Implementation of Predictive Hybrid Redundancy Algorithm using Microcontroller for Safety Critical Systems

M. H. Kim

Division of Advanced Industrial
Science & Technology
DGIST
Daegu, 704-230, KOREA
Email: mhkim@dgist.ac.kr

B. J. Son, K. N. Ha, S. Lee

School of Mechanical Engineering

Pusan National University
Busan, 609-735, KOREA
Email: maligy@pnu.edu,
0vincent@pnu.edu,
slee@pnu.edu

K. C. Lee

Dept of Control and Automation
Engineering
Pukyong National University
Busan, 608-739, KOREA
Email: gclee@pknu.ac.kr

ABSTRACT

The dependence of numerous systems on electronic devices is causing rapidly increasing concern over fault tolerance. For example, a car with electronically controlled steering and no mechanical linkage from the steering wheel to the front tires (x-by-wire) must be fault-tolerant because a critical failure could arise without warning. Fault-tolerant systems have been studied in detail, mainly in the field of aeronautics. This paper presents a predictive hybrid redundancy system that can remove the effect of faults. In addition, this paper introduces an implementation of such a system using an embedded microcontroller unit to show that the predictive hybrid redundancy system outperforms. This experimental results show that the predictive hybrid redundancy system can be a viable choice for the fault-tolerant aspects of safety critical systems.

1. INTRODUCTION

Interest has focused on ways of enhancing safety and convenience for automobile drivers and passengers. For example, in-vehicle network (IVN) systems, in which electronic components such as window motors and switches are connected to an electronic control unit (ECU) through a shared network cable, are widely used in automobiles, trucks, public transportation, and industrial vehicles. Cena (2005). In addition, x-by-wire systems are being developed to expand the application of IVN systems to real-time components, such as brakes, throttle, and steering systems.

In general, x-by-wire systems consist of bus systems, ECUs, electrical actuators, and sensors, to make vehicles lighter, cheaper, safer, and more fuel-efficient. In addition, they are self-diagnosing and configurable; they adapt easily to different vehicle platforms and produce no environmentally harmful fluids. However, replacing rigid mechanical components with dynamically configurable electronic elements triggers an almost organic system-wide level of integration. Ensuring the safety and reliability of x-by-wire systems requires integration methods that are reliable and fault-tolerant. Isermann (2002).

The design of fault-tolerant functions includes

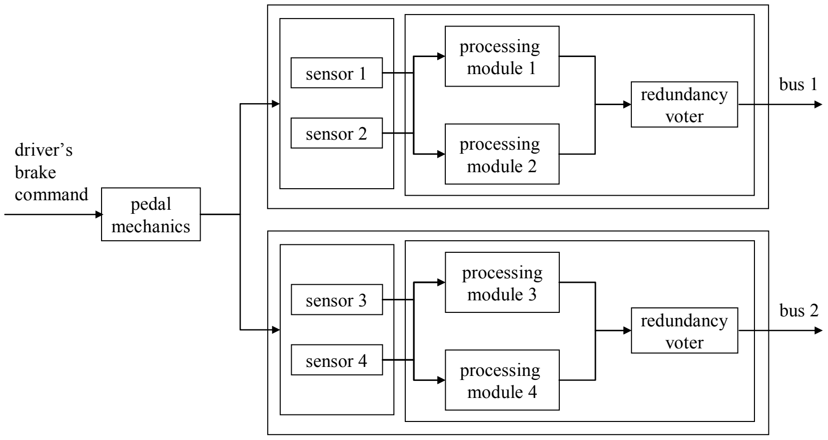


Figure 1. DUO-DUPLEX ARCHTECTURE FOR THE BRAKE PEDAL MODULE FOR A BRKAE-BY-WIRE SYSTEM.

redundant systems that duplicate several modules with the same function, such as motors, microcontrollers, and sensors. If one module is faulty, the fault is isolated, and safe operation is guaranteed by replacing the faulty module with a serviceable module within a predefined interval. One of the most common applications is the electronic brake pedal module with a duo-duplex architecture, which constitutes a type of redundant brake-by-wire system, as shown in Figure 1. In the figure, the duo-duplex pedal module consists of four sensors that measure the output value of the brake pedal, four processing modules that deal with signal conditioning and processing, and two redundancy voters that determine an output value using various voting algorithms. Isermann (2002).

In particular, if a critical safety system such as an x-by-wire system lacks redundancy, a traffic accident may occur if one of the sensing or signal-processing modules is out of order. Fortunately, due to the extensive functionality and low cost of microcontrollers, hardware redundancy systems are widely used to include extra hardware with identical functions. In addition, various fault-detection algorithms have been developed to guarantee the reliability of the hardware redundancy systems.

As an alternative to the hybrid redundancy system, this paper presents the architecture of the predictive hybrid redundancy system with a fault detection algorithm based on the double exponential smoothing method. Experimental performance of the predictive hybrid redundancy system using an embedded microcontroller unit with a brake pedal signal is evaluated.

2. OVERVIEW OF REDUNDANCY SYSTEM

In general, the hardware redundancy system, which is to add an extra hardware with the same functions implemented in the original hardware, can be classified into static redundancy, dynamic redundancy, and hybrid redundancy system according to its architecture and function. The static redundancy system requires a voter that determines the final output of the system. The voter can use majority or average rule as its fault masking algorithm to isolate any faulty input. However, the static redundancy system tends to cost more because it requires at least three parallel modules for majority voter, and it is difficult to detect faults when two or more modules are faulty.

The dynamic redundancy system achieves fault tolerance by having fault detection and reconfiguration functions instead of a voter. In general, the dynamic redundancy system can be classified into hot and cold standby dynamic redundancy system according to whether all modules are always operating or not. In the hot standby dynamic redundancy system, two modules are constantly sending their outputs, and the output switch is connected either of two modules. On the other hand, the cold standby dynamic redundancy system uses only one module at a time, and two switches are controlled by the reconfiguration module to block the signal from the faulty module. Here, the cold standby has a longer module life and better energy efficiency than hot standby because a single module is working at a time. But, the cold standby needs complex fault detection algorithm because it uses only one input value.

Figure 2 shows a type of hybrid redundancy system called n-modular redundancy with spares that combines both static and dynamic approaches. It has a voter along with switch and disagreement detector. By combining two approaches, the system can mask a fault as the static approach does while it can also detect a fault and reconfigure the system just like the dynamic approach. Although this hybrid redundancy system has a certainly better reliability than the previous two, its application has been limited to safety-critical systems such as spacecraft and aircraft due to high cost and complexity. However, by virtue of inexpensive microcontrollers with high computing power, the hybrid redundancy becomes a promising approach to wider range of safety-critical application.

3. STRUCTURE OF THE PREDICTIVE HYBRID REDUNDANCY SYSTEM

We propose the predictive hybrid redundancy system structure to provide redundancy for products such as intelligent vehicles. The rationale for having this type of structure is that we cannot afford too many redundant sensors on a system such as a passenger car. Therefore, the cost and complexity of static or n-modular redundancy with spares are hard to justify. Alternatively, we could consider the hybrid of hot standby dynamic redundancy and static redundancy along with some capability to forecast the input value. The predictive hybrid redundancy system makes use of a powerful microcontroller to forecast the change from the last value of the system output using the double exponential smoothing method. The predicted change is

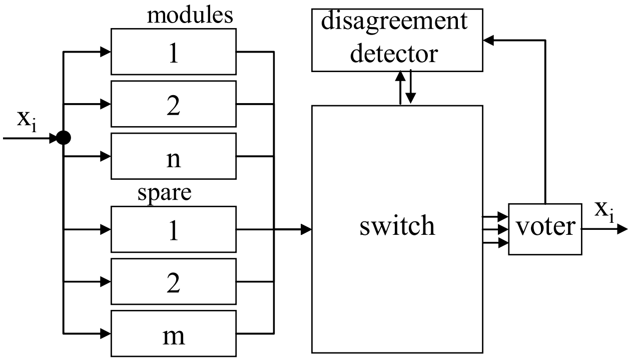


Figure 2. STRUCTURE OF HYBRID REDUNDANCY SYSTEM.

used to judge if a fault exists in the current value of the sensor signal by checking whether the new inputs lie within the predicted interval. This forecast is based on the assumption that the input signal usually changes gradually without large fluctuations.

The predictive hybrid redundancy system consists of four modules: a predictor, a fault detector, an exception handler, and a voter. First, the predictor forecasts a threshold, which is essentially the expected change in the output signal. Second, the fault detector determines whether a fault exists in the two inputs using the forecast threshold. Third, the exception handler decides on an output value when the fault detector determines that both input values are unreliable. Finally, the voter calculates an output value by averaging the input values it receives.

The predictor must first forecast a threshold value for judging whether a fault exists in the values given to the fault detector. To forecast a threshold in the predictive hybrid redundancy system, the double exponential smoothing method, which is a representative method of time series forecasting methodology, was chosen. In general, exponential smoothing is a forecasting method that assigns exponentially decreasing weights as observations get older. In other words, recent observations are given relatively more weight than older observations in the forecasting process. The double exponential smoothing method removes random variation and shows trends and cyclic components in forecasting. Since the encoder output signal or the actuator input signal may follow a trend such as a sine wave or step response in real industrial applications, the double exponential smoothing method may be appropriate for forecasting the patterns of these signals. For forecasting the k-th threshold FT(k) of a predictive hybrid redundancy system, the double exponential smoothing method can be expressed as follows:

$$FT^{[1]}(k) = \alpha RT(k) + (1-\alpha)FT^{[1]}(k-1)$$
$$FT^{[2]}(k) = \alpha FT^{[1]}(k) + (1-\alpha)FT^{[2]}(k-1)$$
$$FT(k) = (2+\frac{\alpha}{1-\alpha})FT^{[1]}(k) - (1+\frac{\alpha}{1-\alpha})FT^{[2]}(k)$$
$$where\ RT(k) = r(k-1) - r(k-2)$$

where $FT^{[1]}(k)$ and $FT^{[2]}(k)$ are the first and second step forecast thresholds determined using the exponential smoothing method in the k-th cycle, respectively, and RT(k) is the real threshold in the k-th cycle. In addition, r(k) is an output value in the k-th cycle, and α, which is generally selected to be in the range 0.05 and 0.3, is the double exponential smoothing parameter. FT(k) is the forecast change in the input signal, which is used as a threshold to judge whether a fault exists in the values input to the fault detector, in the k-th cycle using $FT^{[1]}(k)$ and $FT^{[2]}(k)$. Here, r(–1), r(–2), $FT^{[1]}(0)$, and $FT^{[2]}(0)$ are initialized to zero; r(0) is assumed to be the average of the first input values of the two modules and is assumed to be error-free.

After forecasting the threshold FT(k) in the predictor, the fault detector judges whether a fault exists in the two input values using FT(k), as shown in Figure 3. In the figure, if the difference between the k-th input value $a_i(k)$ and the (k–1)-th output value r(k–1) is within the range (1±β)FT(k), the fault detector determines that ai(k) is error-free. Conversely, if $a_i(k)$ exceeds a permitted limit of the forecast input value, the corresponding input is considered to be erroneous. If at least one input is determined to be error-free, the reconfigurator is called upon to generate the k-th output value r(k) of the predictive hybrid redundancy system. Here, it is necessary to determine the appropriate value of β based on the features of the system. However, if both inputs are determined to be erroneous, the fault detector calculates $d_{ij}(k)$, which is defined as the difference between the k-th values $a_j(k)$ and $a_j(k)$ of two input modules. Here, if $d_{ij}(k)$ is within the range ±FT(k), the two input values are regarded as error-free and the reconfigurator is called upon to calculate r(k). That is, if both inputs exceed a set limit of the forecast input value and the difference between the two inputs is smaller than an allowable error, the two inputs are considered to be varying rapidly due to some unexpected external disturbance. Conversely, if $d_{ij}(k)$ exceeds the range of FT(k), the two inputs are considered erroneous and the exception handler is called upon to determine the output value.

When the two inputs are determined to be erroneous, the exception handler calculates a plausible output to prevent malfunction of the hybrid redundancy system. To determine the output value r(k), the exception handler calculates the difference between the (k–1)th output and the (k–2)th output. If the difference is positive, the exception handler judges that the input value is increasing and determines the output to be the sum of r(k–1) and FT(k). If the difference is negative, the output is determined to be the difference of r(k–1) and FT(k). Because the output can be determined as the second best value even if both inputs are erroneous, it is possible to prevent abnormal operation of the system that may occur due to the absence of output using this method.

4. EXPERIMENTAL PERFORMANCE EVALUATION OF THE PREDICTIVE HYBRID REDUNDANCY SYSTEM

This section evaluates the performance of the predictive hybrid redundancy system along with implementation details for the experimental setup (test bed) using an embedded microcontroller unit as shown in Figure 4. This experimental setup is intended to represent a redundant brake pedal module with two potentiometers for measuring angular displacement of brake pedal in the predictive hybrid redundancy system. In a conventional hydraulic brake pedal system, the pedal is connected to a hydraulic brake booster. In the experimental setup, two electrical potentiometers are attached to the brake pedal axis to measure the angular displacement of the brake pedal. The fault injector is connected to the signal lines of potentiometers to emulate potentiometer faults.

Figure 4(b) shows the implementation details for the experimental setup. A Copal Electronics J45S 10k potentiometer was used for measuring the displacement of the brake pedal, and a Freescale MC9S12DP256 microcontroller was used for the smoothing predictive redundancy module. A notebook computer running Vector CANoe software was connected to the module via a controller area network (CAN) to monitor the processing result of the smoothing predictive redundancy module. Algorithms for the predictor, fault detector, exception handler, and voter were implemented using Mathworks MATLAB Simulink and Stateflow software, converted into a C program using Mathworks Real-Time Workshop software, and stored in the MC9S12DP256 microcontroller using Freescale CodeWarrior software. Finally, to emulate consistent movements of the brake pedal for performance evaluation, we added a Tamagawa 1981N134E9 DC motor with a limit sensor to the brake pedal axis. To compare our system with other redundancy systems, we attached another potentiometer to the brake axis and implemented average and median voters, which are commonly used in triple modular redundancy (TMR) system, using an MC9S12DP256 microcontroller and MATLAB Real-Time Workshop.

Figure 5 shows the brake pedal signal of predictive hybrid redundancy systems with the 4000 injected faults, using values α = 0.3 and β = 0.0625 for the predictive hybrid redundancy system determined by experimental trial and error. In the figure, it appears almost identical to the original signal. This indicates that the predictive hybrid redundancy can be an appropriate algorithm for safety critical systems because the exception handler determines a feasible output for safe operation even if both input values are temporarily faulty.

These results indicate that the performance of the predictive hybrid redundancy is superior to those of the average and median voters. In addition, since the average

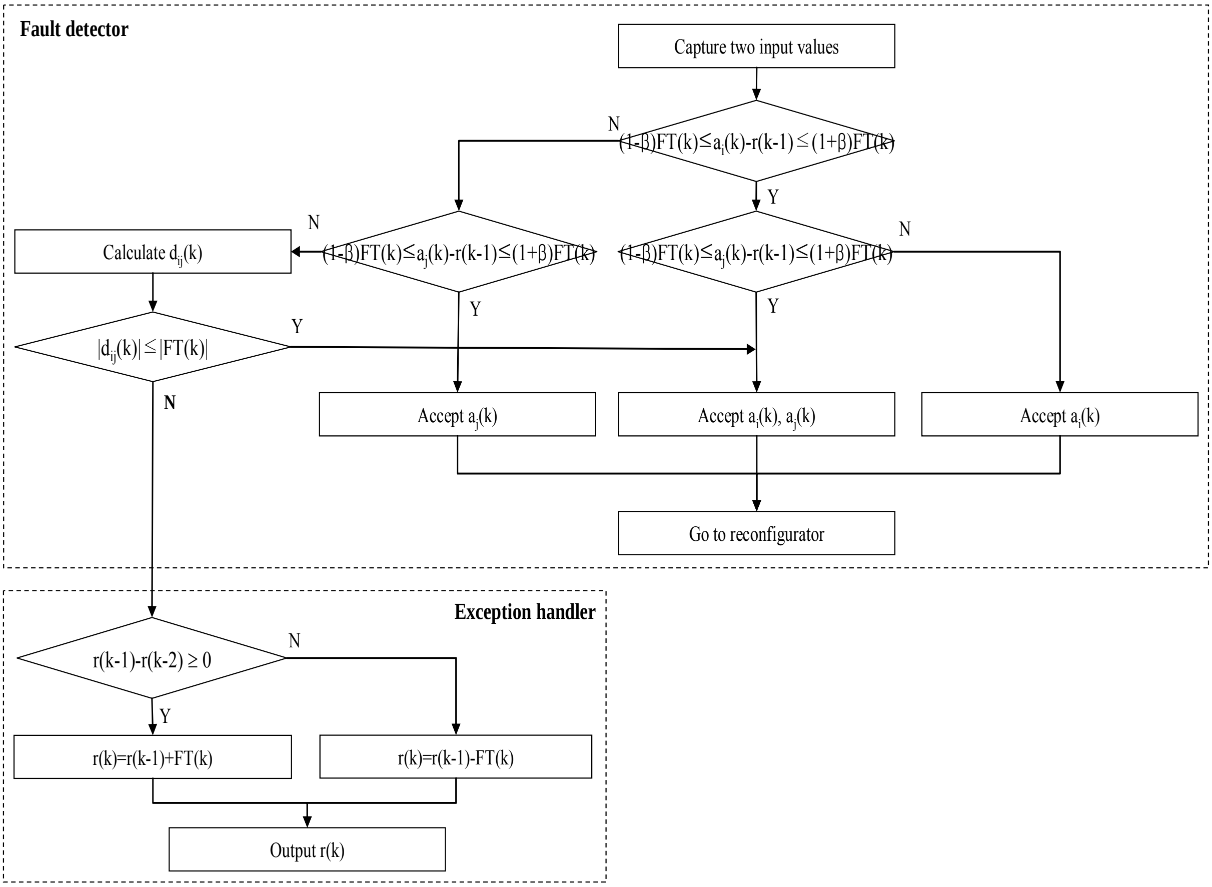

Fault detector
Capture two input values
N
$(1-\beta)FT(k) \le a_i(k)-r(k-1) \le (1+\beta)FT(k)$
Y
N
Calculate $d_{ij}(k)$
$(1-\beta)FT(k) \le a_j(k)-r(k-1) \le (1+\beta)FT(k)$
$(1-\beta)FT(k) \le a_j(k)-r(k-1) \le (1+\beta)FT(k)$
N
Y
Y
Y
$|d_{ij}(k)| \le |FT(k)|$
N
Accept $a_j(k)$
Accept $a_i(k)$, $a_j(k)$
Accept $a_i(k)$
Go to reconfigurator
Exception handler
$r(k-1)-r(k-2) \ge 0$
N
Y
$r(k)=r(k-1)+FT(k)$
$r(k)=r(k-1)-FT(k)$
Output r(k)

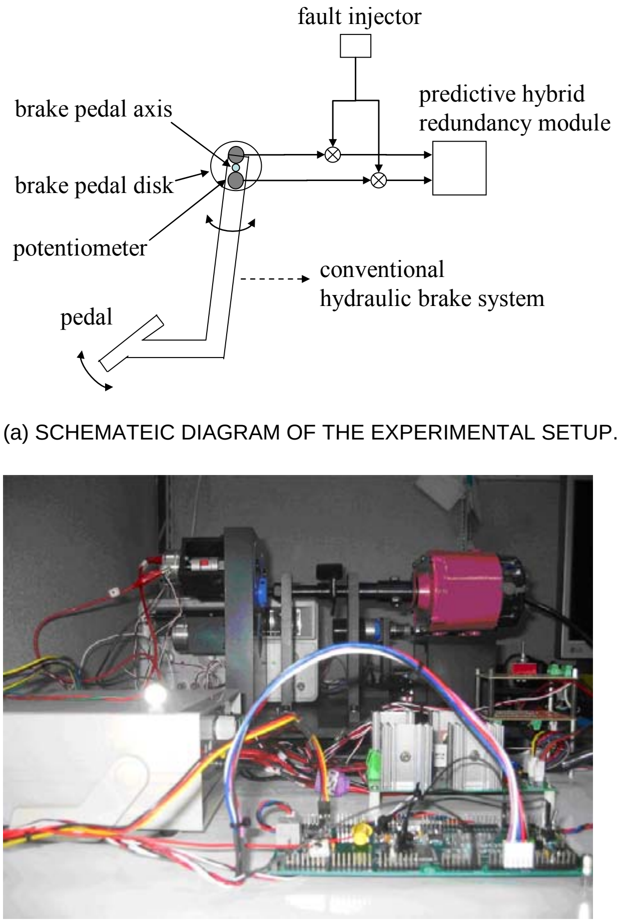
Figure 3. ALGORITHM OF THE FAULT DETECTOR AND EXCEPTION HANDLER OF PREDICTIVE HYBRID REDUNDANCY SYSTEM.

and median voters use three sensors while the predictive hybrid redundancy uses only tow, implementing a redundant system with fewer redundant sensors becomes possible. This may mean a less-expensive system because the sensor will likely cost more than the microprocessor required to execute the algorithm.

4. SUMMARY AND CONCLUSIONS

This paper presents the predictive hybrid redundancy system along with a fault detection algorithm using the double exponential method for safety critical systems such as the x-by-wire systems. To verify the feasibility of predictive hybrid redundancy, we developed a predictive hybrid redundancy system using an embedded microcontroller unit with a brake pedal signal, and evaluated the performance. The conclusions derived from this research are as follows.

First, the experimental results showed that predictive hybrid redundancy can eliminate faults The fault-masked signal was quite similar to the original signal without faults. The simulation demonstrated that predictive hybrid redundancy can be very effective for safety critical systems.



(a) SCHEMATEIC DIAGRAM OF THE EXPERIMENTAL SETUP.

(b) IMPLEMENTION DETAILS FOR THE EXPERIMENTAL SETUP.

Figure 4. EXPERIMENTAL SETUP OF THE PREDICTIVE HYBRID REDUNDANCY SYSTEM.

Second, since the general voting algorithms such s average and median require three sensors while the predictive hybrid redundancy needs only two, implementing a redundancy system more cost-effectively may be possible.

The natural extension of this research is to compare the performance of predictive hybrid redundancy with those of other dynamic and hybrid methods, along with an experimental demonstration of its efficiency. Furthermore, other methods to handle prolonged faults at both sensors should be developed because the proposed method may be inadequate in such situations.

REFERENCES

Cena, G.., Valenzano A., and Vitturi S., “Advances, in automotive digital communications,” Computer Standards & Interfaces, vol. 27, no. 6, pp. 665-678, 2005.

Isermann, R., Schwarz, R., and Stolzl, S., “Fault-tolerant drive-by-wire systems,” IEEE Control Systems Magazine, vol. 22, no. 5, pp. 64-81, 2002.

Kim, M. H., Lee S., and Lee, K. C., “Predictive Hybrid Redundancy using Exponential Smoothing Method for Safety Critical Systems,” International Journal of Control, Automation and Systems, vol. 6, no. 1, pp. 126-134, 2008.

Shabgahi, G. L., Bass, J. M., and Bennett, S., “A taxonomy for software voting algorithm used in safety-critical systems,” IEEE Transactions on Reliability, vol. 53, no. 3, pp. 319-328, 2004.

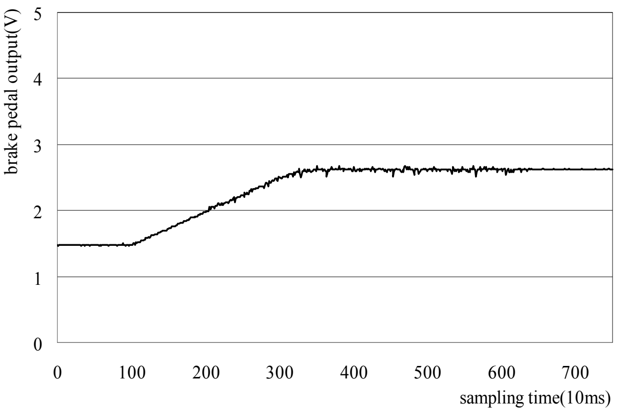


Figure 5. BRAKE PEDAL SIGNAL USING PREDICTIVE HYBRID REDUNDANCY SYSTEM WITH 4,000 INJECTED FAULTS.