International Journal of Hybrid Information Technology
Vol.8, No.4 (2015), pp.125-132
http://dx.doi.org/10.14257/ijhit.2015.8.4.15

Improved Majority Voting Algorithm in Redundancy System

Jincui Yang[1], Lirong Qiu[2] and Yu Zhang[3]
[1]School of Software Engineering, Beijing University of Posts and Telecommunications, Beijing, China
[2]School of Information Engineering, Minzu University of China, Beijing, China
[3]IT Dept., China Post Life Insurance Company Limited, Beijing, China
jincuiyang@bupt.edu.cn[1], qiu_lirong@126.com[2], zybupt@qq.com[3]

Abstract

Voting algorithm is an important indispensable part of the redundancy design in redundant systems. This paper presented an improved majority voting algorithm, and compared the algorithm with the standard majority voting algorithm and the median voting algorithm through experiments, it is proved that the improved algorithm achieves a compromise between the result selection capabilities of the median voting algorithm and the safety features of the majority voting algorithm.

Keywords: *Voting algorithm; Redundancy design; Safety; Availability*

1. Introduction

Redundant technology has a wide application in aviation, aerospace, nuclear industry. One of the key technologies in redundant system is redundant voting algorithm, there have been a variety of standard voting algorithms, such as median voting algorithm, weighted voting algorithm, majority voting algorithm. The majority voting algorithm is a security voting algorithm, which has been widely applied in various systems. It can output an exception flag to the system in cases of no majority, and inform the system that has not a voted result. This character determines that majority voting algorithm has a lower voting correct rate than median voting algorithm and weighted voting algorithm. Aiming at this problem, this paper proposes an improved majority voting algorithm. It will use history information in the majority voting, reduces the probability of exception flag, and improves the availability of the system.

2. Related Works

Voting algorithm is an important indispensable part of the redundancy design in redundant systems. The research about the redundant voting algorithm has been going on in domestic and international, there are some representative voting algorithms. The related research introduced in the following.

2.1. Standard voting algorithms

At present, there are a variety of standard voting algorithms, The widely used algorithm are the majority voting algorithm, the median voting algorithm and the weighted average voting algorithm etc. [3]. These voting algorithms has its own advantages and disadvantages, for example, when the system has requirement of high availability, the median voting method is the most suitable algorithm; when the system has requirement of higher security, the majority voting algorithm is the most suitable.

Following is a brief introduction of the majority voting algorithm and the median voting algorithm.

ISSN: 1738-9968 IJHIT

The basic idea of the majority voting algorithm is: if more than half of the inputs are the same value, the value can be used as the voting result. Because the inputs cannot guarantee completely consistent, so the majority voting need to determine a threshold value, When the difference of two inputs is less than or equal to the threshold value, we can think of these two values are the same. For example, three inputs are x_1, x_2, x_3, the threshold value is ε, if |x1-x2|≤ε, |x2-x3|≥ε, we think of x1is same as x2, x2is different from x3. A feature of the majority voting algorithm is that it can output an exception flag, to inform system the vote failed, so it can effectively ensure the safety of the system.

The basic idea of the median voting algorithm is: select a mid-value as the voting result from N (here N is odd) inputs. Different from the majority voting algorithm, the median voting algorithm will always has a voting result.

2.2. Three Modified Majority Voting Algorithms

Smoothing majority voting algorithm [1] proposed by G.Latif-Shabgahi *et al.*, based on majority voting algorithm, the algorithm references historical voting data. When the majority voting algorithm cannot vote, the smoothing algorithm selects the value which be close to previous voting result.

The hybrid voting algorithm of majority and median [4] proposed by Zhang Yinan *et al.*, The paper published at the Thirteenth National Academic Conference on Fault-Tolerant Computing. The algorithm combines majority and median voting algorithm, if the majority voting algorithm cannot vote, the algorithm select a mid-value as the result.

The weighted majority voting algorithm [2] is proposed by Zhang Yinan Et al. too. By set a threshold value first, when the majority voting algorithm cannot vote, the threshold value use for the selective use of weighted voting algorithm to vote.

3. Improved Majority Voting Algorithm

When the inputs has more than (N+1) /2 different values, the majority voting algorithm can not to vote, the output is an exception flag which makes the system have interrupted time. Median voting method always can choose the output results, but the results are not necessarily correct, it may bring a very big error, caused serious harm. Based on the standard majority voting algorithm, this section will propose an improved majority voting algorithm which is reliable and high correct rate.

3.1. Algorithm

Because the majority voting algorithm may cause the system interrupt when output an exception flag, the improved algorithm will reduce cannot vote probability, and ensure the correct rate at the same time.

Here, for the convenient use, we first give the following definition:

Definition 1: let t denote t times vote.

Definition 2: let t_c denote the correct times of the voting result in t times vote.

Definition 3: let t_{ci} denote the times that the first I algorithm result is same as the voting result in t times vote.

Definition 4: let ε denote the threshold value of the majority voting algorithm to verify whether the results are the same.

Definition 5: let β denote the probability threshold value, 0≤β≤1.

Definition 6: let p_i denote the probability that the first I algorithm result is same as the voting result, $p_i=t_{ci}/t * 100\%$.

The steps of the improved voting algorithm as following:

Step 1. Let $S=\{x_1,x_2,\ldots,x_n\}$ denote the set of N variant results needed to vote.

Step 2. Sort the set S in ascending order to construct the new set $S_a=\{y_1,y_2,\ldots,y_n\}$.

Step 3. Construct the following partitions from S_a, $S_i=\{y_i,y_{i+1},\ldots,y_{i+j-1}\}$, in which: $1\le i\le j$, $j=(n+1)/2$

Step 4. If at least one of the partitions $S_i(1\le i\le j)$ satisfies the property $d(y_i,y_{i+j-1})\le\varepsilon$, then selecting the output using the majority voting algorithm

Step 5. If none of the partitions $S_i(1\le i\le j)$ satisfies the property $d(y_i,y_{i+j-1})\le\varepsilon$, then select $p_m=\max\{p_1,p_2,\ldots,p_n\}$, in which: $1\le m\le n$

Step 6. If p_m satisfies the property $p_m\ge\beta$, then select x_m as the voter output, If $p_m<\beta$, then the vote failed, output an exception flag

The improved voting algorithm can vote a correct result in the condition of majority. In cases of no majority, it can choose the input of the high accuracy algorithm as voting output. If the historical correct rate of the highest correct rate algorithm is not up to the probability threshold value β, the improved voting algorithm outputs an exception value. So it can prevent the system to choose the wrong results in abnormal situations, to avoid serious consequences for the system.

The threshold value β is extremely important in the algorithm, if β is too small, the vote result may be wrong, the system is not safe. If β is too big, the condition ($p_m\ge\beta$) is difficult to meet. With the increase of β value, the improved voting algorithm tends more and more the standard majority voting algorithm.

3.2. Experiment

In order to elaborate the improved voting algorithm, we gave simulation in two ways.

3.2.1 Voting Process Simulation: We gave the voting process using given data, then we compared the improved voting algorithm results with majority voting algorithm and median voting algorithm are.

We gave seven sets of data in table 1, in which x_1, x_2, x_3 are three inputs, denote the results of three redundant algorithms, p_1, p_2, p_3 denote the probability of the three different algorithms result is the same as the final voting result, the initial value of p_1, p_2, p_3 is 0. Take the ε value is 0.1, when β value is 60%, the voting results as shown in Table 1.

Table 1. The Voting Process of Improved Majority Voting Algorithm

(ε=0.1, β=60%)

	X_1	X_2	X_3	P_1	P_2	P_3	voting results
1	1.2	1.5	1.4	0	0	100%	1.4
2	2.1	2.2	2.4	50%	0	50%	2.1
3	3.2	3.5	3.4	33.3%	0	66.7%	3.4
4	4.8	4.5	4.9	50%	0	50%	4.8
5	5.5	5.2	5.4	40%	0	60%	5.4
6	6.5	6.9	6.3	33.3%	0	66.7%	6.3
7	7.3	7	7.7	28.6%	0	71.4%	7.7

In Table 1, the data of the first set, the second set and the third set meet the conditions of majority voting, the voting results were 1.4, 2.1 and 3.4. In the first three cycles of voting, the times of the voting result same as x_1 is 1, the times of the voting result same as x_3 is 2, so after the first three cycles of vote, p_1=33.3%, p_2=0, p_3= 66.7%. The data of the fourth set also satisfy the conditions of majority voting, the voting result is 4.8, same as x_1. Therefore, after 4 cycles of voting, p_1=2/4*100%=50%, p_2=0/4*100%=0, p_3=2/4*100%=50%. The data of the fifth set also satisfy the conditions of majority voting, the voting result is 5.4, after 5 cycles of voting, p_1=40%, p_2=0, p_3=60%. The sixth sets of data do not satisfy the conditions of majority voting, we select the maximum value from p_1,p_2 and p_3, p_3(=60%) is the maximum value, meet the conditions $p_3\ge\beta$, The voting result take x_3, is 6.3. after 6 cycles of voting, p_1=33.3%, p_2=0, p_3=66.7%. The data of the seventh set also do not satisfy the conditions of majority voting, we select the maximum value from p_1,p_2 and p_3, p_3(=66.7) is the maximum value, meet the conditions $p_3 \geq \beta$, The voting result take x_3, is 7.7.

When the value of β change, the voting results will be not the same as Table 1, when increasing β to 70%, the voting results were shown in Table 2.

Table 2. The Voting Process of Improved Majority Voting Algorithm

(ε=0.1, β=70%)

	X_1	X_2	X_3	P_1	P_2	P_3	voting results
1	1.2	1.5	1.4	0	0	100%	1.4
2	2.1	2.2	2.4	50%	0	50%	2.1
3	3.2	3.5	3.4	33.3%	0	66.7%	3.4
4	4.8	4.5	4.9	50%	0	50%	4.8
5	5.5	5.2	5.4	40%	0	60%	5.4
6	6.5	6.9	6.3	33.3%	0	50%	voting failed, output exception flag
7	7.3	7	7.7	28.6%	0	42.9%	voting failed, output exception flag

In Table 2, the voting results is same as β=60% for the first 5 cycles. In cycle 6, the maximum value of p_1,p_2 and p_3 is p_3, p_3=60%<β, the vote failed, output exception flag. Cycle 7 is similar as cycle 6, the vote failed, output exception flag. In the future more cycles, if a certain value of p_1、 p_2 or p_3 is greater than or equal to 70%, the situation of voting failed will avoid.

The results of majority voting algorithm are shown inTable 3.

Table 3. The Voting Process of Majority Voting Algorithm

	X_1	X_2	X_3	voting results
1	1.2	1.5	1.4	1.4
2	2.1	2.2	2.4	2.1
3	3.2	3.5	3.4	3.4
4	4.8	4.5	4.9	4.8
5	5.5	5.2	5.4	5.4
6	6.5	6.9	6.3	voting failed, output exception flag
7	7.3	7	7.7	voting failed, output exception flag

The results of median voting algorithm are shown in Table 4.

Table 4. The Voting Process of Median Voting Algorithm

	X_1	X_2	X_3	voting results
1	1.2	1.5	1.4	1.4
2	2.1	2.2	2.4	2.2
3	3.2	3.5	3.4	3.4
4	4.8	4.5	4.9	4.8
5	5.5	5.2	5.4	5.4
6	6.5	6.9	6.3	6.5
7	7.3	7	7.7	7.3

From the results of majority voting algorithm and median voting algorithm, We find, when β=70%, the improved majority voting algorithm has the same results with the standard majority voting algorithm, cycle 6 and cycle 7 are voting failed, It shows that improved majority voting algorithm is safety. When β=60%, the improved majority voting algorithm can vote for all 7 cycles. Though the results are different from median voting algorithm, but it also shows that the improved majority voting algorithm is flexibility.

3.2.2. Experiment: Because of the randomness and uncertainty of the actual system output, in order to verify the validity of the improved algorithm in practical applications, we designed the detailed simulation to verify the algorithm. Reference experiment of Latif-Shabgahi *et al.*, we set the signal source as: s=10*sin（t）+100, the experimental environment was MATLAB; The experiment used the way of error-injection, compared the improved algorithm with majority voting algorithm and median voting algorithm in the three module redundancy system.

In our experiments, in order to achieve the effect of uniform error distribution, we used the formula: s_i=2*p*s*rand(1)+(1-p)*s (i=1,2,3), outputted three injected error results s_1,s_2,s_3. Let s_i gets a random value in interval [(1-p)s, (1+p)s]. p is the error magnification, the interval set for p is [0.5,1.5]. We set the threshold value ε as 30, the probability threshold value as 60%. The interference of each signal source are independent of each other.

Figure 1, 2, 3 are the comparison chart of experimental results of improved majority voting algorithm, standard majority voting algorithm and median voting algorithm. For the convenience of express, in these figures, ‘ima’ represents the improved majority voting algorithm , ‘ma’ represents the standard majority voting algorithm, ‘me’ represents the median voting algorithm .

Figure 1 shows that the improved voting algorithm has higher correct rate than the standard majority voting algorithm; Figure 2 shows that the improved voting algorithm has lower error rate than the standard majority voting algorithm and median voting algorithm ; Figure 3 shows that for situation of unable to vote, the improved voting algorithm has great improvement than the standard majority voting algorithm(Because the median voting algorithm can vote all the time, the constant is 0, so there is no marked in Figure 3)

Through analysis and contrast the curve changes of the above three graphs, we can draw a conclusion, the improved majority voting algorithm is very good to deal with the situation that the standard majority voting algorithm unable to vote, but also maintained the higher correct rate and a relatively low error rate than the standard majority voting algorithm. This proves that the improved majority voting algorithm is effectiveness, with the error magnification increase, the performance of the improved algorithm is more excellent.

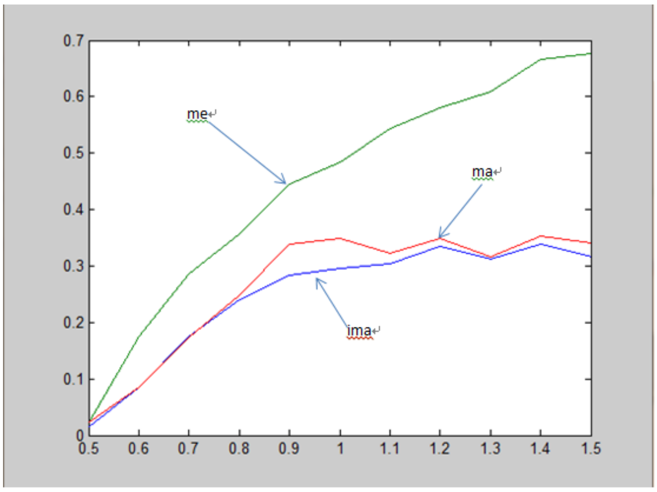


Figure 1. The Relationship between the Error Magnification and Correct Rate

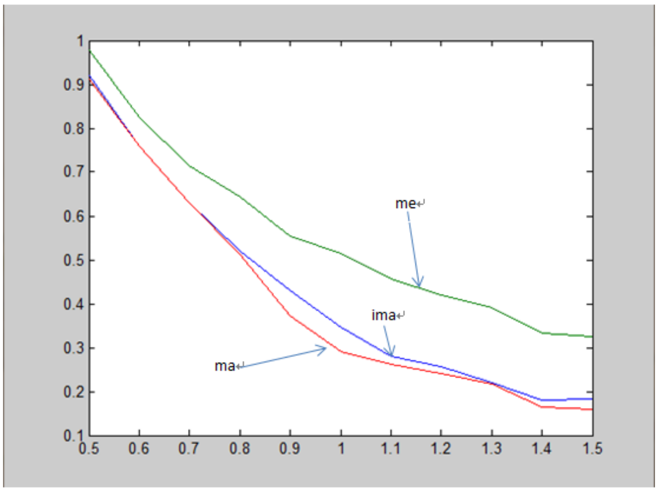


Figure 2. The Relationship between the Error Magnification and Error Rate

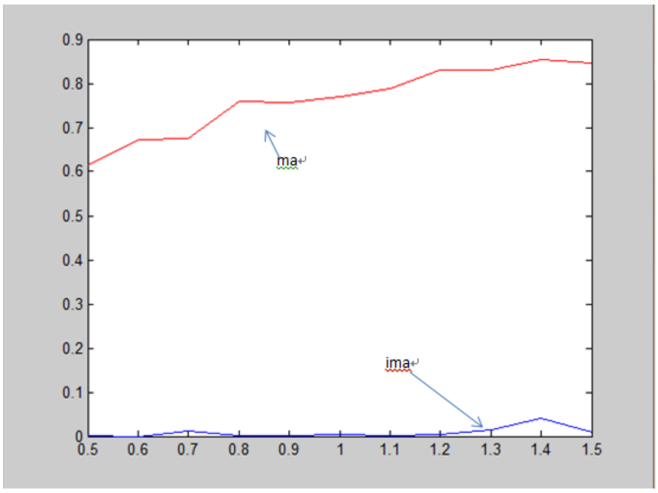


Figure 3. The Relationship between the Error Magnification and the Rate of Unable to Vote

4. Conclusions

This paper presented an improved majority voting algorithm, and compared the algorithm with the standard majority voting algorithm and the median voting algorithm through experiments, it is proved that the improved algorithm achieves a compromise between the result selection capabilities of the median voting algorithm and the safety features of the majority voting algorithm.

5. Acknowledgements

This paper is supported by the Key Laboratory of Trustworthy Distributed Computing and Service (BUPT), Ministry of Education.

6. References

6.1. Journal Article

[1] G. Latif-Shabgahi, S. Bennett and J. M. Bass, "Smoot-hing voter：a novel voting algorithm for handling multi-ple errors in fault-tolerant control systems", Micro-processors and Microsystems, **(2003)**, pp. 303-313.

[2] Y. Zhang and P. Guo, “Decision Tree to Optimize Voter”, Computer Development & Applications, vol. 23, no. 2, **(2010),** pp. 89-91.

6.2. Conference Proceedings

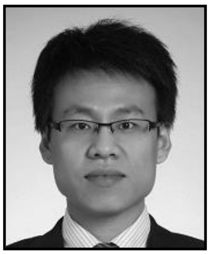
[3] R. Paul and K. Lovczak, Alper, “A Theoretical Investigation of Generalized Voters for Redundant Systems”, Proceedings of IEEE 19th International Symposium on Fault Tolerant Computing Systems, **(1989)**, pp. 444-451.

[4] Y. Zhang and P. Guo, “Modified Majority Voter for Triple Modular Redundancy System”, The Thirteenth National Conference on fault-tolerant computing, China Computer Society, **(2010),** pp. 27-30.

Authors

Jincui Yang, She received her PhD (2013) in computer science and technology from Beijing University of Posts and Telecommunications. Now she is a lecturer of software engineering at Software Engineering Department, Beijing University of Posts and Telecommunications. Her current research interests include different aspects of Software Engineering, Trustworthiness & The Internet of Things.

Lirong Qiu, She received her PhD in Information Sciences (2007) from Chinese Academy of Science. Now she is an assistant professor of computer sciences at Information Engineering Department, Minzu University of China. Her current research interests include different aspects of natural language processing, artificial intelligence and distributed systems.



Yu Zhang, He received his M. E. (2013) in computer science and technology from Beijing University of Posts and Telecommunications. Now he is an engineer at Information Technology Department, China Post Life Insurance Company Limited. His current research interests include software engineering, software project management.