Developing a New Weighted Voting Algorithm based on Markov Model

Omid Norouzifar[1], Manouchehr Kazemi[1,*], Mahdieh Nadi Senejani[1]

[1]Department of Computer Engineering, Faculty of Engineering, Ashtian Branch, Islamic Azad University, Ashtian, Iran
*corresponding author : univer_ka@yahoo.com

Abstract

Voting algorithms play a significant role in most fault-tolerant and control systems so these algorithms are continually in progress and with regard to specific situations in different systems, new voting algorithms are growing and increase the reliability and fault tolerability of systems. In this paper, a new combined algorithm using a weighted voter based on Markov chain is presented that is increased the reliability of system than both of median voter and average voter. In this paper, the proposed algorithm on average weather data of a year in Tehran was tested and simulation results show the effectiveness of the method than the other methods.

Keywords: Voting algorithms, fault-tolerant, Markov chains, reliability, temperature prediction algorithm, weighted voting

1. Introduction

In the top priority and fault-tolerant systems that the availability and reliability have extremely important, in the control systems in order to improve reliability usually they try to cover errors by using redundancy techniques and to provide the best outputs among different modules. Redundancy techniques in hardware or software are used to produce the best output and to cover errors due to the operation of the digital systems.
Redundancy has four samples:
1. Hardware redundancy
2. Software redundancy
3. Information redundancy
4. Time redundancy

There are three types of hardware redundancy: passive, active and hybrid. Voting algorithm is more about passive hardware redundancy. In this method, after the occurrence of an error in system, at first the error is found but the system continues to operate and active hardware redundancy uses backup module in addition to appearing error and the system continues to operate and finally the hybrid hardware redundancy that is actually a combination of two active and passive types has high price compared to other types and it is most used in systems with high life time. There is stiff competition between different voting algorithms. In general, the voting algorithm can be divided into several categories, passive hardware redundancy is a model in which all the modules is used initially and while operation or occurrence of defect there is no change in its configuration. In this way, system failure is prevented immediately because it does not require reconfiguration after failure (which causes delays in the operation of the system).

In general, passive redundancy has not many species. The main species of this redundancy is N-Modular Redundancy or NMR [10] in which there are N same copies of a module that all of them are doing the same processing on the input of system and their results go to a majority vote. If at least M numbers of them are correctly working, voters will produce the correct result. Usually M is selected equal to 1+2/N, but to increase the trust of voter output, value of M can also consider further. One of the special cases of NMR that is used very much is TMR or Triple Modular Redundancy. In this case, at least two modules from three existing module must be working properly till the system work properly. As can be seen in Figure 1, TMR redundancy works in this way [7].
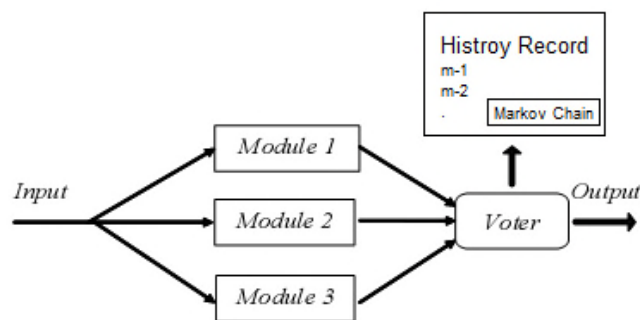

Fig.1: TMR redundancy

As can be seen in Figure 1, by using this algorithm, median and average voters among the desired outputs give appropriate output. These algorithms are used in systems that must always be available and the availability of the system has high importance. In this paper we have introduced a new model that by using a weighted algorithm [4, 5] and Markov chain [1, 2] gives better results than the other voters and increases the system reliability. As in weighted voters [3], the output result uses a series of available information or information history in addition to redundancy modules, this information can help us to get a better result as output by using a series of special techniques. In the next section we introduce the Markov chain as well as how to use it in new voting algorithm that is produce the output by using a series of information.

2. Research literature and background

Numerous papers about voting algorithm are studied. The issue of voting algorithm is constantly evolving and creating new algorithms using the new rules and upgrading old algorithms using new and update methods. In general, voters can be divided into various categories in term of its use and performance. Some of the voting algorithms are useful in applications that a result must be produced in every voting cycle. Another group of voting algorithms announced the value of one of the inputs as output if there is a general agreement among a certain number of redundancy input. In the case of disagreement these algorithms can establish one of the following cases or stop system activity by generate an Exception Flag.

1. It directed the system to a safe mode.
2. It activated the detection and isolate of fault.
3. It activated the performance decline mechanisms.
4. It tested and reviewed the accuracy of voting inputs with this purpose that one of them show the specific case. This type of voting algorithm is used in some of the disagreement case that can create flexible output for critical security applications.

This function can also be explored from a different perspective. Sometimes the voting algorithms judged only between the input redundancy values to produce the output. Sometimes it used some additional information such as online reports and information recessions of the past and so on. And sometimes it used acceptance test procedures to produce output.

Voting algorithm category can be implemented from some point of views:

1. Type of implementation method which is divided into two categories: software implementation and hardware implementation.
2. Type of agreement in voting that can be used in two ways: deterministic or probabilistic.
3. Based on the output space size (large - small)
4. Nature of environment where decision-making is performed (synchronous - asynchronous).

In general, flexibility of software voters is much more than hardware voters and the main difference in the hardware and software models is more related to the flexibility and speed. Other voting algorithms can be plurality voting algorithm that actually is the developed form of majority algorithm and perform the m out of n voting [8]. This voter selects output that more modules have agreed on it. In the case of TMR redundancy, the majority and plurality voter have the same output. Weighted average voter algorithm uses Zack Lavark formula [4, 5, and 7] and it was presented in 1989. Smoothing voting algorithm [9] is the developed form of majority algorithm that a type of Acceptance Test has been added to it. In smoothing when disagreement occurs between the variable results, the closest result to the previous output of voter is selected as a potential result for this voting cycle and if the measured value is less than the predetermined value that is called smoothing threshold, this result is selected as voter output result.

As mentioned in introduction section, the number and types of voting algorithms are constantly developing and the developers of this type of algorithms are constantly striving to enhance the reliability and safety of existing algorithms. For this purpose, the most famous mathematical algorithms can be used in this discussion and a new type of voter algorithm can be created that is used in various applications depending on their type. Some of these algorithms are: Bit Voter algorithms [12], neural voter algorithms [3], genetic voter algorithm [11], Hardware Voter algorithm, internal voter algorithm, threshold voter algorithm, predictive voter algorithm, first, second and third order voter algorithms, Hybrid Voter algorithm, fuzzy voter algorithm, plurality voter algorithm and unanimity voter algorithm.

3. The proposed method

Markov chain is a stochastic process at time that is in specific mode at every step and deforms at each step randomly. Steps are often considered as moments of time but they can be considered as physical distance or any other discrete variable. The process future depends only on the current state and not related to the process past. In other words, the current state of the system keeps all information about next developments of the process. In this paper, we use m order Markov chains (m is finite) and past data until two previous steps. General formula of Markov chain is as follows:

$$\Pr(X_n = x_n \mid X_{n-1} = x_{n-1}, X_{n-2} = x_{n-2}, \ldots, X_1 = x_1)$$
$$= \Pr(X_n = x_n \mid X_{n-1} = x_{n-1}, X_{n-2} = x_{n-2}, \ldots, X_{n-m} = x_{n-m}) \quad for \; n > m \tag{1}$$

In other words, the next state is depending on the previous m state.

The possibility of change the state from i[th] state to j[th] state in n move is:

$$P_{ij}^{(n)} = \Pr(X_n = j \mid X_. = i \tag{2}$$

In this paper, the second order Markov chain (i.e. data from 2 previous steps) is used.

3.1. Simulation

In the following, we introduce this algorithm to demonstrate the performance of this algorithm for detection the temperature of an environment using a TMR redundancy module and output production based on Markov model. Performance of this voter is as follow: To obtain a suitable output and to compare the weighted voter algorithm based on Markov model, the outputs of these three modules in median and average voter algorithm are obtained and compared with weighted voter algorithm based on Markov model. In the case of median voter algorithm, the median output of 3 modules is checked and the output is produced. In the case of average voter algorithm, the outputs of 3 modules are added together and divided by 3 and the output is produced. But in weighted voter algorithm based on Markov model we are using m order Markov chain and using the temperature prediction data of environment, to obtain the possibility of predictions accuracy transfer for two previous steps and according to the percentage of predicted accuracy (current situation of our system), we predict the future state of the system and we produce the output of the algorithm based on this prediction. In the following, we explain this by an example. We considered the average temperature data in Tehran for 2015 as a raw data to prove this. This data has been received from reputable meteorological sites. We tested the weighted voter algorithm based on Markov model by injection of false data to the modules and changing daily temperature. In this example, we injected daily temperature with [0, 1] °C to the actual temperature and also we injected TMR modules with [0, 1] and [-2, 2] and [-5, 5] °C as false information. In the following, variables N, N_a, N_c, N_{ic}, NC/N show the number of voter components, the number of successful results between n output, the number of accepted correct results, the number of accepted incorrect results, the number of correct results onto the number of voter components, Respectively. This parameters show the reliability and safety of the system that is obtained by $S = 1 – Nic/N$. In the following, we will examine the above model by using these parameters.

3.2. Implementation

How to implement the presented algorithms is explained at the following, first we read the predicted temperature data of a year from simulation program database and we show the read data in table, then at the same time, we read the information of 3 TMR module output in other table. It should be noted that the output information of modules are produced already and are available in the database. Then, by using an output loop of modules for each day, three median, average and weighted voter algorithms based on Markov model are calculated. The median and average operation principles are very simple and the voting output is easily calculated. But in calculation of voting output of the proposed algorithm, at first the desired day is found based on information of first table and the accuracy of temperature in two previous days is calculated given the second order Markov model and according to the accuracy percentage of predictions, the nearest module output that is closer to predicted number is selected as the proposed algorithm output and finally, outputs with more accurate answers is marked with * sign and then accuracy and inaccuracy of each of used algorithms is shown in output.

As can be seen in Table 1, these results are the implementation of our algorithm for 365 days in a year and parameters that were introduced in previous section are calculated for three mentioned algorithms. You can compare these parameters. It should be noted that the simulation these algorithms have been studied by writing a program in C # and the desired program will be available if requested.

Table 1: *Computational parameters*

Parameters	Average	Median	The proposed method
N	365	365	365
N_a	365	365	365
N_c	126	90	177
N_{ic}	237	273	186
N_a/N	100	100	100
N_c/N	34.52	24.65	48.49
N_{ic}/N	64.93	74.79	50.95
N_c/N_{ic}	53.16	32.96	95.16

And also 20 order of above algorithms implementation is shown in Table 2. The columns in this table are: Column 1 is the date of the desired day in 2015, Column 2 is Average temperature predicted by the meteorological sites, Column 3 is the average temperature of that day, Columns 4, 5 and 6 are the output of TMR modules, Column 7 is the accuracy percentage of temperature predictions in two days ago and Columns 8, 9 and 10 are the outputs of voting algorithms. The operation of three voters is shown in this table which indicates that weighted voting algorithm based on Markov model has better output than the other algorithms. For example, the number of correct results in weighted algorithm based on Markov model is 177, but this number in the median algorithm is 90 and in the average algorithm is 126, also the number of incorrect results in weighted algorithm based on Markov model is 186, but this number in the median algorithm is 273 and in the average algorithm is 237 and also the reliability that is obtained by using Nc/N is 48.49 in weighted algorithm based on Markov model but this number in the median algorithm is 24.65 and in the average algorithm is 34.52, this indicates that the above algorithm is better than the other algorithms. The numbers of studied days and the amount of false information injected on modules have a huge influence on the outputs.

Table 2: Implementation results

Date	C1	C2	Module1	Module2	Module3	Percent	Markov	Median	Average
1/24/2015	11	10	10	9	15	99%	10.89 *	15	11.33333
1/25/2015	12	12	13	14	17	99%	11.88 *	17	14.66667
1/26/2015	12	11	10	13	10	99%	11.88	13	11 *
1/27/2015	11	11	11	12	14	99%	10.89 *	14	12.33333
1/28/2015	13	12	13	10	17	99%	12.87 *	17	13.33333
1/29/2015	10	9	9	7	14	99%	9.9 *	14	10
1/30/2015	9	9	8	9	9	97%	8.73	9 *	8.666667
1/31/2015	8	9	9	10	13	99%	7.92 *	13	10.66667
2/1/2015	7	8	7	9	13	99%	6.93 *	13	9.666667
2/2/2015	6	7	8	5	9	97%	5.82	9	7.333333 *
2/3/2015	4	5	5	5	8	97%	3.88	8	6 *
2/4/2015	6	5	4	6	0	97%	5.82 *	6	3.333333
2/5/2015	8	9	10	7	14	97%	7.76 *	14	10.33333
2/6/2015	9	8	8	10	6	97%	8.73	10	8 *
2/7/2015	10	11	10	12	7	97%	9.7	12 *	9.666667
2/8/2015	11	10	9	9	10	97%	10.67	10 *	9.333333
2/9/2015	12	13	13	13	11	97%	11.64	13 *	12.33333

2/10/2015	11	11	10	12	14	97%	10.67 *	14	12
2/11/2015	12	11	12	10	6	99%	11.88 *	12	9.333333
2/12/2015	10	10	11	10	13	99%	9.9 *	13	11.33333

Items those are specified by (*) in Table 2 have more accurate output than other used voting algorithms.

4. Results

In this paper we introduce new voting algorithm based on Markov chain. As you saw earlier, we operate the voting algorithm by using a redundancy TMR module and by using Markov chain of order m. The use of this algorithm than other algorithms led to increase the reliability of the system as well as the number of correct results is more than other algorithms.
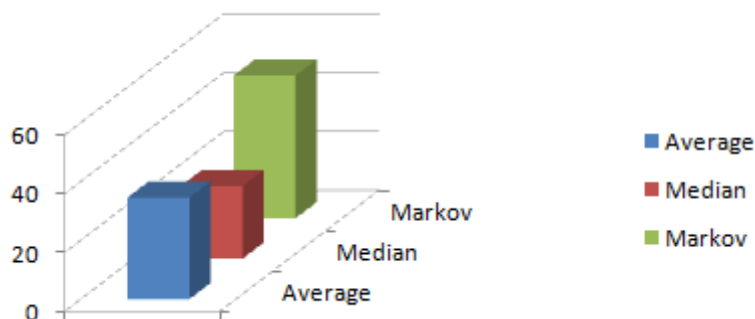

Fig. 2: Number of correct answers

Also, the number of incorrect results is less than other algorithms.
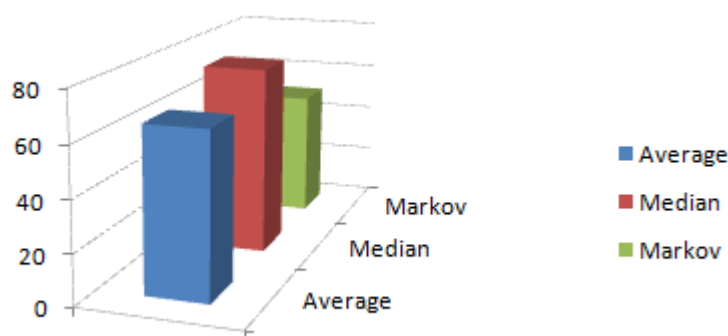

Fig. 3: Number of incorrect answers

In the high-availability and safety-critical systems, it is very important to put a high degree of reliability in these systems.

References

[1] Hassan H. Halawa ;Ramez M. Daoud ; Hassanein H. Amer ; Gehad I. Alkady ; Ali AbdelKader "FPGA-based reliable TMR controller design for S2A architectures" Computer Science and Information Technology (ICCSIT), 2015 IEEE 20th Conference on Emerging Technologies & Factory Automation (ETFA) , 11 Sept. 2015 , pp 1 – 8

[2] Kwang Seop Sona, Dong Hoon Kima, Chang Hwoi Kima, Hyun Gook Kangb, "Study on the systematic approach of Markov modeling for dependability analysis of complex fault-tolerant features with voting logics" Reliability Engineering & System Safety , vol.150,pp 44 – 57 , 2916

[3] Zarafshan F, G .Latif-Shabgahi ,and Karimi A, "A novel weighted voting algorithm based on neural networks for fault-tolerant systems" Computer Science and Information Technology (ICCSIT), 3rd IEEE International Conference , vol.9,pp 135 – 139 , 2010

[4] G. Latif-Shabgahi, "A Novel Algorithm for Weighted Average Voting Used in Fault-Tolerant Computing Systems, " Microprocessors and Microsystems, vol. 28, pp. 357-361, 2004.

[5] G. Latif-Shabgahi, J. M. Bass, and S. Bennett, "History-Based Weighted Average Voter: A Novel Software Voting Algorithm for Fault-Tolerant Computer Systems, " Euromicro Conference on Parallel, Distributed, and Network-Based Processing, pp. 402-409, 2001.

[6]   Z. Tong and R. Y Kain, "Vote Assignments in Weighted Voting Mechanisms, " IEEE Transactions on Computers, vol. 40, pp. 664-667, 1991. (Pubitemid 21675667)

[7] Iuri A. C. Gomes ; Mayler Martins ; Andre Reis ; Fernanda Lima Kastensmidt, "Using only redundant modules with approximate logic to reduce drastically area overhead in TMR, " 2015 16th Latin-American Test Symposium (LATS), pp. 1-6 , 2015.

[8]   Thomas Gamer ; Manuel Oriol ; Michael Wahler, "Increasing efficiency of M-out-of-N redundancy, " Proceedings of the 2014 IEEE Emerging Technology and Factory Automation (ETFA) , pp. 1 - 8, 2014.

[9] G. Latif-Shabgahi, S. Bennett, and J. M. Bass, "Smoothing voter: a novel voting algorithm for handling multiple errors in fault-tolerant control systems," Microprocessors and Microsystems, vol. 27, pp. 303-313, 2003.

[10] Eric P. Kim ; Naresh R. Shanbhag, "Soft N-Modular Redundancy, "IEEE Transactions on Computers ,V.61, pp. 323 - 336, 2012.

[11] Zahra Latifi, Abbas Karimi ,"A TMR Genetic Voting Algorithm for Fault-tolerant Medical Robot," Medical and Rehabilitation Robotics and Instrumentation, V.42, pp. 301-307, 2014.

[12] B. Parhami , "Design of m-out-of-n bit-voters ," Signals, Systems and Computers, 1991. 1991 Conference Record of the Twenty-Fifth Asilomar Conference on, V.2, pp.1260-1264,1991.