Definition and Empirical Evaluation of Voters for Redundant Smart Sensor Systems

Definición y Evaluación Empírica de Algoritmos de Voteo para Sistemas Redundantes de Sensado Inteligente

H. Benítez Pérez*, J.L. Ortega Arjona and G. Reza Latif Shabgahi*****

***** Departamento de Ingeniería de Sistemas Computacionales y Automatización, IIMAS, UNAM, Apdo. Postal 20-726, Admon. No. 20, Del. A. Obregón, México D. F., CP. 01000, México; e-mail: hector@uxdea4.iimas.unam.mx
****** Departamento de Matemáticas, Facultad de Ciencias, UNAM, Ciudad Universitaria, CP. 04510, México City, México.
*******Telematics Dept., Technology Faculty, The Open University, Milton Keynes, MK7 6AA, UK. Internet (e-mail): g.latif@sees.bangor.ac.uk.

Article received on August 23, 2005; accepted on October 02, 2007

Abstract
This study is the first attempt for integration voting algorithms with fault diagnosis devices. Voting algorithms are used to arbitrate between the results of redundant modules in fault-tolerant systems. Smart sensors are used for FDI (Fault Detection and Isolation) purposes by means of their built in intelligence. Integration of fault masking and FDI strategies is necessary in the construction of ultra-available/safe systems with on-line fault detection capability. This article introduces a range of novel software voting algorithms which adjudicate among the results of redundant smart sensors in a Triple Modular Redundant (TMR) system. Techniques to integrate replicated smart sensors and fault masking approach are discussed, and a classification of hybrid voters is provided based on result and confidence values, which affect the metrics of availability and safety.Thus, voters are classified into four groups: Independent-diagnostic safety-optimised voters, Integrated-diagnostic safety-optimised voters, Independent-diagnostic availability-optimised voters and Integrated-diagnostic availability-optimised voters. The properties of each category are explained and sample versions of each class as well as their possible application areas are discussed.
Keywords: Ultra-Available System, Smart Sensor, Fault Masking, Triple Modular Redundancy.

Resumen
Este estudio es una primer aproximación para la integración de algoritmos de voteo con dispositivos de diagnóstico de fallas. Los algoritmos de voteo son usados para arbitrar entre los resultados de elementos redundantes en sistemas tolerantes a fallas. Los sensores inteligentes son usados para propositos de detección y separación de fallas (FDI) dada la capacidad su capacidad de inteligencia construida. La integración de enmascaramiento de fallas y las estrategias de FDI es necesaria en la construcción de sistemas altamente disponibles y seguros con la capacidad de detección de fallas en línea. Este artículo introduce un rango de algoritmos de voteo los cuales adjudican un resultado entre los resultados generados por los sensores inteligentes en un módulo de redundancia triple. Las técnicas para integrar los sensores inteligentes replicados y la aproximación de enmascaramiento de fallas son revisadas en este artículo. Una clasificación de algoritmos de voteo híbrido es provista con base en el resultado y los valores de confianza los cuales afectan las métricas de disponibilidad y seguridad de estos algoritmos. De hecho los algoritmos de voteo son clasificados en cuatro grupos: Diagnóstico-Independiente con seguridad-optimizada, Diagnóstico-Integrado con seguridad-optimizada, Diagnóstico-Independiente con disponibilidad-opitimizada y Diagnóstico-Integrado con disponibilidad-optimizada. Las propiedades de cada categoria son revisadas asi como muestras de sus implementaciones son discutidas.
Palabras clave: Sistemas con Alta Disponibilidad, Sensores Inteligentes, Enmascaramiento de Fallas, Redundancia Modular riple.

1 Introduction

Safety-critical distributed embedded systems must achieve stringent reliability and safety goals. A relevant aspect of these systems is the validation of information obtained from interfaces with the external environment, such as FDI. The application of Fault Detection and Isolation techniques has led to the development of self-diagnosing elements such as smart sensors. Some of these sensors use analytical redundancy techniques and

software models to identify inaccuracies (such as drifts) as they develop over time. Replicated smart sensors overcome some of these problems and tolerate *local* sensor faults. On the other hand, voting algorithms are commonly used to mask the errors arising from such faults.

There has been considerable work on FDI, for example Gertler (1998) and Blanke *et al.* (2003). Others have developed a significant amount of literature on fault masking issues, like for instance, Johnson (1989), Lee *et al.* (1990), Bass (1995), and Latif-Shabgahi (2004a). An interesting approach for fault tolerance and data fusion has been explored by Desovski et al. (2006). However, comparatively very few approches have addressed the specific issue of integrating fault masking and FDI approaches.

This paper introduces a class of voting algorithms which adjudicates the results of fault tolerance to one of the redundant smart sensors, using confidence levels. This class is called '*hybrid voters'* in this work. Hybrid voters in the sense that the voting algorithm is combined with smart sensors in order to increment performance. How this integration takes place is illustrated in this paper. A taxonomy of voters presented by Latif-Shabgahi et al. (2004b) mention that these voters improve some aspects of the system dependability performance (e.g. availability and safety) over the use of either fault-masking or smart sensor alone. The approach is similar to N-Self checking programming (Laprie, 1995) in software fault tolerance area.

Key contributions made in the present paper to the state of the art in the field of fault tolerance include:

- Techniques to integrate FDI and fault masking approaches; and
- Introducing a class of novel voting algorithms to capture the smartness information of its inputs.

The experimental evaluation of hybrid voters is presented afterwards. The behaviour of voters is investigated in producing correct, incorrect, and benign results under different fault conditions within a specially implemented experimental framework. Empirical results (not shown here) has shown that those hybrid voters (in which the smartness information of redundant sensors is used in all voting cycles) give higher safety and availability performance than those voters which use this information in some voting cycles. Moreover, integration of result and confidence values of redundant sensors provides higher availability than processing either smartness information or the sensor result values alone.

The novelty of this work is the integration of a confidence value into the definition of the voter in order to enhance reliability and availability during fault scenarios. This is stated in section 3 and highlighted in section 5.

The organisation of this article is as follows: related works on voting algorithms as well as on smart sensor model used in this paper are presented in section 2; approaches to arbitration of TMR configuration of smart sensors are discussed in section 3; section 4 presents the implementation of smart sensors; section 5 shows the experimental methodology; section 6 presents a review of smart sensor behaviour under fault scenarios; section 7 presents some experimental results; and, finally, some conclusions are presented in section 8.

2 Related Work

2.1. Voting Algorithms

Several well known voting algorithms have been widely used in commercial applications:

- The majority voter produces a correct result if the majority of its inputs match (agree with each other). In cases of no majority, the voter generates an exception code which can be detected by the system supervisor to move the system towards a safe state.
- The median voter is a mid-value selection algorithm. This algorithm successively eliminates pairs of outlying values until a single result remains (the algorithm assumes an odd number of redundant inputs).
- The weighted average voter calculates the weighted mean of its redundant input values. Weight values can be determined in various methods; see for instance, (Broen, 1975), and (Latif, 1999). Calculated weights, w_i, are then used to compute the voter output, $z = \sum w_i . x_i \ / \ \sum w_i$, where x_i values are the voter inputs and z is the voter output. An example of this type of voters is the distance metric based weighted average voter, in which the weight values are dynamically calculated based on the distances between the voter inputs in each voting cycle. A voter input, which is far away from the other inputs, is assigned to a smaller weight compared with a voter input that is close to any of the other inputs. Thus, the algorithm does not select a value from the voter inputs, but instead, produces a new result.

The majority, median and weighted average algorithms have been generalised to a **N**-Modular Redundant (NMR) system by Lorczak *et al.*, (1989). NMR is the term to describe redundancy as a way to obtain fault masking between Nequal measures.

- Step-wise negotiated voting has been developed for a long-life space application (Kanekawa, *et al.*, 1989). Diagnosis information is obtained from periodic self-test operations. However, the self-test information is used to supplement voting in cases of disagreement voting cycles. In this type of voters the diagnostic information is given priority, i.e. in cases of disagreement the self-test results are used to select a result. It has been shown that step-wise negotiated voting offers improvements over independent majority voting or stand-by redundancy technique. The problem of exact (or bit-by-bit) voting on vectors of information has been investigated by Gersting, *et al.* (1991). A key issue is the resolution of conflicting agreements between the different fields within a set of vectors. The solutions proposed include the calculation of a composite vector rather than selection of any of the input vectors and the use of application specific weightings, to minimise the importance of information in less important vector fields.

The voters described in this paper integrate diagnostic (rather than self-test) and result information of redundant sensors for adjudication. The algorithms use inexact rather than exact voting because small variations between redundant sensor result values are expected. A good survey on voting algorithms can be found in (Latif, *et al.*, 2003).

2.2. Smart Sensors

Smart sensors are those devices in which the sensors and circuits co-exist, and their relationship with each other and with higher-level processing layers goes beyond the meaning of transduction. However, the concept of smart sensing (i.e. sensor information processing without redundant and unnecessary data acquisition, and with at-sensor-level processing) is relatively new. Smart sensors are information sensors, not transducers and signal processing elements; they are used to enhance product quality, plant efficiency and availability, and measurement quality in feedback control (Henry, *et al.*, 1993).

The need for non-expensive, high-performance, and high-reliable smart sensors has been sufficiently disscussed in the literature (see for example (Henry, *et al.*, 1991), (Nguyen, *et al.*, 1996), (Meulen, 2004) and (Benítez-Pérez, *et al.*, 1997)) for industrial production machines, consumer applications and in space systems such as launchers and sattelites.

A typical self-diagnosis scheme for a smart sensor is shown in Fig. 1.

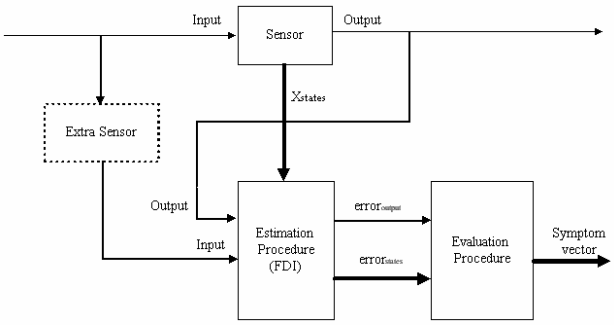

Fig. 1 Smart Sensor Implementation

This model-based implementation uses an observer and an extended Kalman filter within the Estimation Procedure (FDI) block, in order to estimate the output and the states of the sensor. An error is produced from the difference between the current values (output and states) and their respective estimated values. Having produced the error *value*, an evaluation procedure is performed in order to produce a symptom *value*.

The block diagram of the smart sensor used in this work is shown in Fig. 2A. The smart sensor produces two outputs: the result value *x*, which represents the current output of the sensor and a confidence value *J*, which provides a measure from interval [0…1] regarding the confidence of the result value *x*. Hence, *J* indicates a *correctness level* of the sensor result *x* output. This means that a result value x_1, supported by a confidence value $J = 0.99$, is considered more acceptable (reliable) than a result x_2 supported by the confidence value $J = 0.8$. Notice that, by definition, the *J* confidence value cannot be a negative number.

In spite of their rapid progress, smart sensors could not be provided with sufficient computation power to perform important functions such as calibration, compensation, digital filtering, programmed self-testing, and strong interface to bus systems (Huijsing, 1992). These issues make questionable the use of smart sensors in

ultra-high available systems. A possible solution for their application to ultra-high available systems is the use of redundant smart sensors in a TMR configuration. This has the benefits of both masking and FDI (intelligent) techniques. Such an approach is shown in Fig. 2B. Notice that increasing the entire system availability is the primary philosophy of integrating masking algorithms and smart sensors.

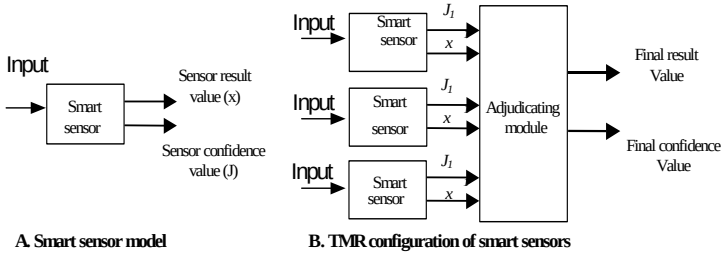

Fig. 2 Integration of smart sensor and adjudicating modules

By considering a smart sensor as a 2-output component, there are four possible cases regarding the metrics of availability and safety of its result value *x* and confidence value *J*:

- **Type A:** 'result value *x*' is correct and 'confidence value *J*' is produced correctly. This case is a Fault Free scenario.
- **Type B:** 'result value *x*' is incorrect and 'confidence value *J*' is produced correctly. In this case the value of *J* detects the incorrect result value and thus, the system safety is preserved but its availability is decreased.
- **Type C:** 'result value *x*' is correct but 'confidence value *J*' is produced incorrectly due to the failure of diagnosing module of smart sensor. In this case, the correct result value *x* may be interpreted as a wrong result by system supervisor and is discarded. Thus, the system availability is threatened in this case.
- **Type D:** 'result value *x*' is incorrect and 'confidence value *J*' is also computed incorrectly. In this case the wrong result *x* may be interpreted as a correct answer and is propagated within the system. This situation threatens the system safety.

In this study, like other research works (suh as Buskens, *et al.,* 1993), it is assumed that the diagnostic module of the smart sensor operates correctly, and *J* is always produced correctly. Therefore, integration methods for cases where the possibility of failure of sensor diagnosing module is taken into account (cases C and D) are not considered in this article.

3 Classification of Hybrid Voting Algorithms

The novelty presented in this paper is the proposal of the hybrid voter. A hybrid voter receives result values *x*, as well as confidence values *J* from redundant sensors to produce final outputs. Several approaches are taken to arbitrate between smart sensors. We classify hybrid voters into four categories based on the following criteria:

- How the result and confidence values are handled by the voter (independently or by integration)?
- What is the output domain of the voter? Whether it produces always an output (correct or incorrect) or it has potential to generate an exception flag in some (e.g., complete disagreement) voting cycles (thus gives correct, incorrect and benign outputs). The former is called a *two-phase voter* and the latter is named a *three-phase voter*.

The second criterion requires a further justification. Generally, voting algorithms can be divided into two groups (based on their capability to produce a benign error): majority-type voting algorithms (three-phase voter) and average-type voting algorithms (two-phase voter). Majority-type voters produce a result if there is an agreement between a small group of redundant inputs (the number of inputs to be in agreement differs in various voting strategies). In cases of disagreement, these voters can produce a benign output (an exception flag which moves the system toward a safe state). Examples of this type are majority, plurality, predictor, and maximum likelihood voters.

On the other hand, average-type voters always produce an output regardless of being agreement or disagreement between redundant inputs. They have no capability to generate benign outputs. These voters either

integrate the redundant inputs or select some of them to give the voter output. Examples of this category are average, mean selector, and weighted average voting algorithms.

The authors have applied the same classification method to hybrid voters. Majority-type hybrid voters have been called 'three-phase' and average-type voters are named 'two-phase' voters. Table 1 summarises this classification in which four types of hybrid voters are distinguished. They are explained in following subsections.

Table 1. Classification of Hybrid voters

Processing of Output inputs space	**Independent**	**Integrated**
Three phase	*Independent-diagnostic safety-optimised* (Type A)	*Integrated-diagnostic safety-optimised* (Type D)
Two phase	*Independent-diagnostic availability-optimised* (Type B)	*Integrated-diagnostic availability-optimised* (Type C)

Independent and integrated diagnostics is referred to as how the voter takes confidence value. A general representation of the algorithm for classification is shown in Fig. 3.

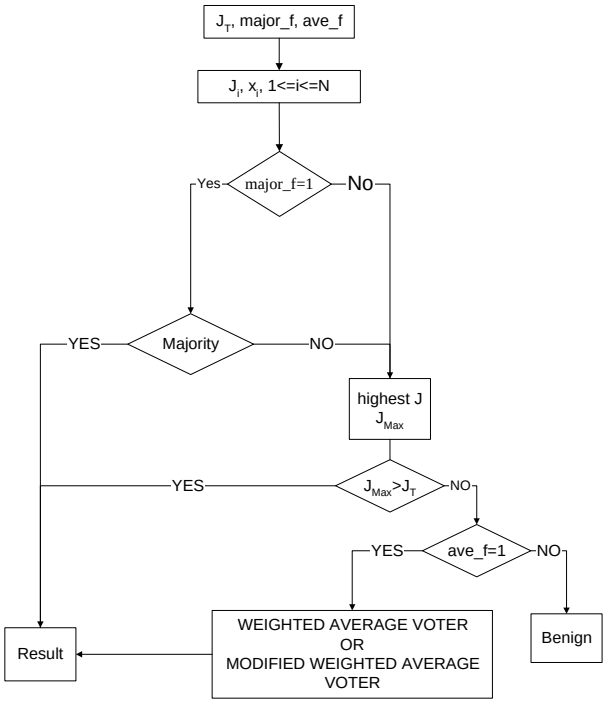

Fig. 3. Flow chart of Voter Selection

The variables J_T, *major_f*, *ave_f*, *N*, J_*, and x_* need to be defined, in order to choose one of the algorithms:

- J_T is the threshold value for comparison between as selected confidence value *J* and its own.
- *major_f* and *ave_f* are flags defined by the user in order to choose which algorithm is selected.
- N is the number of smart elements involved in this scenario. J_* is the confidence values and x_* is the output related to smart sensors.

Most of the cases can be regarded as special cases of two types of voters. However, as discussed in the following sections, these variations produce different results, highlighting the need to independently study them.

3.1. Independent diagnostic, safety-optimised voters (Type A voters)

This group of voters produces a result by examining the validity of result values and/or confidence values of smart sensors (Three-phase voter type). For instance, if there is a majority between the result value of sensors, or if the confidence value of a subset of sensors meet an application specific threshold value, the voter produces an answer, otherwise it generates an exception flag which moves the system toward a safe state such as fail-safe or a fail-stop mode (the voter output, in this case, is called *benign error*). The voter output may be incorrect in some cases due to improper selection of threshold values. As a result, the output set of this group of voters includes *correct*, *incorrect* and *benign* results. Two voters in this category are described below.

A.1. Majority supplemented with conditional maximum J selector voter

This voter uses the result value of smart sensors to produce the voter output in agreement cases. In cases of disagreement between sensor result values, the voter selects the result value of the sensor with the highest confidence value, J_{max}, as the more likely acceptable answer of the present voting cycle. If $J_{max} > J_T$ where J_T is an application specific threshold, the selected result value is regarded as a correct value and is sent to the voter output, otherwise an exception flag is generated. Note that if the value of J_T is selected improperly the voter output may be incorrect in some cases. According to Fig. 3, in this case J_T is a value between 0 and 1, *major_f*=1 and *ave_f*=0.

A.2. Conditional maximum J selector voter

In this voter, the result value with the highest confidence value, $J_{max} = max\{J_1, \ldots, J_N\}$, is selected as voter output if confidence value is greater than a given threshold J_T. When $J_{max} < J_T$ the candidate result value is regarded as incorrect value and the voter generates an exception flag. Here, the voting function is performed upon the diagnostic values and a sensor result is selected based on the "best sensor" as determined by the self-diagnosis subsystem. According to Fig. 3, J_T is bounded to $0 \leq J_T \leq 1$, *major_f*=0 and *ave_f*=0.

3.2. Independent diagnostic, availability-optimised voters (Type B voters)

This group covers those two-phase voters that are, in fact, selection procedures rather voting algorithms. Hence, the output set of this group of voters includes *correct* and *incorrect* results. Three voters of this group are explained below.

B.1. Maximum J selector voter

This voter selects the result value of a sensor with the highest confidence value and masks the other two sensor result values. In fact, the voter is a selector routine. The major drawback of this voter is its potential to produce low reliable results in cases of all faulty sensors. In such cases, even the result value of a sensor with the highest confidence value is an unreliable value. This configuration has the ability to mask coincident result value faults.
In addition, the diagnostic output of the best sensor is passed to the diagnosis management system. Now the rest of the system is aware of the correctness level of voter output by means of J_{max}, but it fails to perform an action in cases that J_{max} has an unacceptable value since the output has already been propagated into the rest of the system (it may cause a catastrophic failure). It is also possible to use the sensor confidence values within the diagnosis management system for detection or reconfiguration of faulty sensors.
Following Fig. 3.1 description, this voter is shown as J_T=0, *major_f*=0 and *ave_f*=0.

B.2. Majority and Maximum J selector voter

This voter behaves as the standard majority voter in cases of agreement between result values. It selects the result value with the highest associated confidence value as the voter output in cases of disagreement from the majority voter.

Following Fig. 3 this voter is chosen by J_T=0, *major_f*=1 and *ave_f*=0.

B.3. Conditional maximum J selector and weighted averaging voter

The result value with the highest confidence value, $J_{max} = max\ \{J_1,\ldots, J_N\}$, is selected as the voter output if J_{max} is greater than threshold J_T. Where $J_{max} < J_T$, the weighted average of result values is calculated as the voter output.

Following Fig. 3, this voter is selected by J_T bounded as $0 \leq J_T \leq 1$, *major_f*=0 and *ave_f*=1. In this case weighted average voter is selected.

3.3. Integrated diagnostic, availability-optimised voters (Type C voters)

Integrated-diagnostic availability-optimised voters integrate the result and confidence values of redundant smart sensors to produce a single voter output. A modified version of distance metric-based weighted average voter (Lorczak *et al.,* 1989) is explained.

Modified weighted average voter

This voter uses both the confidence and result values of sensors. It seeks to arbitrate where a conflicting choice of element emerges. Here, a hybrid arbitrator is presented that integrates the sensor confidence and result values to compute the voter result.

A weighting procedure, which is applied to each sensor result, is proportional to its associated confidence value. Thus, results with high confidence value receive a large weighting factor while those with lower confidence value receive a small weighting value. The output x_o for weighted average voter is given by:

$$x_o = \sum_{i=1}^{N} \left(\frac{w_i}{s}\right).x_i$$

Where

$$w_i = J_i^2$$

Similar to the least squared criterion, this squared metric increases the sensitivity of weighting values for diagnostic value of sensors, that is, a sensor result value with a higher confidence value receives a high weighting factor and a sensor result with a lower confidence value is assigned to a very small weighting value.
Based upon Fig3, this voter is selected by J_T bounded as $0 \leq J_T \leq 1$, *major_f*=0 and *ave_f*=1. In this case modified weighted average voter is selected.

3.4. Integrated diagnostic, safety-optimised voters (Type D voters)

Type D voters produce an output by integrating redundant result values and confidence values. A voter in this category is described below for which the flow chart is shown in Fig. 4.

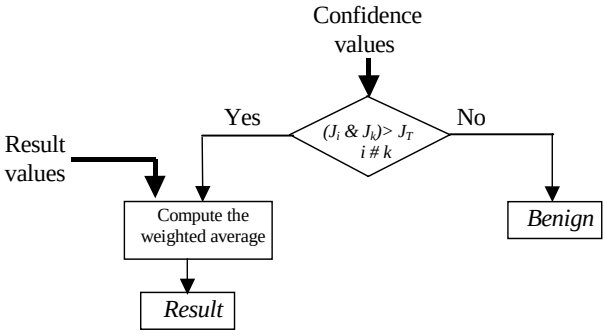

Fig. 4. An example of "Integrated diagnostic, safety-optimised" voters

The voter examines the validity of the confidence values by means of a predefined application-specific threshold to ensure that at least two-out-of-three sensors function correctly and produce acceptable result values. Where 2 confidence values satisfy the threshold J_T, the weighted average of result values is computed and sent to the voter output; otherwise, a benign error is generated.

Although the voter algorithm is not modified it produces a result only when *J* reach an agreement.

4 Implementation of smart Sensor

Following smart sensor definition, this element proposes the evaluation of the current condition of the sensor based on either the states or the parameters available from the model. In order to obtain this information, it is necessary to evaluate the unit from various sources. Different approaches may be followed to the implementation of smart sensors, such as Self-Validation (SEVA) scheme (Yang, 1993) or Analytical Redundancy. For the present work the second approach is used as shown in Fig. 5. This type of transducer measures pressure in a static form. The dynamics of the case study is basically linear.

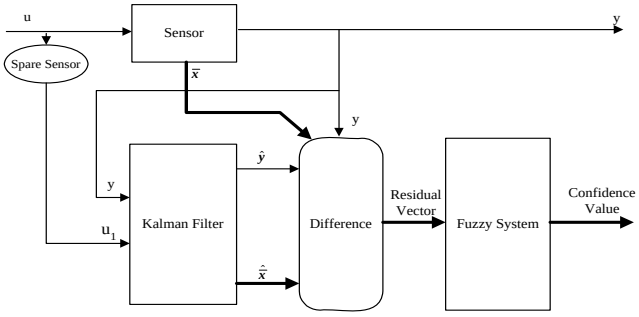

Fig. 5 smart Sensor Implementation

Where *u* is current input, *x* is the current output, *s* is the current state vector, $\hat{s}$ is the estimated state vector, u_1 is the current input after been monitored by spare sensor.

A Kalman filter generates estimation for measurements related to the current element. The evaluation of the residual vector is performed by using a fuzzy system. The fuzzy system produces a confidence value explained in (Benítez-Pérez *et al*, 1998). An extended review of dynamical model is given in the same paper.

Confidence value represents the evaluation result of the entire element. For this work, Confidence Value (*J*) is used as the performance measure of smart sensor. The range of this signal is between 0 and 1. The value *J*=0 represents a faulty sensor and the value *J*=1 denotes a correctly operating smart sensor. Fig. 6 and Fig. 7 show a fault-free and fault scenario respectively. Fig. 6.a shows the residuals with respect to the states and the output for a fault-free scenario. Fig. 6.b shows the response of confidence value generated from the fault-free scenario shown previously.

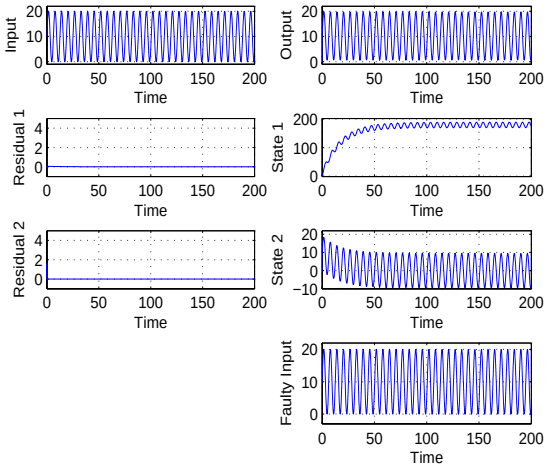

Fig. 6. a Fault-Free Scenario

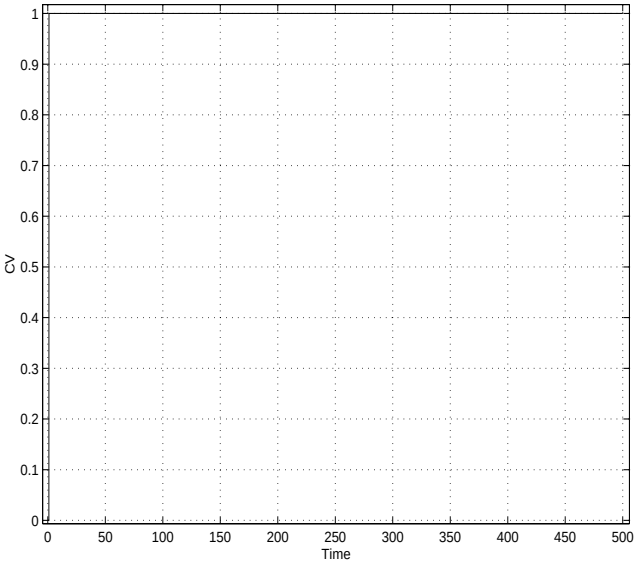

Fig. 6.b Fault-Free Scenario Modified SEVA Response

Fig. 7.a shows the response of the smart sensor for a fault scenario of noise. Fig. 7.b shows the related response of the symptom vector.

For Fig. 7.a noise signal as fault is injected at 20 seconds until 100 seconds. The amplitude value is around 10 % grater than the current input when this faulty signal is injected, and there is a reaction shown by the residuals of the sensor. Fig. 7.a presents how the three residuals respond during time period of the fault. The response of the smart sensor is presented in Fig. 7.b. In here, *J* suffers a degradation proportional to the effects of the fault into the sensor. *J* shows a degradation of 99 % at 40 seconds until 100 seconds. The meaning of this result is that the confidence of the element is 1%. For the purpose of this research confidence value is used as health measure of the smart sensor. This is due to it represents the behaviour of the whole element (dynamics and current output).

These Figs. illustrate how confidence value suffers degradation in the presence of a fault (compare Fig. 6.b and Fig. 7.b). The measure *J* is used later in this paper to integrate the smart sensor with different types of voters. This value has been selected as the measurement of the "health" of the sensor.

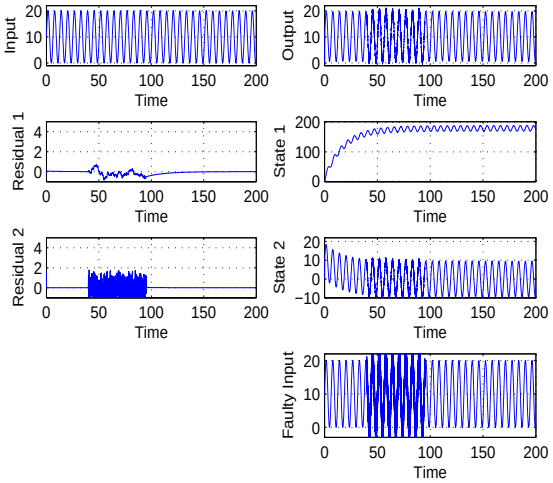

Fig. 7.a Fault Scenario (Noise, Amplitude 5)

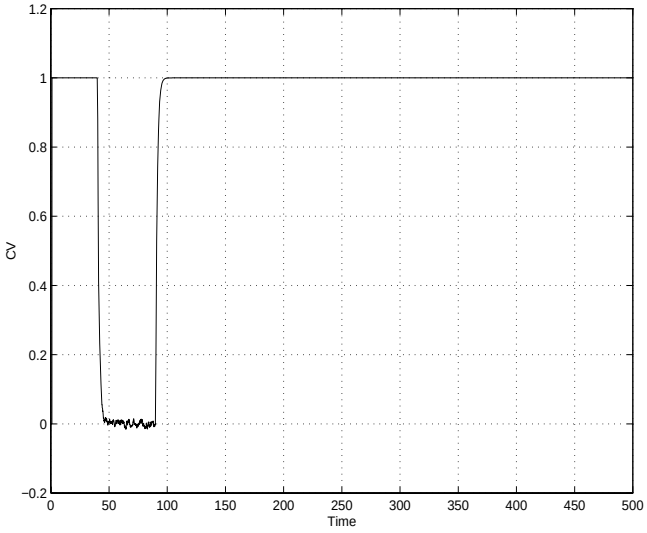

Fig. 7.b Fault Scenario (Noise, Amplitude 5)

5 Experimental Methodology

To evaluate the hybrid voters a special test harness with fault injection capability has been developed. Fig. 8 indicates the general schematic diagram of this implementation. It is used to examine the smart sensor, and the fault masking mechanisms.

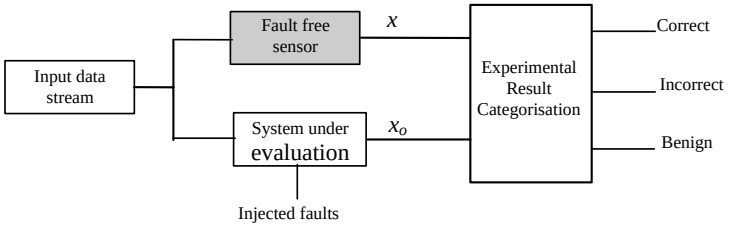

Fig. 8. Test Harness

A stream of input data with sinusoidal trajectory and adjustable arrival rate feeds both the system under evaluation (Triple Modular Redundancy (TMR) configuration of non-smart sensors, or TMR configuration of smart sensors or a smart sensor) and the notional fault-free sensor for which it is assumed that the output x is always correct. Random faults are injected into the system under test by special software fault injection tool. The output of the system, x_o, as well as the output of fault-free sensor is presented to a result categorisation module. Based on the numerical distance between x and x_o values, the result categorisation module interprets the output of the system under evaluation in each voting cycle as correct, incorrect, or benign output. It uses an application-specific threshold value, T, in this comparison. It is more reasonable to define the threshold value T as a percentage of the full-range of input trajectory, i.e., *T = 5% * (full range input).* This has been chosen in order to have a very sensitive decision making strategy. If this value is chosen higher it becomes difficult to separate between scenarios. For injected faults the time of injection, persistence period of faults and probability distribution of faults are important. In this work, the following assumptions are made:

- faults have a Gaussian distribution;
- faults are injected in each voting cycle;
- the amplitude of faults is adjustable from interval *[-a +a]* where *|a| <75% of input peak value*; and
- faults from interval [0 5] (less than 25% of the maximum input value) are regarded as small faults, from interval [5 10] (25% - 50% of the maximum input value) are assumed as medium faults, and from interval [10 15] (greater than 75% of the maximum input value) input value are called large faults.

5.1 Empirical method of smart Sensor

In this case, the system under evaluation is a single smart sensor. The intelligence information of a smart sensor (which diagnosis its behaviour) taken from the symptom vector is not used by the result categorisation module (Fig. 8). Therefore, the evaluation is specifically related to the model sensor rather than the response of the symptom vector. In each voting cycle the sensor output x_o is compared with the notional correct output x. Where the difference of these values is less than a predefined application-specific threshold T (In this work $T = 2.5\%$ is selected), the smart sensor output regarded as a correct value, otherwise, it is taken as an incorrect answer. Table 2 summarises these arguments. Fig. 9 indicates the fault injection method used to perturb a smart sensor.

Table 2. Output Classification

Condition	Interpreted output
$\|x - x_o\| < T$	Correct
$\|x - x_o\| >= T$	Incorrect

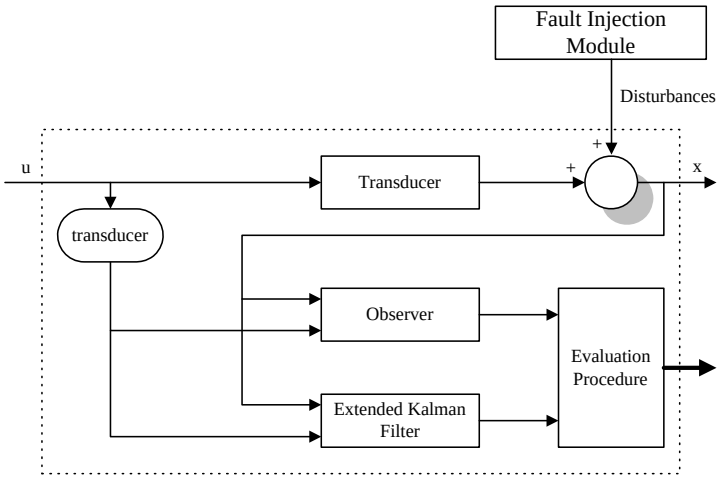

Fig. 9 Fault Injection Point

5.2 Empirical Method of Integrated Voters

Fig. 10 indicates the implemented test harness for evaluation of integrated voters, functioning in a TMR system.

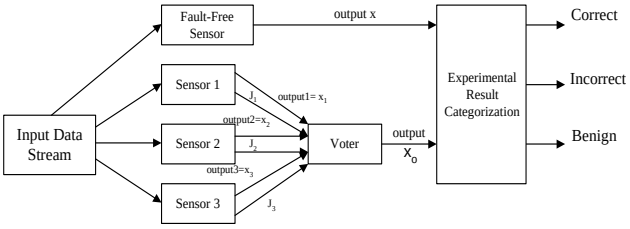

Fig. 10. Implemented Test Harness

For type A voters the output of the TMR system, x_o, is interpreted by the result categorisation module based on Table 3. This table has been constructed by detailed analysis of type A voters which either produce an output (*correct* or *incorrect*) or cease to generate an output (*benign* result). Note that the intelligence information of smart sensors has been considered within the voting algorithm.

Since J is considered fault free J_T is chosen as a value from $0.5<J_T<1$. In order to increase the number of valid values J_T is chosen close to 0.75. This value is experimental based and the only recommendation is the level of trustiness from J results.

Table 3. Output Classification of Type A Voter

Condition	Interpreted output
$\|x - x_o\| < T$ AND [agreement on result values OR $max\{j_1, j_2, j_3\} >= j_3$]	Correct
$\|x - x_o\| >= T$ AND [agreement on result values OR $max\{j_1, j_2, j_3\} >= j_3$]	Incorrect
Disagreement on Result values OR max $\{j_1, j_2, j_3\} < j_3$]	Bening

For type B and C voters which always produce an output, the output value is interpreted by the result categorisation module based on Table 4.

Table 4. Output Classifications of Types B and C

Condition	Interpreted output
$\|x - x_o\| < T$	Correct
$\|x - x_o\| >= T$	Incorrect

For each voter, the results of n system run (n is selected 10^4) are classified based on tables 3 and 4. In this way, n_c correct results, n_{ic} incorrect outputs, and n_b benign results are collected. It is obvious that $n_c + n_{ic} + n_b = n$ and for type B and C voters $n_b = 0$. These data are, then, used for evaluation and comparison of selected integrated voters. Three performance measures are defined for this propose: voter availability, voter safety, and voter error/fault detection capability. The number of correct results is taken as a measure of voter availability, A. The number of incorrect results are taken as a measure of voter safety, S, and the number of benign results are taken as a measure of voter error/fault detection capability, Fault Detection (FD), Thus, $A = f(n_c)$, $S = f(1/n_{ic})$ and $FD = f(n_b)$. For example, among two voters, the one with a smaller number of incorrect results is taken as a more safe voter.

6 Smart Sensor Behaviour Under Fault Scenarios

To examine the behaviour of a smart sensor in different fault conditions and to ensure that the confidence measure *J* reflects correctly the sensor functionality, three smart sensors are examined in various fault conditions for a sinusoidal input trajectory. The first sensor is perturbed by random faults with small amplitudes in each run, the second sensor is perturbed with medium faults, and the third sensor is perturbed with large errors. Fig. 11 shows their response for 45 runs where sensors are fed with input data within the time period [20 30] with *sample-rate = 0.1*.

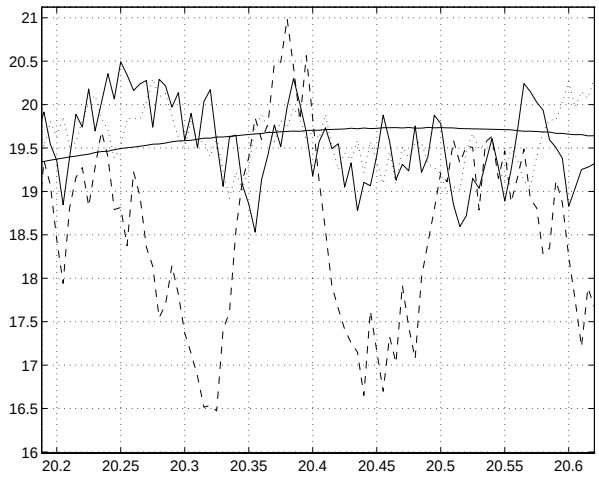

Fig. 11. Response of the Smart Sensor under Fault Conditions

In Fig. 11, the smoothest response belongs to the fault-free sensor. The dotted line represents the response of the first smart sensor, which is perturbed by small faults, the continuous line indicates the response of the second smart sensor perturbed by medium faults, and the dashed line represents the third smart sensor response perturbed by large faults. The plots indicate the correct behaviour of sensors in the presence of faults; a sensor affected by a larger fault, gives an output more far from the expected correct answer.

Fig. 12 shows the *health* (Confidence Value) response of the smart sensors under the same fault conditions presented in Fig. 11. Again, the dotted line shows the response of the first smart sensor, the continuous line represents the second smart sensor, and the dashed line represents the third smart sensor.

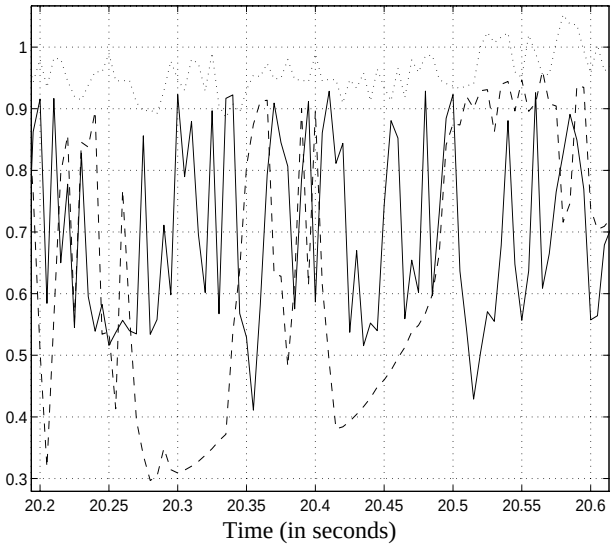

Fig. 12. Response of *J* from the same fault condition of Fig. 11

The response of confidence value from the smart sensors demonstrates how accurate the output value is with respect to each raw response. For instance, at time 20.33 seconds the confidence measure of the third smart sensor (Fig. 12, dashed line) has a value of 0.36, whereas, the second smart sensor has a value of 0.91 and the first smart sensor gives value 0.89. Alternatively, at the same time, Fig. 11 shows that the third smart sensor has the worst response (dashed line). Meanwhile, the second smart sensor has the best response (continuous line) and the first smart sensor presents a medium response (dotted line).

This empirical explanation gives the basis of how the system chooses the best response by using of intelligence information of sensors.

Having defined the experimental test-harness and sensor behaviour in a faulty environment, the results of the smart sensor and integrated voters are presented. In subsection 6.1 the experimental results of a smart sensor is presented. In subsection 6.2 the behaviour of integrated voters are studied in order to highlight the benefits of integration of FDI and fault masking features.

6.1 Experimental Results for Smart Sensors

Figure 13 indicates the number of correct outputs of a smart sensor for different fault scenarios, each comprising 10^4 runs. As expected, by increasing fault amplitude the number of correct results decreases. However, there is a fast decrease in gradient of the number of correct results followed by increasing fault amplitude from interval [0 5]. The gradient of decreasing the number of correct outputs for large faults becomes slower than that with small and medium faults. These results demonstrate that integration of fault masking and smart sensors must increase the system availability for small and medium faults. In section 5.2 it is shown that the use of sensor smartness value in the integrated system (fault masking and FDI systems) improves the availability of the response for small faults and more drastically for medium faults. Note that the number of correct results of a system has been chosen as a measure of its availability.

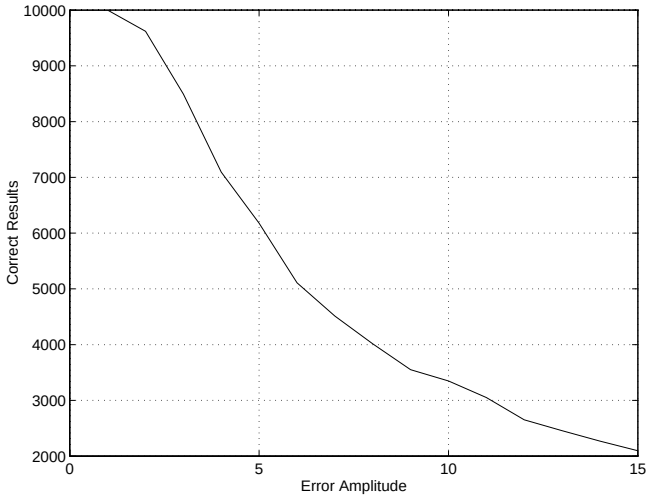

Fig. 13. Correct Results from a Smart Sensor

Fig. 14 shows the plot of incorrect results of a smart sensor versus fault amplitude. There is a peculiar response for faults of interval [0,…, 5], which is an increment about 12% of incorrect results for every scenario. The main goal of the use of smart sensors with a fault masking approach is to reduce this increment in a reasonable manner. This is addressed in the following section.

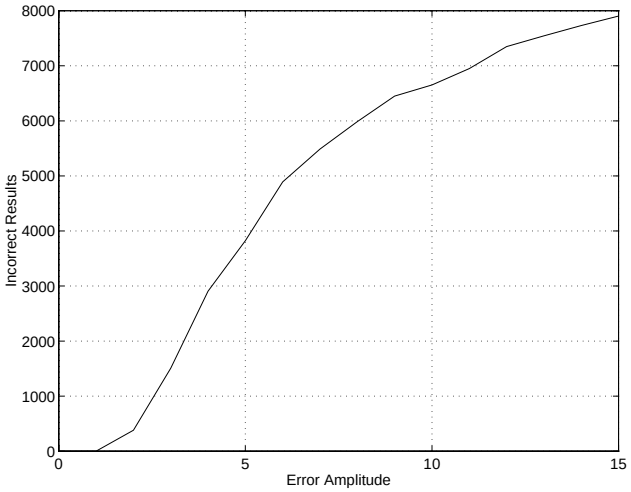

Fig. 14. Incorrect Results from a Smart Sensor

6.2 Experimental Results of Hybrid Voters

6.2.1. Experimental Results for Type A Voters

Figs. 15, 16, and 17 indicate the number of correct, incorrect, and benign results of Type A voters versus error amplitude. In each figure, the plot of type A voters is compared with that of the standard majority voter as the basis algorithm for voters of this category. The *conditional maximum J selector* voter gives the largest number of correct outputs among type A voters. Therefore, it is the most available voter in this group. On the other hand, the majority voter produces the smallest number of correct outputs; hence it is the least available voter of type A voters. The *majority supplemented with conditional maximum J selector* voter has a compromise availability between the standard majority and *conditional maximum J selector* voters.

From the safety viewpoint (see Fig. 16), the *conditional maximum J selector* voter is also superior to the other two voters. The *majority supplemented with conditional maximum J selector* voter is the least safe voter of type A voters, since it gives a huge number of incorrect outputs. Finally, according to Fig. 17 the fault detection capability of majority voter is higher than that of the other two voters. Similar to its safety, the *majority supplemented with conditional maximum J selector* has the lowest fault detection property. There is one peculiarity related to Fig. 16, where at fault amplitude of 12 *Conditional Maximum, J Selector* suffers an abrupt increment in the number of incorrect results. This behaviour is related to the unreliable result of this voter beyond fault amplitude of 8.

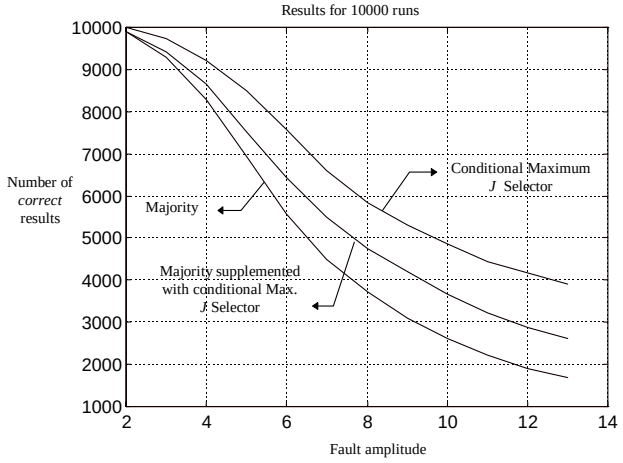

Fig. 15. Correct Results for Type A voters

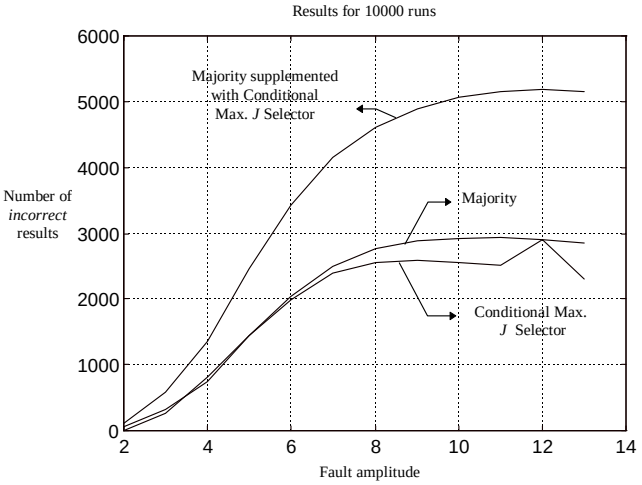

Fig. 16. Incorrect Results for Type A voters

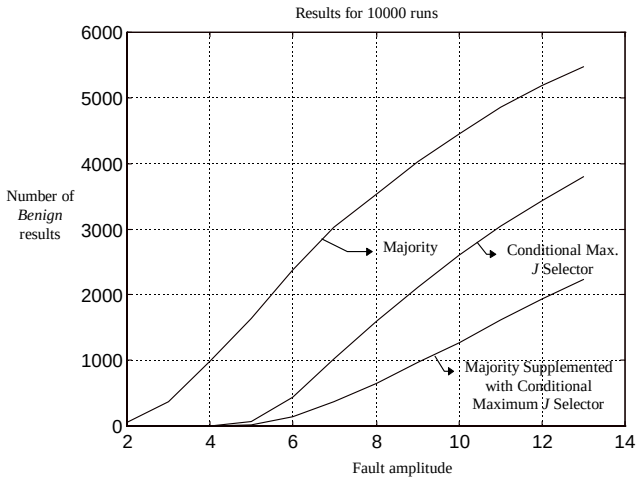

Fig. 17. Benign Results for Type A voters

We now study, in detail, the effects of using confidence value of redundant smart sensors in voting process. For this purpose, we consider the results of the above mentioned type A voters at error point 10 (where error amplitude = 10). Table 5 shows these results, which are extracted from figures 15 to 17. For simplicity, we have used the following abbreviations in this table and afterward: 'Maj' for majority voter, 'MCMJ' for majority supplemented with conditional maximum *J* selector voter and 'CMJ' for conditional maximum *J* selector voter.

Table 5. Results of Type *A* voters at Error-amplitude=10

Voter	n_b	n_c	n_{ic}
Maj.	4458	2621	2921
MCMJ	1269	3657	5074
CMJ	2600	4846	2554

Recall that Maj voter does not use smartness information of redundant sensors, MCMJ voter uses this information in disagreement voting cycles and CMJ voter use this data in all voting cases. We first concentrate on the results of Maj and MCMJ voters.

The table shows that MCMJ gives 3189 less benign errors than the Maj voter. This means that it has ability to generate 3189 more voter outputs than the Maj voter. Of these outputs, only 1036 output (30%) are correct and the remaining 2153 outputs (70%) are incorrect. In other words by using the smartness information of sensors in disagreement cases of the standard majority voter, the voter availability ($A \sim n_c$) slightly improves in the price of considerable decreasing of voter safety as well as its fault detection capability. Therefore, this strategy is not suitable to be taken in those voting used in highly safe systems.

Similarly, by comparing the results of Maj and CMJ voters it can be concluded that where smartness information of sensors is used in all voting cycles (as used in CMJ), both the voter availability and safety increase.

It may be argued that the performance of MCMJ and CMJ voters depend on the value of J_T (used within these voters); a lower J_T, more correct outputs are produced. This is not true, since in reality we need voter outputs supported with higher level of confidence values; otherwise, the system safety will be threatened.

6.2.2. Experimental results of Type B voters

Figs. 18 and 19 show the plot of the number of correct and incorrect results of Type B voters versus error amplitude, respectively. From this group, 'Maximum *J* selector' voter gives the best response (the largest number of correct and the smallest number of incorrect outputs) for all fault scenarios. *Majority and maximum J selector* voter has the worst performance and the *Conditional maximum J selector and weighted average* voter gives a moderate behavior between those two. For small faults the response of the *maximum J selector* voter is quite similar to the *conditional maximum J selector and weighted average* voter. However, for large faults *Maximum J selector* voter gives better results. Comparing Figs. 18 and 19 shows that the availability level of type B voters is higher than that of the standard majority voter. Thus type B voters are availability-optimized voters.

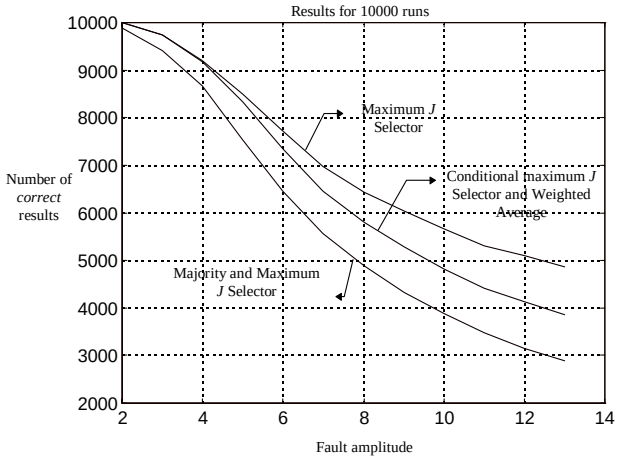

Fig. 18. Correct Results for Type B Voters

Comparing type B voters arises another important issue. *Maximum J selector* voter, as the simplest voter from the viewpoint of functionality and structure, gives the best results among type B voters. This means that supplementing additional features and mechanisms into this voter (as being done in the *conditional maximum J selector and weighted average* voter) or using the voter in disagreement cycles of standard majority voter (as being done in the *majority and maximum J selector* voter) not only does not improve the voter performance but also increases its complexity, which turns threatens the system safety and availability performance. Similar to Type A voters, a Type B voter which uses the confidence information of smart sensors in all sampling periods gives higher performance than the ones which process this information in some of the sampling periods (e.g., only in disagreement voting cycles).

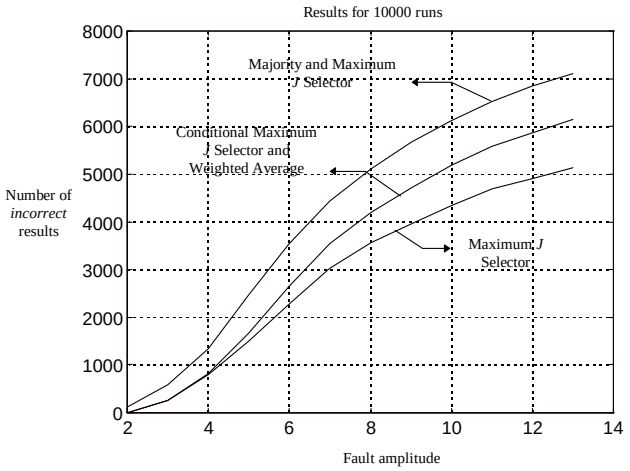

Fig. 19. Incorrect Results for Type B voters

6.2.3. Experimental results of Type C voters

Type C voters integrate the result and confidence values of redundant smart sensors to generate the voter output. In this section the behaviour of modified version of standard weighted average voter (in which the intelligence information of smart sensors is used to calculate weight values as explained in (Benítez-Pérez *et al.*, 1999)) and the distance metric-based weighted average voter (Lorczack *et al.*, 1989) is studied. Figs. 20 and 21 indicate the number of correct and incorrect results of these two voters versus fault amplitude. Fig. 20 shows that the 'modified weighted average' voter gives more correct results than the standard weighted average voter; therefore, it has higher availability.

These results present a clear advantage of the use of confidence value integrated to the modified weighted average voter over the naive algorithm. The reason is that the weight values follow the behavior of the respective sensors rather than the distance between sensors result values.

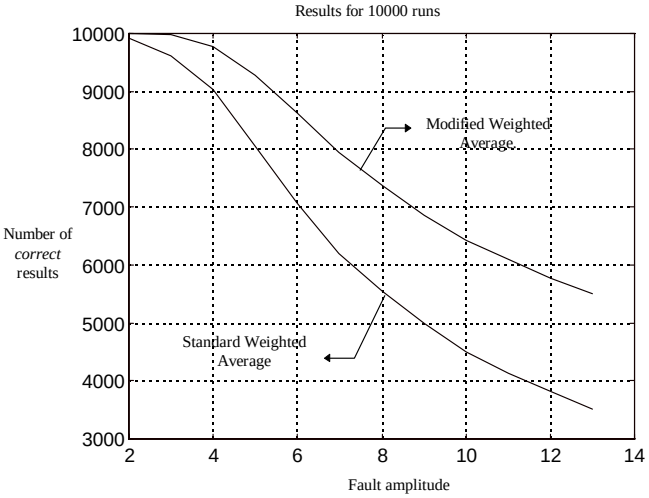

Fig. 20. Correct Results for Type C Voters

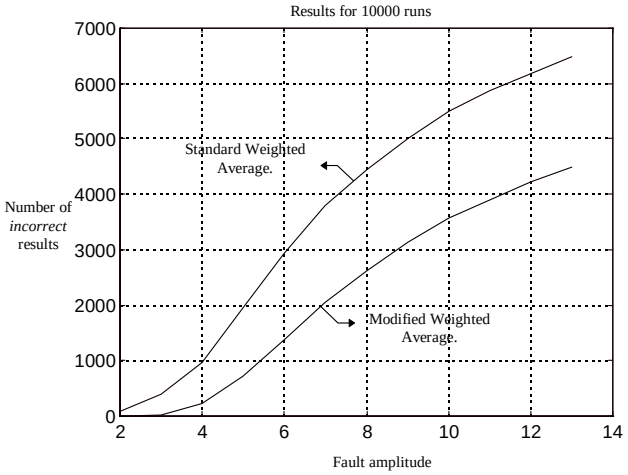

Fig. 21. Incorrect Results for Type C Voters

Comparing of Figs. 20 and 21 shows that the availability level of 'modified weighted average' voter is higher than that of the type A voters. Therefore, it can be regarded as a availability-optimized voter.

The confidence value of the smart sensor is considered as the sensor performance measure. This measure determines the entire behaviour of the smart sensor and gives the certainty level of its result value. Four types of *hybrid voters* to arbitration of smart sensors are proposed and sample voters of each type are introduced. In type A and type B voters both of the result and confidence values of redundant sensors are handled independently. They differ from each other in output domain. Type A voters have potential to produce a benign error in complete disagreement voting cycles whereas type B voters produce always an output. In type C and type D voters the result and confidence values are integrated to produce a single voter output. Since producing the voter output, the certainty level of each sensor result is taking into account, it is expected that (in a NMR system) the use of confidence value of sensors to arbitrate between their result value will improve the entire system availability or safety. It has been shown that those hybrid voters such as 'modified weighted average' and 'maximum J selector' voters in which the smartness information of replicated smart sensors is used in all voting cycles give higher safety and availability performance than those voters which use this information in some voting cycles. Moreover, integration of result and confidence values of redundant sensors gives higher availability than processing either smartness information or the sensor result values alone. The selection of one algorithm is an ad-hoc matter based upon empirical study.

Cumulative number of correct results of Type A voters shows that the *conditional maximum J selector* voter gives higher safety performance than the standard majority voter and the *Majority supplemented with conditional maximum J selector* voter produces lower safety than that voter (see Fig. 16). In fact, when the standard majority voter uses the *maximum J selection* mechanism as its output selection method in all voting cycles, its safety performance increases. In contrast, when the majority voter uses that mechanism in some of the voting cycles (e.g., in disagreement cases), its safety considerably decreases. These considerations imply that integrating smart sensors and the standard TMR system improves the basis system safety if and only if the smartness information of redundant sensors is used in all voting cycles, otherwise, the system safety may be decreased. type A voters which function in this way, are called 'safety-optimized' voters in this work. Moreover, the use of smartness information of redundant smart sensors in a TMR system (in all or some of the voting cycles) improves the entire system availability (see Fig. 15) and decreases the system fault-detection capability (see Fig. 17).

Among type B voters, the *maximum J selector* voter gives the highest safety and availability performance than the other two voters (Fig. 18 and Fig. 19). This voter and the *conditional maximum J selector and weighted average* voter which use the confidence values of redundant smart sensors in all voting cycles to produce the voter output, give better performance than the *majority and maximum J selector* voter that uses those information only in disagreement cases. Moreover, the availability of all type B voters is higher than that of the majority voter (basis voter); they are availability-optimized voters. This is not true for their safety performance.

For type C voters, a comparison between *modified-weighted average* and *standard weighted average* voters has concluded that modified weighted average voter has higher performance than the standard weighted average voter due to the use of the smartness information of sensors in all sampling cycles. In this case, the *J* values are integrated to the modified voter as weights. This integration evaluates the smart sensors in terms of their own confidence values rather than differences between their outputs. The availability of modified weighted average voter is higher than that of the majority voter; therefore it is a availability-optimized voter.

Having obtained the results from both type B and type C voters, it is possible to make a comparison between the best two voters. From type C the *modified weighted average* voter has shown the best availability performance and for type B group, the *maximum J selector* voter is the best. The *modified weighted average* gives higher availability than the 'maximum *J* selector' voter for any fault amplitude. For instance, in fault amplitude 5, *modified weighted average* has 9291 correct results whereas the *maximum J selector* voter gives 8490 correct results. Similarly, for fault amplitude 10, the *modified weighted average* produces 6430 and the *maximum J selector* voter generates 5661 correct outputs. The authors concluded that integration of result and confidence values of voter inputs gives higher availability than processing either *J* alone or the sensor result values alone.

7 Conclusions

Recently smart sensors have been receiving greater attention due to their potential to enhance system safety, availability, and efficiency. However, smart sensors still cannot provide sufficient computation power to perform important functions such as calibration, compensation, digital filtering and programmed self-testing. These issues make questionable the use of smart sensors in ultra-high dependable systems. The use of NMR configuration of smart sensors has been suggested in this article to resolve some of the mentioned problems.

It is concluded that the integration of smart sensors and TMR system in which the information of symptom vector of sensors are used to adjudicate between the sensors result values **in all voting cycles** improves system safety and availability. **Integrating** result and confidence values of redundant sensors gives higher availability than processing either *J* or the sensor result values alone. Where fault/error detection capability of voters is of main interest, a **simple** voter (from the structural viewpoint) which uses the information of symptom vector of sensors in **all voting cycles** gives better results. In this research work, among the implemented voters, Type A voters which use the smartness information of sensors in all voting cycles have a higher level of safety whereas type B and C voters give a higher level of availability than the traditional TMR system with a majority voter.

Acknowledgements

The authors gratefully acknowledge the support of CONACYT Scholarship number 71391 project I35561-A and DISCA-IIMAS UNAM and UNAM-PAPIIT (IN101307 and IN105303) México, and the High Performance Computing Proyect within the "Macroproyecto Tecnologas para la Universidad de la Información y la Computación" of the Universidad Nacional Autonoma de México (UNAM).

References

1. **Bass, J. M. (1995).** "Voting in Real-Time Distributed Computer Control Systems", *PhD thesis, Department of Automatic Control and Systems Engineering, The University of Sheffield*, Sheffield, UK.
2. **Benítez-Pérez, H., Latif-Shabgahi, G., Bass, J. M., Thompson, H. A., Bennett, S. and Fleming, P. J.** (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, P. R. China, July 5-9, pp. 31-36.
3. **Benítez-Pérez, H., Thompson, H. A.** and **Fleming, P. J.** (1998). "Implementation of a Smart Sensor Using a Non-linear Observer and Fuzzy Logic"; *International Conference on Control CONTROL'98, IEE*, Conference Publication Number 455, Volume II, pp. 1474-1479.
4. **Benítez-Pérez, H., Thompson, H., Fleming, P.** (1997). "Implementation of a Smart Sensor Using Analytical Redundancy Technique", *IFAC Safeprocess'97*, Vol. 2, pp. 585-590.

5. **Blanke, M., Kinnaert M., *Lunze* J.** and **Staroswiecki M.;** (2003). "Diagnosis and Fault-Tolerant Control"; *Springer*, Germany.
6. **Broen, R. B.** (1975). "New Voters for Redundant Systems", *Journal of Dynamic Systems, Measurement, and Control*, March 1975, pp. 41-45.
7. **Buskens, R. W.,** and **Bianchini, R. P.** (1993). "Distributed On-Line Diagnosis in the Presence of Arbitrary Faults", *IEEE 23rd Int. Ann. Symp. on Fault-Tolerant Computing Systems*, Toulouse, France, June 1993, pp. 470-479.
8. **Desovski, D.; Liu, Y.; Cukic, B**. (2006), "Linear randomized voting algorithm for fault tolerant sensor fusion and the corresponding reliability model", *IEEE, Proceedings of the Ninth IEEE International Symposium on High-Assurance Systems Engineering (HASE'06*), Page(s):153 - 162
9. **Gersting, J. L., Nist, R. L., Roberts, D. B.,** and **Van Valkenburg, R. L.** (1991). "A Comparison of Voting Algorithms for N-Version Programming", *Digest of papers IEEE 24th Ann. Hawaii Int. Conf. on Systems Sciences*, Vol. 2, pp. 253-62.
10. **Gertler, J. J.** (1998). "Fault Detection and Diagnosis in Engineering Systems", *Marcel Dekker*, Inc., USA.
11. **Henry, M. P.,** and **Clarke, D. W.** (1991). "A Standard Interface for Self-Validating Sensors", *Proc. of the IFAC Safeprocess'91: IFAC/IMAC Symp. on Fault-Detection, Supervision and Safety for Technical Processes*, Baden-Baden.
12. **Henry, M. P.,** and **Clarke, D. W.** (1993). "The Self-Validating Sensor: Rationale, Definitions and Examples", *Control Engineering Practice*, Vol. 1, No. 4, pp. 585-610.
13. **Huijsing, J. H.** (1992). White paper at: (http://www.stw.nl/projecten/D/del2694.html)
14. **Johnson, B. W.** (1989). "Design and Analysis of Fault-Tolerant Digital Systems"; *Addison-Wesley* Publishing Company, USA.
15. **Kanekawa, K., Maejima, H., Kato, H.,** and **Ihara, H.** (1989). "Dependable On-Board Computer Systems with a New Method: Stepwise Negotiated Voting", *Digest of papers IEEE 19th Int. Ann. Symp. on Fault-Tolerant Computing Systems*, Chicago, USA, June 1989, pp. 13-19.
16. **Laprie, J.-C.** (1995). "Architectural Issues in Software Fault-Tolerance", In 'Software Fault Tolerance', *Edited by M. R. Lyu, Published by John Wiley and Sons*, pp. 47-80.
17. **Latif-Shabgahi, G.** (1999). "Performance Analysis of Software Implemented Inexact Voting Algorithms", *PhD thesis, Department of Automatic Control and Systems Engineering, The University of Sheffield*, Sheffield, UK.
18. **Latif-Shabgahi, G.** (2004a). "A Novel Algorithm for Weighted Average Voting used in Fault Tolerant Computing Systems", *Microprocessors and Microsystems*, Vol. 28, pp. 357-361.
19. **Latif-Shabgahi, G.; Bass, J.M.; Bennett, S.,** (2004b), "A taxonomy for software voting algorithms used in safety-critical systems**",** *IEEE Transactions on Reliability*, Volume 53, Issue 3, Sept. 2004 Page(s):319 – 328.
20. **Latif-Shabgahi, G., Bass, J. M., Bennett, S.** (2003). "Smoothing Voter: A Novel Voting Algorithm for Handling Multiple Errors in Fault-Tolerant Control Systems", *Microprocessors and Microsystems Journal*, Vol. 27, No. 7, pp. 303-313.
21. **Lee, P. A.,** and **Anderson, T.** (1990). "Fault-Tolerance Principles and Practice", *Prentice-Hall*.
22. **Lorczak, P. R., Caglayan, A. K.,** and **Eckhardt, D. E.(**1989). "A Theoretical Investigation of Generalised Voters", *Digest of papers, IEEE 19th Int. Symp. nn Fault- Tolerant Computing Systems*, Chicago, USA, June 1989, pp. 444-451.
23. **Meulen, M. (2004).** "On the Use of Smart Sensors, common cause Failure and the Need to Diversity"; *6th International Symposium, Programmable Electronic Systems in Safety Related Applications*.
24. **Nguyen, X.T., Bouzerdoum, A.** and **Moini, A.** (1996). "Velocity measurement using a smart micro-sensor", *CESA'96 IMACS Multiconference: Computational Engineering in Systems Applications*, Lille, France, pp. 937-942.
25. **Yang C.-Y. Janice** (1993). "Self-Validating Sensors"; *Ph.D. Thesis, Queen's College Oxford* University, UK.

Héctor Benítez Pérez *is a full time researcher at IIMAS UNAM (Mexico). He obtained his first degree in Electronic Engineering in 1994 at UNAM. He obtained his phd at Sheffield University, UK in 1999. His research interest are in Real-Time Control and Fault Diagnosis.*

Jorge Luis Ortega Arjona *is a full-time lecturer at the Department of Mathematics, Faculty of Sciences, National Autonomous University of Mexico (UNAM). He obtained his B.Sc. degree in Electronic Engineering, as well as his M.Sc. degree in Computer Science, at UNAM, and pursued his Ph.D. degree at the University College London (UCL) in the United Kingdom. His research interests include parallel processing, object-oriented programming, software patterns and software architecture and design.*

Latif Shabgahi *established his BSc and MSc from the Isfahan University of Technology and Tehran Polytechnique in Iran, respectively, and PhD from the University of Sheffield, UK. He is currently with the Telematics Department of the Open University, Milton Keynes. His research interests include software fault tolerance, dependability aspects of safety-critical systems, using expert systems for dependability assessment, and distributed computer control systems. He has published over 35 papers in national and international conferences and journals.*