	0
	V_4	1	1	0	1
		c_1	c_2	c_3	c_4
	V_1	0	0	0	0
GMed =	V_2	0	0	0	0
	V_3	0	0	0	0
	V_4	0	0	0	0
GInf =		c_1	c_2	c_3	c_4

	V_1	0	1	1	0
	V_2	1	0	1	0
	V_3	1	0	0	1
	V_4	0	0	1	0
		Results matrices			
	Candidates:	c_1	c_2	c_3	c_4
Gs =	*AN*	2.0000	3.0000	1.0000	3.0000
	MC	2.5000	2.7500	2.2500	2.5000
		c_1	c_2	c_3	c_4
Gm =	*AN*	0.000	0.000	0.000	0.000
	MC	2.5000	2.7500	2.2500	2.5000
		c_1	c_2	c_3	c_4
GInf =	*AN*	2.0000	1.0000	3.0000	1.0000
	MC	2.5000	2.7500	2.2500	2.5000

Based on the approval totals in *GSup*, candidates c_2 and c_4 are tied. But in view of their arithmetic means, c_2 is the elected candidate because he has the highest average.

c) Example 3: Example of 5 Candidates and 7 Voters

Table 20. *Score matrices.*

	V_1	V_2	V_3	V_4	V_5	V_6	V_7
c_1	2	1	2	1	3	3	4
c_2	4	1	1	4	3	3	1
c_3	2	1	2	1	4	3	3
c_4	3	3	3	3	3	3	3
c_5	4	4	1	3	3	4	1
Averages	3	2	1, 8	2, 4	3, 2	3, 2	2, 4

Table 21. *Voter and candidate averages.*

V_1 : 3.0000						
V_2 : 2.0000	Candidates:	c_1	c_2	c_3	c_4	c_5
V_3 : 1.8000	MC	2.2857	2.4286	2.2857	3.0000	2.8571
V_4 : 2.4000						
V_5 : 2.4000						
V_6 : 3.2000						
V7: 2.4000						

Table 22. *Matrix of comparisons.*

		c_1	c_2	c_3	c_4	c_5
	V_1	−1.0000	1.0000	−1.0000	0	1.0000
	V_2	−1.0000	−1.0000	−1.0000	1.0000	2.0000
D =	V_3	0.2000	−0.8000	0.2000	1.2000	−0.8000
	V_4	−1.4000	1.6000	−1.4000	0.6000	0.6000
	V_5	−0.2000	−0.2000	0.8000	−0.2000	−0.2000
	V_6	−0.2000	−0.2000	−0.2000	−0.2000	0.8000
	V_7	1.6000	−1.4000	0.6000	0.6000	−1.4000

The different classes of candidates

Table 23. *Upper Class Approval Matrix.*

	c_1	c_2	c_3	c_4	c_5
V_1	0	1	0	0	1

		c_1	c_2	c_3	c_4	c_5
	V_2	0	0	0	1	1
GSup =	V_3	1	0	1	1	0
	V_4	0	1	0	1	1
	V_5	0	0	1	0	0
	V_6	0	0	0	0	1
	V_7	1	0	1	1	0

Table 24. *Matrix of approval in the middle class.*

		c_1	c_2	c_3	c_4	c_5
	V_1	0	0	0	1	0
	V_2	0	0	0	0	0
	V_3	0	0	0	0	0
GMed =	V_4	0	0	0	0	0
	V_5	0	0	0	0	0
	V_6	0	0	0	0	0
	V_7	0	0	0	0	0

Table 25. *Matrix of approval in the lower class.*

		c_1	c_2	c_3	c_4	c_5
	V_1	1	0	1	0	0
	V_2	1	1	1	0	0
	V_3	0	1	0	0	1
GInf =	V_4	1	0	1	0	0
	V_5	1	1	0	1	1
	V_6	1	1	1	1	0
	V_7	0	1	0	0	1

Table 26. *Upper Class Results Matrix.*

		c_1	c_2	c_3	c_4	c_5
Gs =	*AN*	2.0000	2.0000	3.0000	4.0000	4.0000
	MC	2.2857	2.4286	2.2857	3.0000	2.8571

Table 27. *Middle Class Outcomes Matrix.*

		c_1	c_2	c_3	c_4	c_5
Gm =	*AN*	0	0	0	1.0000	0
	MC	2.2857	2.4286	2.2857	3.0000	2.8571

Table 28. *Matrix of results in the lower class.*

		c_1	c_2	c_3	c_4	c_5
GI =	*AN*	5.0000	5.0000	4.0000	2.0000	3.0000
	MC	2.2857	2.4286	2.2857	3.0000	2.8571

Based on approval totals in *GSup*, applicants c_4 and c_5 are tied. But in view of their arithmetic means, c_4 is the elected candidate because he has the highest average.

4. Conclusion

Traditionally, voting rules are designed to aggregate preferences over small sets of candidates, or alternatives. However, some situations involve a significantly large number of alternatives for this to become an important issue [5]. Manually applying the VMAVA method to a large voting problem is tedious because of the number of comparison operations it requires. To generate the sets *GSup*, *GMed* and *GInf*, we used binary matrices to circumvent the difficulty of generating matrices of indexed character strings as described in the literature. The other difficulty encountered is the intersection of the *GSups* because each *GSup* is a binary line vector and therefore the intersection is most often empty. This does not allow us to obtain the winning candidate(s). Similarly, the extraction of the median *GSup* according to the parity of the number of voters gives us ties in certain cases where the winning candidate is nevertheless unique. So, to overcome these difficulties, we have built with the MATLAB software a function that calculates the total number of approvals for each candidate and another that considers the maximum of the maximums of this sum to display the elected candidate (s). In the event of a tie, the maximum of the arithmetic average of the candidates is used for the tie. Despite this, a rather rare but not insignificant situation was encountered. It is the one where two tied candidates have the same arithmetic mean. In this case there are no results, hence the need to review the elections in the best possible case.

Despite these obstacles, we have found quite interesting results with regard to what exists in the literature relative to the VMAVA method, which results remain to be perfected in our future work and this is why we think that these obstacles as well as an algorithmic study of the complexity of the VMAVA method could be the subject of perspectives in our future research projects reflection.

References

[1] Zoïnabo Savadogo and Blaise Somé. Voting method based on approval voting and arithmetic mean; International Journal of Applied Mathematical Research, 9 (1) (2020) 1-8.

[2] Antoinette Baujard, Herrade Igersheim, Experimentation of vote by note and vote by approval during the French presidential election of April 22; HAL Id: halshs-00337290, https://shs.hal.science/halshs-00337290 Submitted on 23 Sep 2010.

[3] Michel BALINSKI, Rida LARAKI, Jugement Majoritaire Vs. Vote Majoritaire (via les Présidentielles 2011-2012).

[4] Adama COULIBALY, Group decision, Facilitation assistance: Adjustment of voting procedures depending on the context, doctoral thesis at the university of sciences, techniques and technologies of BAMAKO (USTTB).

[5] Nathanaël Barrot; On the computational aspects of voting by approval, Thesis defended on March 31, 2016 at Paris-Dauphine University.

[6] The MathWorks, Inc 24 Prime Park Way; Mail: http://www.mathworks.com Natick, MA 01760-1500.

[7] F. Delebecque and J. C. Pesquet, MATLAB: Getting the software hand, https://www.math.u-bordeaux.fr/~ayger/MATLABSignal/Tutorials Quentin.

[8] Glorieux, Mathematical tools and use of Matlab Course 2013-2014, Pierre and Marie Curie University - Paris VI.

[9] D. Bouyssou (CNRS); Th. Marchant (Universiteit Gent); P. Perny (LIP6); Social choice theory and multi-criteria decision support, May 2002 – revised October 13, 2005.

[10] François Durand, Towards less manipulable voting methods, Thesis defended on September 24, 2015, PIERRE AND MARIE CURIE UNIVERSITY; HAL Id: tel-03654945 https://hal.inria.fr/tel-03654945 Submitted on 29 Apr 2022.

[11] Robert J. Weber; Approval Voting, The journal of economics perspectives, Vol. 9, No 1 (Winter, 1995), 39.

[12] Patrick BLANCHENAY, Paradoxes de vote et modes de scrutin en France.

[13] MAURICE SALLES, la théorie du choix social: une introduction a quelques résultats fondamentaux.

[14] Nathanaël BARROT, Sur les aspects computationnels du vote par approbation, Université Paris-Dauphine.

[15] Catherine Petillon, Si on votait autrement 25. 04. 2017.