Automation of counting of the Preferential votes

Rakesh Kumar Pandey
Department of Physics
Kirori Mal College, University of Delhi
Delhi - 110007, India
Email: r.kr.pandey@gmail.com

***Abstract* — This paper discusses the complexity involved in counting of preferential votes wherein the candidates are ranked by the voters. The process of counting passes through the rounds of distribution of surplus votes and votes of the eliminated candidates. It is shown theoretically that the counting has exponential time complexity. Thankfully, the manual counting does not require exploring all the possibilities. For such problems where automating the process theoretically involves complex iterations but relatively has easier manual solutions, an easier way will be to mix up manual and digital process in such a manner that the complete exercise gets tackled as efficiently as possible.**

Keywords—preferential voting; automation of counting of votes; time complexity of the counting algorithm; single transferable vote

I. INTRODUCTION

Preferential voting is a voting procedure in which the voters are required to rank her choices. These votes are then counted in a peculiar way to identify the winning choice(s). Out of the several methods, there are two ways that are widely acceptable for counting of such votes. One is known as the Instant Run-off Voting (IRV) method and the other is termed as the Single Transferable vote (STV) method [1] [5]. In IRV method, the candidates that get the lowest votes are successively eliminated and their votes are distributed to other candidates that were given the next rank. However, in the Single Transferable Votes (STV) process, surplus votes of the winning candidates are first distributed till one is left with only those candidates who have less than the required number of votes to be declared elected [3], [4]. When such a deadlock is encountered then the candidate having the lowest votes is eliminated and their votes are distributed among the choices that are given next lower rank. By executing the distribution of surplus votes and the strategy of elimination successively, the winners are identified in the end.

This idea to work towards finding solution to this complex problem came after observing the election that was carried out in Delhi University [2] for the membership of the Academic Council and the Executive Council of 2017 using preferential voting with Single Transferable Vote system of counting. The counting of votes normally takes two to three days. In this age of automation it is desirable to automate the entire procedure of such a complex voting and counting process. The automation will not only save valuable time of almost hundreds of teaching professionals and other administrative officials of the University but it would also ensure a mistake free counting [6], [11].

II. UNDERSTANDING THE PROCESS

The process of preferential voting system involves two stages. Firstly when the votes are casted and then when the votes are counted.

A. *Casting of Votes*

This process needs to be automated first so that the counting of votes can be automated. It can be done in two ways. It can be done either by designing an EVM like machine [7], [8], [9], [10] that can store ranks that a voter is asked to assign to different candidates or by designing an Optically Readable Document wherein the ranks indicated by pen can be later read on Optical Mark reading machine. This would help in avoiding one by one entry of the votes and ensure an error-free database building exercise.

B. *Counting of votes*

Once the ranks assigned by different voters are uploaded in the database, a strategy to automate the counting process can be attempted. This paper primarily focuses on automation of the process of counting that is done using the Single Transferable Vote (STV) method [7], [8], [9] [10].

III. ANALYSIS OF THE COUNTING PROCESS

Let us consider that there are N number of voters, M number of candidates and P number of posts ($where\ P \leq M$) for which the election is to be held. In the worst case, if all the voters exercise all their options upto the rank P for the candidates then there will be NxP number of entries to be done in the computer database. The nature of counting in such a system is such that one would continuously need answers to the following types of queries. How many votes did $i\ th$ candidate get as their second option, $j\ th$ candidate got the first, $l\ th$ got the second, $k\ th$ got the third and so on. Let us denote this by the following notation:

$$W(j, l, k, n \dots . p, i)$$

Here i, j, k, l .... identify the candidates and the sequence in which they appear in the argument is decided by the ranks given to that particular candidate. The above notation denotes the number of votes that the candidate i gets from those ballots where 'j' got the first rank, 'l' got the second, 'k' got third, 'n' got fourth ...... and 'p' got the 's' th rank (such that $s \leq P - 1$).

After the first round of counting one would require the counts given by $W(i)$ for all the M candidates. At this stage all the candidates get identified who are able to cross the threshold determined by the formula,

$$T = \frac{N}{(P+1)} + 1 \tag{1}$$

Surplus of their votes are then distributed among the next rank in the ballots of those winning candidates. For this, the following values will be needed

$$W(k, i)$$

for all the candidates 'i' who are yet to cross the value estimated by 'T'. Here 'k' identifies the candidates who have crossed the value 'T'. Value of these surplus votes are determined by the following formula

$$(W(k) - T)/\sum_i W(k, i) \tag{2}$$

After distributing all the surplus votes if the number of candidates who have crossed the value T is still less than P then the candidate having the minimum votes is eliminated. On elimination, the votes of eliminated candidates are distributed with full value among the next rank holders. For this again we would require the values indicated by the following notation.

$$W(k, i)$$

for the candidate k who got eliminated. If some of the candidates cross the value T in this process then the surplus votes are again distributed in the same manner as given above by counting the numbers given by

$$W(k, j, i)$$

where k is the candidate who have been either eliminated or crossed the value T. In case a data structure is built up to keep all possible permutations of W's arguments, then it would require huge number of entries as estimated in the following calculations.

A. *Counting of all $W(k, j, \ldots, i)$ required for counting*

If there are r number arguments in the 'W' notation it would have

$$M(M-1)(M-2)\ldots.(M-r+1) = \binom{M}{r} \tag{3}$$

combinations. And as we can have number of arguments varying from 1 to P, we can estimate the total possibilities by the summation,

$$\sum_{r=1}^{r=P}\binom{M}{r} \approx 2^M \text{ (considering } M \approx P) \tag{4}$$

To evaluate all these possibilities, the algorithm will certainly have a time complexity of the order of $O(2^M)$. For example, in the Academic Council elections of Delhi University, typically 35 candidates contest for 26 posts and this would make the time complexity quite high.

B. *Strategy for counting*

- Although the numbers involved in such estimates threaten to be huge but one must be encouraged to notice the fact that such a counting over the years is completed in two-three days. This is because of the fact that during the counting (that is still done manually), one never requires to take all the possibilities into account. The number of rounds always has an upper limited decided by the difference factor = $M - P$, since we either need to eliminate or to make them cross the threshold value T only by some countable number of times so that we are left with P candidates at the end.
- One may recommend appropriate use of Excel sheets to automate such a complicated counting process once the data is imported on the same. This would prescribe a strategy that would mix the manual effort with that of the Excel Sheet computation. One can keep on determining the required W values by selecting and finding a particular pattern out of the ranking-pattern chosen due to the choice of candidates made by the voters by ranking the candidates.
- Easiest way to prepare for all possible types of information so that one may be able to get the W values by utilizing simple SQL type of queries, will be to create strings of the candidates for each ballot by placing them as per their ranking. For example, one should make a string 'BHDE' for a ballot that indicates that the voter had chosen candidate 'B' as its first, 'H' as his second, 'D' as his third and 'E' as his fourth choice.
- Counting the occurrence of a pattern such as 'APO' can then readily extract the desirable information by properly using the match function.

C. *Illustration*

To show the process explicitly a table (Table 1: Excel Sheet) is shown below that has already imported the information of 40 ballots wherein voters have ranked a maximum of six candidates out of the sixteen candidates contesting for an election. Candidates are marked from 'A' to 'P' and are ranked as 1 to 5.

The last column displays the string suggested in this paper by concatenating all the choices rank-wise placing the first ranker first and so on.

Table 1. Maximum up to six Ranks given to 16 candidates contesting for six posts by 40 voters.

	rakesh	anil	surender	pradumn	rishi	riyaz	Shashi	Suman	rasal	asha	Jitender	sulochana	albert	brijmohan	ashok	Chtragupt	Rank-string
	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	
1			1	5			2		4			3				6	CGLIDP
2		1						4			3		5		2		BOKHM0
3	1				3		4					5		2			ANEGL0
4		1		2				3		5						4	BDHPJ0
5	4		5			3			2				1				MIFAC0
6		1		3				5			2					4	BKDPH0
7	5	4	3	2	1					6							EDCBAJ
8		1				5			4				3			2	BPMIF0
9	1				3			2			4				5		AHEKO0
10	2	1		3													BAD000
11				1			2			4				3			DGNJ00
12	4		3			1			6				2			5	FMCAPI
13			1				2		4			3				6	CGLI0P
14		1						4			3		5		2		BOKHM0
15	1				3		4					5		2			ANEGL0
16		1		2				3		5						4	BDHPJ0
17	4		5		6	3			2				1				MIFACE
18		4		3				5			2					1	PKDBH0
19	5	1	3	2	4												BDCEA0
20		1		6		5			4				3			2	BPMIFD
21	1				3			2			4				5		AHEKO0
22		1															B00000
23				1			2			4				3			DGNJ00
24	4		3			1							2			5	FMCAP0
25			1				2		4			3				6	CGLI0P
26		5						4			3		1		2		MOKHB0
27	1				3		4					5		2			ANEGL0
28		5		2				3		1						4	JDHPB0
29	4		5			3			2				1				MIFAC0
30		4		3				5			2					1	PKDBH0
31	5	4	3	2	1							6					EDCBAL
32		1				5			4				3			2	BPMIF0
33	1				3			2			4				5		AHEKO0
34		1															B00000
35				1			2			4				3			DGNJ00
36	4		3			1	6						2			5	FMCAPG
37			1				2		4			3				6	CGLI0P
38		5						4			3		1		2		MOKHB0
39	1				3		4		6			5		2			ANEGLI
40		5		2				3		1						4	JDHPB0

In the Excel Sheet a Formula given by

COUNTIF(R3:R42, "MO*")

gives the value of the W parameter denoted by

$$W(M, O) = \text{COUNTIF(R3: R42, "MO * ")}$$

Using queries like the above, counting officer can get all information required for the counting to progress from round after round. As these figures will be determined by the computer, they will be error-free and can also be done very fast. The process that usually takes two to three days can be expected to be over within two hours.

ACKNOWLEDGMENT

I thank Kirori Mal College for providing me the necessary infrastructure to carry out this work. I hope that this effort would eventually help in automation of the process of counting that has continued to be done mannually for an inexplicably long period of time.

REFERENCES

[1] Wikipedia, "Preferential Voting", https://en.wikipedia.org/wiki/Preferential_voting

[2] Rakesh Kumar Pandey, 2017, 'Bound to Ballot Boxes, How Delhi University refuses to modernise itself',

https://www.youthkiawaaz.com/2017/02/problem-of-counting-in-the-preferential-voting/

[3] The Victorian Electoral System, An illustration for the STV method of counting of preferential votes.
https://www.vec.vic.gov.au/files/activitysheet19.pdf

[4] Electoral Commission, Newzealand, "How Preferential Voting works",
http://www.elections.org.nz/events/referendums-new-zealand-flag-0/voting-first-referendum/how-preferential-voting-works

[5] Donald G Saari, "Basic Geometry of Voting", Springer Science and Business media, 1994 page 244.

[6] Bernhard Beckert, Rajeev Gore and Carsten Schurmann, "International Conference on Automated Deduction CADE 2013", page 135-144.

[7] Adiba S. Shaikh, Ramesh R Marza and Ganesh Manza, "Architecture of Electronic Voting Machine for Preferential voting used in University elections", July 2012, issue Volume 1, issue 5.

[8] Shaikh Adiba, Dr. Ramesh R. Manza, "The Calibration and Voting Process of Preferential Electronic Voting Machine using Microcontroller", International Journal of Innovative Research in Computer and Communication Engineering, Vol. 3, Issue 7, July 2015, DOI: 10.15680/ijircce.2015. 0307057 6609.

[9] Akash Konnur, Archana Nandibewoor, Nita Kakhandaki, "Preferential Electronic Voting Machine - An Embedded Computing Solution", International Journal of Emerging Technology and Advanced Engineering (www.ijetae.com (ISSN 2250-2459, ISO 9001:2008), Volume 3, Issue 3, March 2013.

[10] Shaikh Adiba, Dr. Ramesh R. Manza, "Counting and Vote Distribution Mechanism in Preferential Election Using PEVM", Volume 6, issue 5, 2016.

[11] James Heather, Steve Schneider, Vanessa Teague, "E-Voting and Identity: 4th International Conference, Vote-ID 2013", Guildford, UK, July 17-19, 2013, Proceedings, Springer, 08-Jul-2013.