A NOVEL FAMILY OF WEIGHTED AVERAGE VOTERS FOR FAULT-TOLERANT COMPUTER CONTROL SYSTEMS

G. Latif-Shabgahi[*], A. J. Hirst[*], and S. Bennett[+]

[*]*Department of Telematics, Open University*
Walton Hall, Milton Keynes, MK7 6AA, UK
Fax. +44(1908)-653658 , Email: *g.r.latif@open.ac.uk*

[+]*Automatic Control and Systems Engineering Dept,*
The University of Sheffield, Sheffield, S1 3JD, UK

Key Words. *Computer Control, Fault Tolerance, Triple Modular Redundancy, Dependability, Software Voting.*

Abstract

Fault masking is a widely used strategy for increasing the safety and reliability of computer control systems. The approach uses some form of voting to arbitrate between the results of hardware or software redundant modules for masking faults. Several voting algorithms have been used in fault tolerant control systems; each has different features, which makes it more applicable to some system types than others. This paper introduces a novel family of weighted average voters suitable for redundant sensor (and other inertial measurement unit) planes, at the interface level, of control systems. It uses two tuneable parameters, each with a ready interpretation, to provide a flexible voting performance when using the voter in different applications. The weight assignment technique is transparent to the user because the impact of the degree of agreement between any voter input and the other inputs is directly reflected in the weight value assigned to that input. The voter can be tuned to behave as the well-known inexact majority voter that is generally used in safety-critical control systems at different voting planes. We evaluated the performance of four versions of the novel voter through a series of fault injection experiments, and compared the results with those of the well-known Lorczak's weighted average voter. The experimental results showed that the novel voter gives more correct outputs (1%-12% higher reliability) than the Lorczak's voter in the presence of small permanent and transient errors. With large errors, lower-order versions of the novel voter give better performance than the ones with higher orders.

1 Introduction

Increasing dependability is one of the primary concerns in many real-time control systems. These applications include safety-critical computer control systems (e.g., flight control systems and nuclear power plant control), highly reliable applications (e.g., railway-interlocking system), and highly available systems (e.g., distributed databases). Such applications often use redundancy to reduce the risk associated with relying upon any single component operating flawlessly, and to ensure safety, reliability, availability and data integrity. Triple Modular Redundancy, TMR, and 3-Version Programming, 3VP, are commonly used in fault tolerant systems to provide passive redundancy for masking runtime faults at hardware and software levels respectively [6, 14]. The outputs from three independently developed but functionally identical modules operating in parallel with the same inputs are supplied to a voting unit that arbitrates between them to produce an overall output (Figure 1). Several voting techniques have been described in the literature; examples are majority, plurality, median, weighted average, predictor, step-wise negotiated voting, and maximum likelihood voters [1, 4, 7, 11, 12, 13]. They are generally divided into two main categories: *type A* (*agreement-based) voters* which produce an output from redundant inputs if there is agreement between a particular number of voter inputs (e.g., majority and plurality voting), and *type B voters* that always produce an output regardless of the agreement, or otherwise, between redundant inputs. Type B voters either amalgamate the inputs or simply select one of them based on a particular metric (e.g., weighted average voter and mid-value selector respectively).

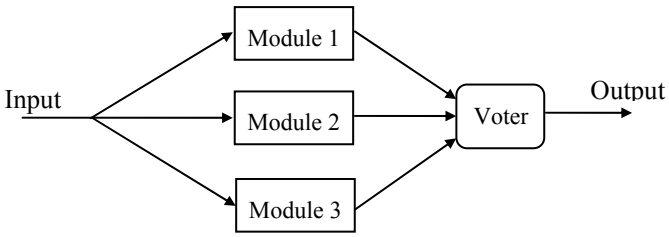

Figure 1. A Triple Modular Redundancy System

In many cases *type B* voting algorithms have to be used to produce a single value from the results of redundant modules. Voting on the output of redundant sensors and inertial measurement units at the interface level of control systems or chemical plants, and on the reading of distributed clocks in distributed computing nodes for clock synchronisation purposes are two popular examples of the application of the type B voters. In such applications, a weighted average voter is more trustable than a median voter. This is because a median voter simply selects the mid-value of all its redundant inputs whereas a weighted average voter generates weights $w_1, w_2, \ldots w_N$ that scale the contribution of each input, $x_1, x_2, \ldots x_N,$, to the output result. In this paper, we introduce and describe a novel family of weighted average voters that have some benefits over the previously introduced weighted average voters for computing weights while giving better performance in terms of safety and reliability.

The paper is organised as follows. Section 2 reviews the related works. Section 3 introduces a novel weighted average voting scheme. Section 4 describes the experimental test harness, methodology, and test results, comparing reliability of the Lorczak's distance-based weighted average voter with that of the four versions of the novel weighted average voter. The impact of the roll-off parameter on the performance of the novel voter is also investigated. Finally, some conclusions are presented in section 5.

2 Related Works

The weighted average voter with n inputs computes the weighted mean of module results in any voting cycle. A weighting factor, w_i is assigned to any voter input x_i and the final output y is calculated as $y = \sum w_i . x_i / \sum w_i$. The weights can be predetermined or can be adjusted dynamically. Pre-determined weights can be based on *a priori* estimates of the reliability of the modules or on the *a priori* probability of failure of redundant modules [15]. For dynamic adjustment three strategies have been suggested in the literature. In the first method weights are computed either from some mechanism in the module, e.g., if the input data is coming from redundant self-validating sensors [5], weights can be considered as a function of confidence measures given by sensors [2]. In the second method weights are calculated based on the distances between module results, $d_{ij} = | x_i - x_j | : \mathrm{i, j} = 1,2,\ldots \mathrm{N}$ and $\mathrm{i} \neq \mathrm{j}$ [4]. An input value that differs greatly from the other input values is assigned a smaller weight than an input value that is close to any of the other input values. Among the distance measure-based weighting approaches, a modification to one of the algorithms introduced in [4], which we shall refer to as Lorczak's algorithm, has received most attention in the literature. This algorithm uses the following equation (1) to calculate the weight values from which the voter output is computed by means of equation (2). The algorithm uses an application-specific tolerance factor β to tune the weight values.

$$w_i = \frac{1}{1 + \prod_{\substack{i=1, j=1 \\ j \neq i}}^{N} \frac{d^2{}_{ij}}{\beta^2}} \quad (1) \qquad y = \frac{\sum_{i=1}^{N} w_i . x_i}{\sum_{i=1}^{N} w_i} \; | \, i = 1,\ldots \mathrm{N} \quad (2)$$

In the third method, introduced in [9], on-line history record of modules is used to adjust weights. At any time, the history record of a given module indicates the number of its contribution to consensus with other modules from a time base. A module result with a higher history record is assigned to a higher weighing value than those with lower history record values. Experimental results have shown that this voter produces more correct and less incorrect results than the Lorczak's distance based weighted average voter. This paper introduces another group of distance measure-based weighted average voters that are described in detail in the next section. They are applicable for handling the output of an array of skewed sensors in safety related applications such as weapons and transportation systems. In the remainder of this paper, for simplicity, we will refer the "distance metric-based weighted average" voter as the "wa" voter.

3 Voter Implementation

The novel m-input voters are implemented as follows:

1. Let $x_1, x_2, \ldots x_m$ be the voter inputs and y its output.
2. Determine the 'degree of closeness', or 'level of agreement', of all $m(m-1)/2$ voter input pairs based on the equation (3).

$$s_{ij} = \frac{1}{1 + p d_{ij}{}^q} \quad (3)$$

This function (called as '*agreement indicator*', in this paper) gives a value for the agreement of voter inputs j and i based on their numerical distance. The parameter q tunes the rate of roll-off of this function and the parameter p is used to set the midpoint value of the function, a. The function is continuous and generates $s_{ij} = 1$ for numerically equivalent input pairs. For differing inputs, it produces a real-value in the range (0 1) such that as d_{ij} gets larger, s_{ij} tends towards zero. For comparison purposes, we characterise the curves using the midpoint value $d_{ij} = a$ such that $s_{ij} = 0.5$. Therefore, $p = a^{-q}$ and $s_{ij} = \frac{1}{1 + (\frac{d_{ij}}{a})^q}$. Figure 2 indicates this function for different values of q $(q=1, 2, 3, 4)$ where $a=0.5$. It shows that by increasing the parameter q the behaviour of the novel voter moves toward that of the standard inexact majority voter with a hard threshold a [1].

Where $q=1$ → $s_{ij} = \frac{1}{1+(\frac{d_{ij}}{0.5})^1} = \frac{1}{1+2.d_{ij}}$,

where $q=2$ → $s_{ij} = \frac{1}{1+(\frac{d_{ij}}{0.5})^2} = \frac{1}{1+4.d_{ij}}$,

and so on.

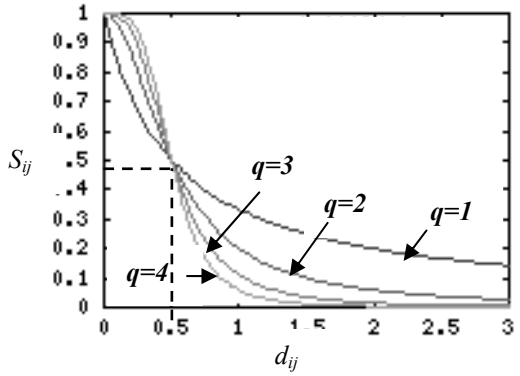

Figure 2. Continuous agreement indicator relating pairwise agreement (s_{ij}) and distance (d_{ij})

3. Having computed *agreement indicator* values for all of the voter input pairs, assign the weight value of each voter input *i* based on equation (4).

$$w_i = \frac{\sum_{j=1, i\neq j}^{m} s_{ij}}{m-1} \qquad (4)$$

This type of weighting directly indicates the influence of the agreement between any particular input and the other inputs on its weight value. This explicit relationship cannot be seen in the weighting approach of Lorczak's algorithm and most of the methods introduced in [4].

4. The voter output is then given by equation (2).

4 Experimental Methodology

The details of experimental test harness for software voters used in this work, and the method of experiments have been presented in [3]. These are briefly explained below.

4.1 Test harness structure

The established experimental test harness, shown in Figure 3, simulates a TMR system. The test harness comprises an input data generator, a replicator, three saboteurs (to inject errors to replicated input data), a voter, and a comparator. The input generator produces one notional correct result in each test cycle. This sequence of numbers simulates identical correct results generated by redundant modules. Copies of the notional correct result are presented to each saboteur in every cycle. The saboteurs can be programmed to introduce selected module error amplitudes, according to selected random distributions. In a given set of tests one, two or three saboteurs may be activated to simulate module result errors on the voter inputs. The outputs of all saboteurs are presented as inputs to the voter under test, and the voter output is compared to the notional correct value for that cycle by the comparator. It is assumed that:

- That all voters perform correctly. This assumption is made due to the fact that the voting algorithm is usually a simpler program than the modules it monitors.
- That all voters are used in a cyclic system where there exists some relationship between correct results from one cycle to the next (e.g. controlling the fuel supply to an engine);
- That faults cause errors whose symptoms appear to the voter as numerical input values perturbed by varying amounts.
- At any voting cycle the "notional correct result" is known.
- An *accuracy threshold*, ε, is used, in the comparator, to determine if the distance between the notional correct result and the voter output is within acceptable limits. A voter result which has a distance from the notional correct answer less than the accuracy threshold is taken as a *correct* output, otherwise it is considered as an *incorrect* output.

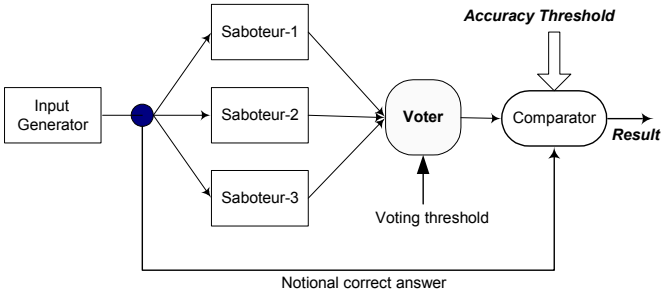

Figure 3. Experimental Test Harness

4.2 Experimental method

The input to the modules is given by a sinusoidal function $u(t) = 40.\sin(t)+40$ sampled at *0.1 sec*. The accuracy-threshold value, ε = 0.5. To examine the performance of voters in the worst case, all modules are perturbed with uniformly distributed errors in the range $[-e_{max} \quad +e_{max}]$. Therefore, in all voting cycles, the maximum possible deviation between any two module results is $d_{max} = 2.e_{max}$. The *a* parameter of the novel weighted average voters is set the same as β for the Lorczak's weighted average voter.

Depending on the numerical distance between a voter output and notional correct result, any output value of a voter can be interpreted as a *correct, incorrect, or benign* (disagreed) answer. For each voter, the results of $N=10^4$ voting actions are classified. In this way, n_c correct results, n_{ic} incorrect

outputs, and n_d benign results are collected. These data are used to compare the voters. A number of performance measures can be defined for this purpose. For example, we can define the ratio of correct voter outputs to the number of voting actions, $A = n_c / N$, as the *reliability measure* of a TMR system (it represents the capability of a voter to produce correct results). Ideally $A=1$. The ratio n_{ic} / N is a measure of catastrophic outputs; it can be used as a measure of system *safety*. From a safety point of view, the smallest number of incorrect outputs (a low value of the ratio n_{ic} / N) is desirable, thus ideally $S=0$. Other measures such as '*disagreement selectivity*', '*availability*', '*cumulative value of square divergence value of agreed results*', and '*distribution of distances of agreed results from the notional correct results*' may also be used. However, due to page limitation, only the reliability performance of voters is investigated in this paper.

5 Experimental Results

The results of voter comparison in two sets of experiments, in terms of reliability, are presented in this subsection. Four versions of the novel weighted average voter using $q = 1…4$ are compared against Lorczak's weighted average.

5.1 Reliability performance with transient errors

The results of the voters when subjected to uniformly distributed transient errors from the range [0.5 5] are shown versus the amplitude of injected errors, $|E|$, in Figure 4. Before going forward, it must be noted that faults encountered by control systems are either transient or permanent. As hardware manufacturing technology and software engineering methods improve, permanent faults/errors are gradually decreasing, and instant failures of computers due to transient faults/errors remain the main reasons for system failures. According to [8] more than 90% of field failures have been reported as being caused by transient faults/errors. As the figure shows, for small errors (where $|E|<\sim 1.5$) all the versions of the novel voter give more correct outputs than the Lorczak's weighted average voter does. For larger errors, the versions of the novel voter with $q=2, 3$ give better performance than that of the Lorczak's voter. The version with $q=4$ has slightly lower performance than the Lorczak's voter. This is because as q increases, the performance of the novel voter deteriorates and tends towards that of the standard inexact majority voter with a hard threshold at $d_{ij} = a$. The best performance belongs to a version with $q=1$ which achieves up to 1% better performance (e.g., at error point 2.5) than the Lorczak's voter, a considerable improvement for transient errors. Since the number of correct outputs of a voter has been defined as a measure of reliability, it is concluded that the novel voter behaves more reliably than Lorczak's weighted average voter for the examined errors. However, the interpretation of the agreement function for $q=1$ is problematic, since even significant differences between inputs result in a non-zero agreement value for them.

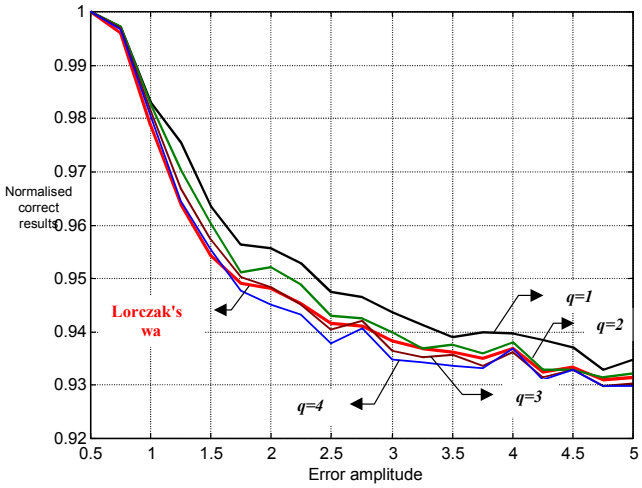

Figure 4. Comparing normalised correct output of voters for 10^4 voting cycles with transient errors when $a=1$

The impact of changing the parameter q on the behaviour of the novel voter can also be seen from Figure 4. Increasing q results in less correct outputs for all error cases. Once again, by increasing q the behaviour of the novel voter moves toward that of the standard inexact majority voter with a hard threshold a. This voter is known to have lower reliability (yet higher safety) than the weighted average voters [3, 10]. For large errors (the tail of the plots in Figure 4) the performance of all the novel voters converges.

5.2 Reliability performance with permanent errors

Figure 5 indicates the comparative behaviour of voters in the presence of small permanent errors in terms of producing correct outputs. The figure shows the improved performance of the versions of the novel voter with this type of error (1% up to 12%).

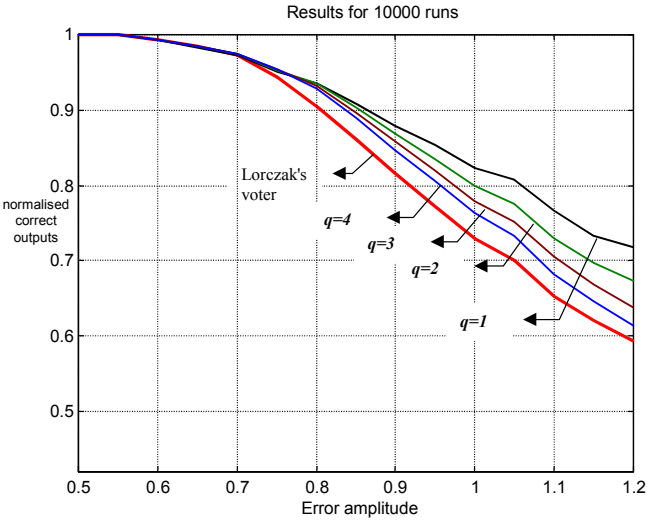

Figure 5. Comparing normalised correct output of voters for 10^4 voting cycles with permanent errors when $a=1$

6 Conclusions

The family of curves described herein are not the only curves that can be used to define a continuous agreement function. We can generalise the approach further to define $s_{ij} = \frac{1}{1+f(d_{ij})}$ where $f(0)=0$ and $f(d_{ij}): d_{ij} > 0$ is strictly increasing. We also require $s_{ij} \to 0$ for large d_{ij}. More generally, we might specify an upper limit $d_{ij,\max} : s_{ij} < \varepsilon$ in order to ensure that where disagreement between inputs is significant the resulting weight value is negligible.

The weighted averaging approach used in this paper for the novel family of weighted average voters has three benefits over the previously published voters. Firstly, in the novel voters, the influence of the agreement between any input and the others is directly reflected in the weight value assigned to that input. Such a straightforward relationship cannot be seen in the previously published weighted average voters. Secondly, in the novel voters, there are two tuneable parameters (*a* and *q*) which each have a ready interpretation: *a* sets the nominal threshold (that is, the midpoint) of the agreement indicator, and *q* the roll off of the function. In the other weighted average voters the value of weights can be adjusted only through a single scaling parameter (β in Lorczak's and Broen's voters) and its interpretation is difficult. This tunability provides flexibility when using the novel voters in a variety of applications with different requirements. For example, the performance of the version of the novel voter with higher *q* moves toward that of the inexact majority voter with hard threshold value *a*. Finally, the novel voters are computationally simpler than the other mentioned voters. The experimental results showed that for all type of small errors (transient and permanent) the versions of novel voter gives more correct results (higher reliability) than that of the Lorczak's weighted average voter. With larger errors versions of the novel voter with smaller *q* parameter give better outputs. Moreover, the version of the novel voter with lower *q* gives more correct outputs that the one with higher *q*.

References

[1]. J. M. Bass, P. R. Croll, P. J. Fleming, and L. J. C. Woolliscroft (1994). "Three Domain Voting in Real-Time Distributed Control Systems", Second IEEE Euromicro Workshop on Parallel and Distributed Processing, pp. 317-324.

[2]. H. Benitez-Perez, G. Latif-Shabgahi, J. M. Bass, H. A. Thompson, S. Bennett, and P. J. Fleming (1999). "Integration and Comparison of FDI and Fault Masking Features in Embedded Control Systems", Proc. of the 14th World Congress of Int. Federation of Automatic Control, Vol. P, Beijing, China, July 5-9, pp. 31-36.

[3]. S. Bennett, and G. Latif-Shabgahi (1999). "Evaluation the Performance of Voting Algorithms Used in Fault Tolerant Control Systems", Proc. 14th World Congress of IFAC, Vol. Q, Beijing, China, July 5-9, pp. 525-530.

[4]. R. B. Broen (1975). "New Voters for Redundant Systems", ASME Journal of Dynamic Systems, Measurement and Control, March, pp. 41-45.

[5]. M. P. Henry, and D. W. Clarke (1993). "The Self-Validation Sensors: Rationale, Definitions and Examples", IFAC Control Engineering Practice, Vol. 1, No. 4, pp. 585-610.

[6]. B. W. Johnson (1989). "Design and Analysis of Fault-Tolerant Digital Systems", Addison-Wesley, New York.

[7]. K. Kanekawa, H. Maejima, H. Kato and H. Ihara (1989) "Dependable On-Board Computer Systems with a New Method: Stepwise Negotiated Voting", Proc. IEEE 19th Ann. Int. Symp. on Fault-Tolerant Computing Systems, Chicago, USA, pp. 13-19.

[8]. H. Kim, and K. G. Shin (1995). "Design and Analysis of an Optimal Instruction Retry Policy for TMR Controller Computers", IEEE Trans. on Computers, Vol. 45, pp. 1217-1225.

[9]. G. Latif-Shabgahi, J. M. Bass, and S. Bennett (2001). "History-Based Weighted Average Voter: A Novel Software Voting Algorithm for Fault-Tolerant Computer Systems", Proc. of the PDP2001: 9th Euromicro Workshop on Parallel and Distributed Processing, February 7-9, Mantova, Italy.

[10]. Latif-Shabgahi, G., Bass, J. M. and Bennett, S. (2002). "A Taxonomy for Software Voting Algorithms Used in Safety-Critical Systems", To appear (September 2003) in the IEEE Transactions on Reliability.

[11]. Y. W. Leung (1995). "Maximum Likelihood Voting for Fault Tolerant Software with Finite Output Space", IEEE Trans. on Reliability, Vol. 44, No. 3, pp. 419-427.

[12]. P. R. Lorczak, A. K. Caglayan, and D. E. Eckhardt (1989). "A Theoretical Investigation of Generalised Voters", Digest of papers FTCS'19: IEEE 19th Ann. Int. Symp. on Fault- Tolerant Computing Systems, Chicago, IL, pp. 444-451.

[13]. S. Mitra, and E. J. McCluskey (2000). "Word Voter: A New Voter for Triple Modular Redundant Systems", Proc. 18th IEEE VLSI Test Symposium, Montreal, Canada, April 30-May 4, pp. 465-470.

[14] A. L. Pullum (2001). "Software Fault Tolerance, Techniques and Implementation", Artech House Inc, MA, USA.

[15]. Z. Tong, and R. Y. Kain (1991). "Vote Assignments in Weighted Voting Mechanisms", IEEE Trans. on Computers, Vol. C-40, No. 5, pp. 664-667.