

Kalman Predictive Redundancy System for Fault Tolerance of Safety-Critical Systems

Man Ho Kim, Suk Lee, *Member, IEEE*, and Kyung Chang Lee, *Member, IEEE*

Abstract—The dependence of intelligent vehicles on electronic devices is rapidly increasing the concern over fault tolerance due to safety issues. For example, an x-by-wire system, such as electromechanical brake system in which rigid mechanical components are replaced with dynamically configurable electronic elements, should be fault-tolerant because a critical failure could arise without warning. Therefore, in order to guarantee the reliability of safety-critical systems, fault-tolerant functions have been studied in detail. This paper presents a Kalman predictive redundancy system with a fault-detection algorithm using the Kalman filter that can remove the effect of faults. This paper also describes the detailed implementation of such a system using an embedded microcontroller to demonstrate that the Kalman predictive redundancy system outperforms well-known average and median voters. The experimental results show that the Kalman predictive redundancy system can ensure the fault-tolerance of safety-critical systems such as x-by-wire systems.

Index Terms—Fault-tolerant system, intelligent vehicle, Kalman filter, Kalman predictive redundancy system, safety-critical systems, x-by-wire system.

I. INTRODUCTION

INTEREST has been recently focused on intelligent vehicles that offer a significant enhancement of safety and convenience to drivers and passengers [1], [2]. As a component of an intelligent transportation system, intelligent vehicles use various intelligent sensing and control algorithms to assist safe driving [3], [4]. In particular, x-by-wire systems that replace rigid mechanical components with dynamically configurable electronic elements are being developed to expand intelligent functions, such as adaptive cruise control or lane departure warning system [5], [6]. However, x-by-wire systems require a higher level of fault tolerance than traditional systems based on mechanical links, because mechanical systems can provide drivers with some warning feedback of their status, while electronic systems may abruptly fail without any warning. In addition, because the malfunction of safety-related modules controlling critical functions, such as brakes, throttle, and

steering in x-by-wire systems, can cause injury or death of drivers and passengers, the fault tolerance is the most critical issue for automotive vendors and parts suppliers.

Generally, the design of fault-tolerant functions includes redundancy systems using duplicate modules (e.g., motors, micro-controllers, or sensors) with identical functions in the x-by-wire systems [5], [7], [8]. If one module is faulty, the fault will be isolated, and safe operation can be guaranteed by replacing the faulty module with a serviceable module within a predefined interval.

Redundancy systems can be classified into the following groups according to the architecture and function: static redundancy, dynamic redundancy, and hybrid redundancy, as shown in Fig. 1, [9]–[13]. Fig. 1(a) shows the static redundancy system with multiple parallel modules. The static redundancy system requires a voter to determine the final output of the system. The voter can use a median, average, or majority rule as its fault-masking algorithm to isolate any faulty inputs. However, the static redundancy tends to cost more because it requires at least three parallel modules for median or average voter, and it is difficult to detect faults when two or more modules are faulty. A dynamic redundancy system achieves fault tolerance with fault detection and reconfiguration functions instead of a voter. A hot standby dynamic redundancy system [Fig. 1(b)] uses two modules simultaneously for processing outputs. The fault detector determines which module is correct and the reconfigurator selects either of the two modules by using output switch. On the other hand, a cold standby dynamic redundancy system [Fig. 1(c)] uses only one module at a time. The fault detector decides which model is fault free, and the reconfigurator operates the switches so that the input is directed to, and the output is from the selected module. The dynamic redundancy needs complex fault detection algorithm for the fault detector and the reconfigurator. By combining these two approaches, a hybrid redundancy system can mask a fault as the static approach does using a voter, and it can also detect a fault and reconfigure the system just like the dynamic approach using a switch with fault detection algorithm. Fig. 1(d) shows an example of the hybrid redundancy system, which is called self-purging redundancy [13]. The switch removes or purges its associated module from the system when the module fails. The voter determines the output by using weighted sum.

In order to implement more effective and cost-efficient hardware type of the hybrid redundancy system, this paper presents the Kalman predictive redundancy system. This paper proposes a fault detection algorithm for a disagreement detector to detect and isolate faulty elements from a signal by forecasting the change from the last value of the input signal using Kalman

Manuscript received February 27, 2009. First published May 27, 2009; current version published February 05, 2010. This work was carried out under the General R/D Program sponsored by the Ministry of Education, Science and Technology (MEST) of the Republic of Korea. Paper no. TII-09-02-0033.

M. H. Kim and S. Lee are with the School of Mechanical Engineering, Pusan National University, Busan 609-735, Korea (e-mail: mhkim@dgist.ac.kr; slee@pnu.edu).

K. C. Lee is with the Department of Control and Automation Engineering, Pukyong National University, Busan 608-739, Korea (e-mail: gclee@pknu.ac.kr).

Color versions of one or more of the figures in this paper are available online at <http://ieeexplore.ieee.org>.

Digital Object Identifier 10.1109/TII.2009.2020566

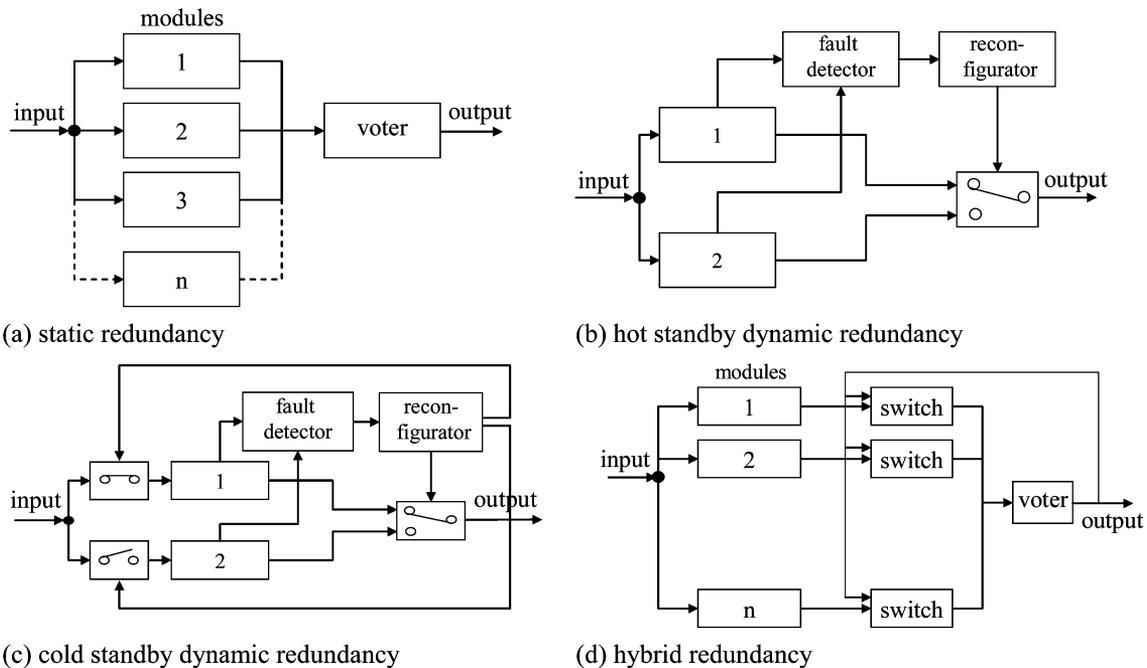


Fig. 1. Various types of hardware redundancy.

filter. Kalman filter is an efficient recursive filter that estimates the state of a dynamic system from a series of noisy measurements, in a way that minimizes the mean of the squared error [14]–[19]. In general, since the inputs such as encoder or actuator signal usually changes gradually without large fluctuations, the Kalman filter may be more appropriate for forecasting the patterns of these signals than any other prediction method.

This paper is organized into four sections including this introduction. Section II presents the structure of the Kalman predictive redundancy system with prediction, fault detection, and an exception-handling algorithm. Section III describes the detailed implementation of such a system using an embedded microcontroller to demonstrate that the Kalman predictive redundancy system outperforms well-known average and median voters. The experimental results show that the Kalman predictive redundancy system can ensure the fault-tolerance of safety-critical systems such as x-by-wire systems. Finally, the conclusion is presented in Section IV.

II. STRUCTURE OF KALMAN PREDICTIVE REDUNDANCY SYSTEM

In order to enhance redundancy for safety-critical systems such as the x-by-wire systems in an intelligent vehicle, this section proposes structure of the Kalman predictive redundancy system shown in Fig. 2. The Kalman predictive redundancy system uses a powerful microcontroller to forecast the change from the last value of the input signal using a Kalman filter. The predicted change is 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 Kalman predictive redundancy system consists of four modules: a Kalman predictor, a fault detector, an exception han-

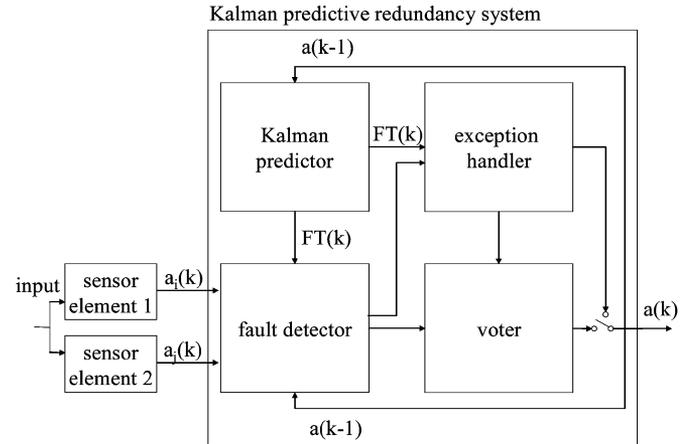


Fig. 2. Schematic diagram of the Kalman predictive redundancy system.

dlar, and a voter. The Kalman predictor forecasts a threshold ($FT(k)$), which is essentially the expected change in the output signal; this threshold is calculated using Kalman filter as explained below. The fault detector determines whether a fault exists in two input values ($a_i(k)$ and $a_j(k)$) using the forecast threshold in the Kalman predictor. If an input value lies within an interval centered on the system output in the previous step ($a(k-1)$), the fault detector decides that the input is error-free. The interval's width is directly proportional to the forecast threshold. The exception handler determines an output value ($a(k)$) when the fault detector determines that both input values are unreliable. When an unexpected external disturbance affects the system, the fault detector determines that both input values are erroneous and that an output cannot be determined from them. When that occurs, the exception handler calculates an output value by incrementing or decrementing the last output value to prevent abnormal operation due to the absence of an

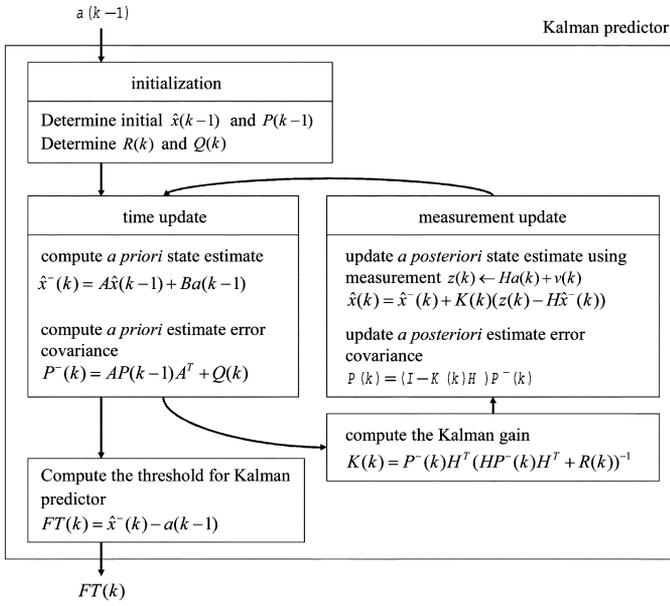


Fig. 3. Summarized procedure for operation of the Kalman predictor.

output value. Finally, the voter calculates an output value by averaging the input values it receives.

The Kalman predictor forecasts a threshold value for judging whether a fault exists in the values provided to the fault detector. To forecast the k th threshold $FT(k)$ of the Kalman predictive redundancy system, the Kalman predictor estimates the state of the discrete-time controlled process that is governed by the linear stochastic difference equation [14]

$$\begin{aligned} x(k) &= Ax(k-1) + Ba(k-1) + w(k-1) \\ z(k) &= Hx(k) + v(k) \end{aligned} \quad (1)$$

where $x(k) \in \mathfrak{R}^n$ and $z(k) \in \mathfrak{R}^m$ are the state and measurement, respectively, and $a(k-1)$ is the input. Parameters $w(k)$ and $v(k)$ are the process and measurement noise, respectively. They are assumed to be independent with normal probability distributions as follows:

$$\begin{aligned} p(w(k)) &\sim N(0, Q(k)) \\ p(v(k)) &\sim N(0, R(k)). \end{aligned} \quad (2)$$

In practice, the process noise covariance $Q(k)$ and measurement noise covariance $R(k)$ may change with each time step or measurement; however, we have assumed these to be constant.

The Kalman predictor defines the *a posteriori* state estimate and the *a posteriori* estimate error covariance $P(k)$ as follows:

$$\begin{aligned} E[a(k)] &= \hat{x}(k) \\ E[(x(k) - \hat{x}(k))(x(k) - \hat{x}(k))^T] &= P(k). \end{aligned} \quad (3)$$

Fig. 3 shows a summarized procedure for operation of the Kalman predictor. The first step is to determine the process

noise covariance $Q(k)$ and the measurement noise covariance $R(k)$. In the actual Kalman filter implementation, regardless of whether there is a rational basis for choosing these parameters, superior filter performance can often be obtained by tuning them. The tuning is usually performed offline, often with the help of another distinct Kalman filter in a process generally referred to as system identification [14]. We also need to determine the initial *a posteriori* state estimate $\hat{x}(k-1)$ and the *a posteriori* estimate error covariance $P(k-1)$. The second step is to predict the *a priori* state estimate $\hat{x}^-(k)$ and the *a priori* estimate error covariance $P^-(k)$ in the time update. The third step is to compute the Kalman gain $K(k)$ to update the *a posteriori* state estimate $P(k)$ and the *a posteriori* estimate error covariance $z(k)$. After gaining the k th step measurement $\hat{x}(k)$ from the discrete-time controlled system, the fourth step is to update $P(k)$ and $\hat{x}^-(k+1)$ in the measurement update. These will be used to compute $P^-(k+1)$ and $\hat{x}^-(k)$ in the next step. Finally, the Kalman predictor calculates $FT(k)$ using the difference between the k th *a priori* estimate and the $(k-1)$ th input value $a(k-1)$.

After forecasting the threshold $FT(k)$ in the Kalman predictor, the fault detector determines whether a fault exists in the two input values using $FT(k)$, as shown in Fig. 4. If the difference of the k th incoming sensor value $a_i(k)$ and the $(k-1)$ th determined sensor value $a_i(k-1)$ is within the range $(1 \pm \beta)FT(k)$, the fault detector decides that $a_i(k)$ is error-free. Conversely, if $a_i(k) - a_i(k-1)$ exceeds a permitted limit of the range, the corresponding sensor input $a_i(k)$ is considered to be erroneous. If at least one input is judged to be error-free, the voter determines the k th sensor value $a(k)$ value by averaging the previous input values. Thus, it is necessary to determine the appropriate value of β by trial and error based on the features of the system. However, if both sensor inputs are determined to be erroneous, the fault detector calculates $d_{ij}(k)$, which is defined as the difference of the k th incoming sensor values $a_i(k)$ and $a_j(k)$. If $d_{ij}(k)$ is within the forecast threshold range $\pm FT(k)$, the two input values are considered to be error-free and the voter calculates $a(k)$ by averaging the two values. This is because that two inputs are considered to be varying rapidly due to an unexpected external disturbance when both inputs exceed a permitted limit of the range $(1 \pm \beta)FT(k)$ and the difference between the two inputs is smaller than an allowable error threshold range $\pm FT(k)$. Conversely, if $d_{ij}(k)$ exceeds the forecast threshold range $\pm FT(k)$, the two input values are considered to be erroneous.

When the two input values are determined to be erroneous, the exception handler calculates a feasible output value to prevent malfunction of the Kalman predictive redundancy system. To determine the output value $a(k)$, the exception handler calculates the difference between the $(k-1)$ th output value and the $(k-2)$ th output value. If $a(k-1) - a(k-2)$ is positive, the exception handler decides that the input value is increasing and determines the output value to be $a(k-1) + FT(k)$. If $a(k-1) - a(k-2)$ is negative, the output value is determined to be $a(k-1) - FT(k)$. Since the output value can be determined as the second-best value, even if both input values are erroneous, it is possible to prevent the abnormal operation of the system that may occur in the absence of an output value.

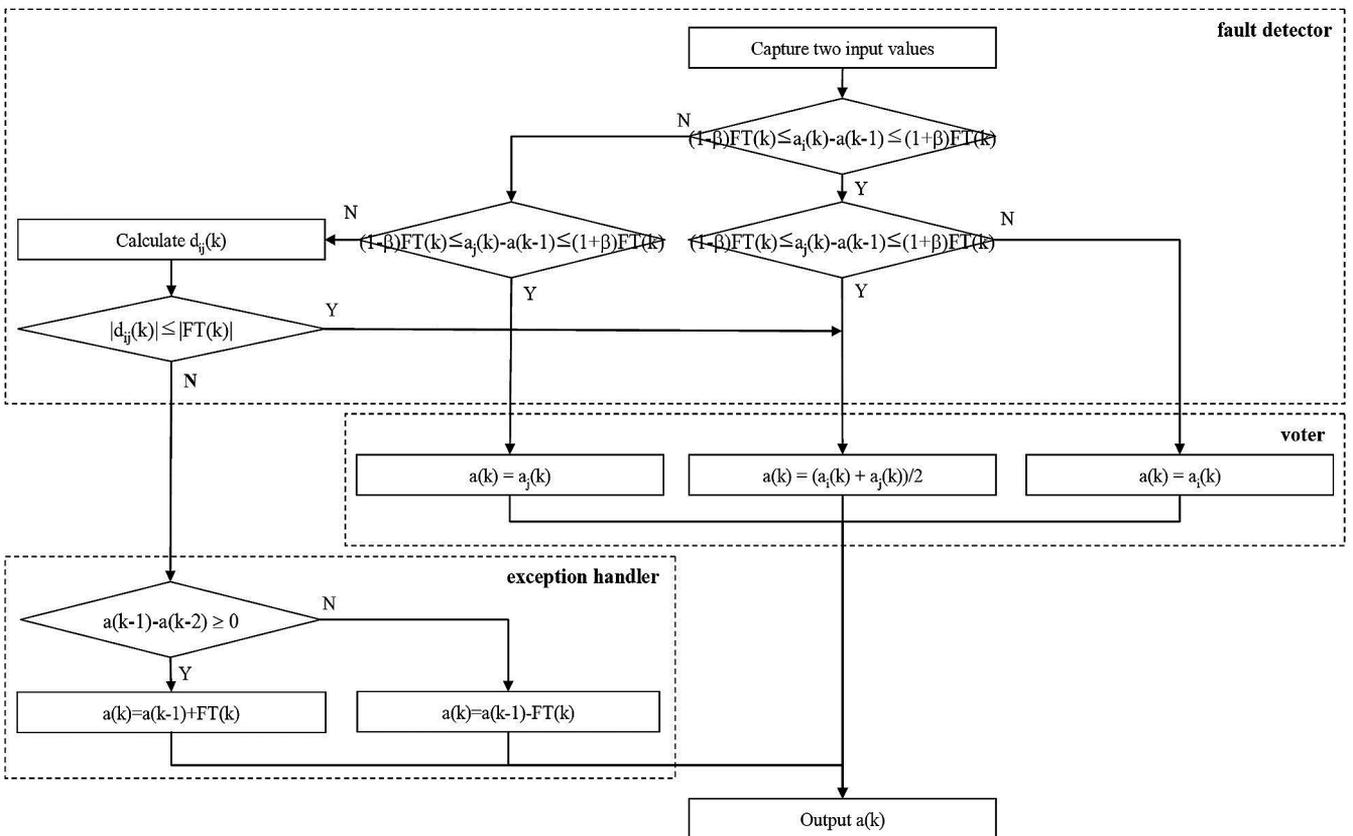


Fig. 4. Algorithm of the fault detector, exception handler, and voter.

III. EXPERIMENTAL PERFORMANCE EVALUATION OF THE KALMAN PREDICTIVE REDUNDANCY SYSTEM

This section describes the performance of the Kalman predictive redundancy system along with implementation details for the experimental setup (test bed) using an embedded microcontroller unit, as shown in Fig. 5(a). This schematic diagram for the experimental setup is intended to represent a redundant brake pedal module with two potentiometers for measuring the angular displacement of the brake pedal in a Kalman predictive redundancy system. In a conventional hydraulic brake pedal system, the pedal is connected to a hydraulic brake booster. In our schematic diagram for the experimental setup, electrical potentiometers were attached to the brake pedal axis to measure the angular displacement of the pedal. In addition, a fault injector was connected to the potentiometer signal lines to emulate potentiometer faults.

Fig. 5(b) shows the implementation details for the experimental setup. A Copal Electronics J45S 10-kΩ potentiometer was used to measure the displacement of the brake pedal, and a Freescale MC9S12DP256 microcontroller was used for the Kalman predictive redundancy module. A notebook computer running Vector's CANoe software was connected to the module via a controller area network (CAN) to monitor the output of the Kalman predictive redundancy module. The Kalman predictive redundancy system was implemented using Mathworks MATLAB Simulink software, converted into C language code using Mathworks Real-Time Workshop, and downloaded to the

MC9S12DP256 microcontroller using Metroworks CodeWarrior. In the implementation details for the experimental setup, it is necessary to press on the brake pedal in the same pattern for comparing experimental results. Hence, to emulate consistent movement of the brake pedal, we added a direct current (DC) motor with a limit sensor to the brake pedal axis.

The Kalman predictor was implemented using the Kalman filter function block of Simulink. The fault detector, exception handler, and voter were implemented using general Simulink function blocks. We chose parameters for the Kalman predictor with the MATLAB Simulink model using simulational trial and error as follows:

$$\begin{aligned}
 A &= [1], & B &= [1], & H &= [1] \\
 Q(k) &= 1 \times I, & R(k) &= 0.5 \times I
 \end{aligned} \quad (4)$$

where the initial values of $a(0)$, $P(0)$, $\hat{x}(0)$, and $FT(0)$ were zero; $a(1)$ was assumed to be error-free and set to the average of the first input values of the two modules.

For comparison with general redundancy systems, we attached another potentiometer to the brake axis and implemented an average and a median voter using a MC9S12DP256 microcontroller and Real-Time Workshop; such a system is commonly used in triple modular redundancy (TMR) systems. The average voter and median voters are the most representative static redundancy systems, and are used for various application areas such as vehicle or aircraft [5], [7], [8]. Besides, the

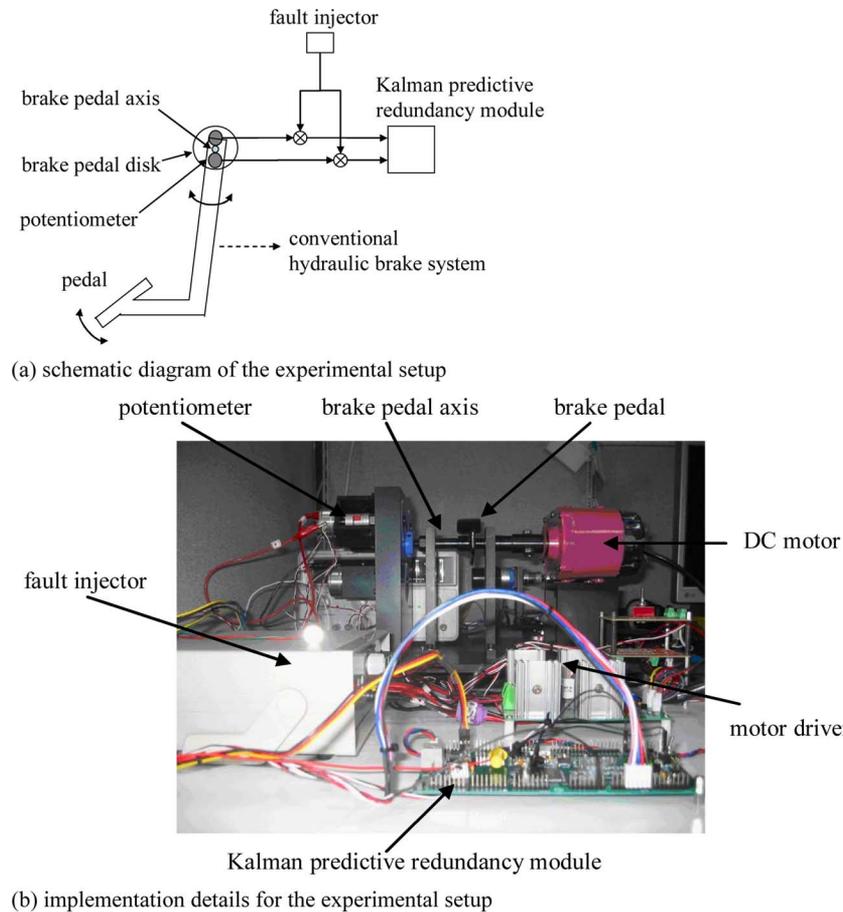


Fig. 5. Experimental setup of the Kalman predictive redundancy system.

two voters are often used for comparison among the other voters [11].

In order to determine an appropriate value for the fault detection parameter β , we defined the integral of the absolute magnitude of the error (IAE) as follows [21]:

$$IAE = \sum_{k=1}^n \Delta T \cdot |e(k)| \quad (5)$$

where $e(k)$ is the difference between the original brake pedal value and the fault-masked brake pedal processed, and ΔT is the sampling time of the brake pedal signal.

Fig. 6 shows the IAE performance index of the Kalman predictive redundancy for varying fault values in the Mathworks MATLAB Simulink model. Here, the fault value is defined the ratio of faulty signal and normal signal. From trial and error simulations, we have chosen the value of 0.0625 for β .

Fig. 7(a) shows an example of a brake pedal signal captured from the potentiometer when the DC motor was activated. The figure shows that the potentiometer voltage was 1.4 V when the brake pedal was released, and it increased to 2.6 V when the brake pedal was slowly pressed. Fig. 7(b) shows an example of a brake pedal signal measured from a faulty potentiometer. To emulate a faulty potentiometer, we used a fault injector, as shown in Fig. 5(a), to add an impulse-type fault to the normal potentiometer signal. This fault signal can appear and disappear

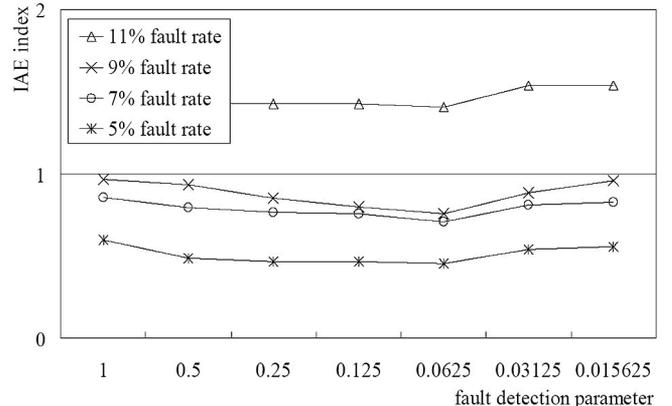
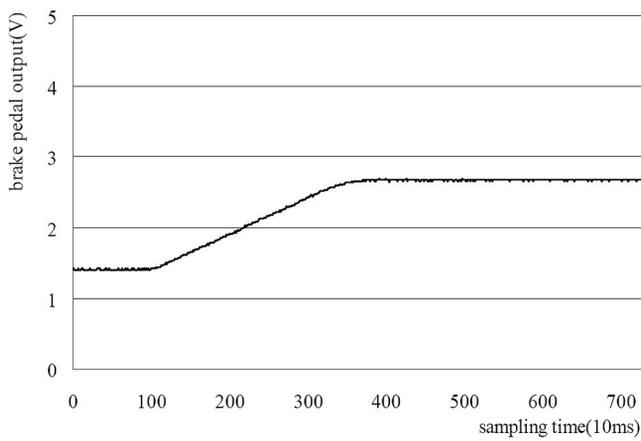
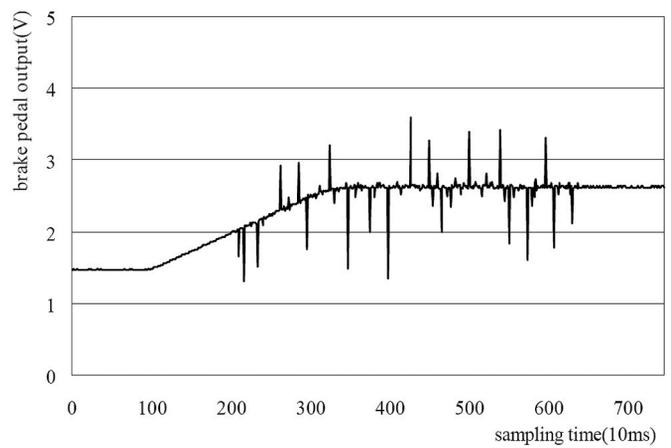


Fig. 6. IAE performance index for fault detection parameter (β) for varying fault values.

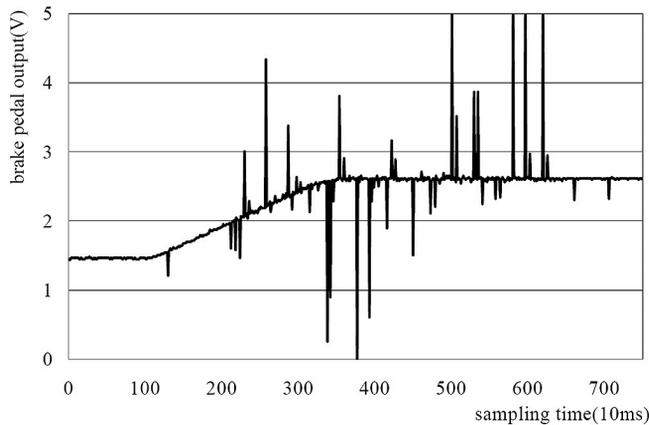
within a very short period of time due to white noise or impulse noise in electric circuit or sensor elements and includes transient and intermittent fault characteristics [13], [20]. The fault injector inserted a Gaussian-distributed random signal with a range of ± 5 V, a mean of 0, and standard deviation of 1, based on the number of injected faults. For example, sample 257 shown in Fig. 7(a) was 2.18 V, while the faulty value shown in Fig. 7(b) became 4.33 V by adding a noise voltage of 2.15 V.



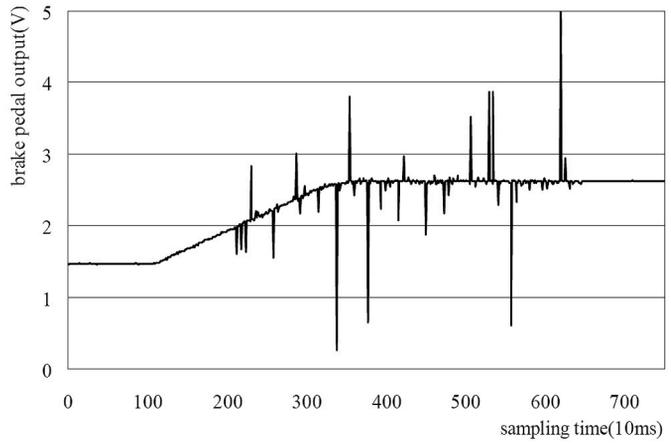
(a) brake pedal signal from a potentiometer without a fault



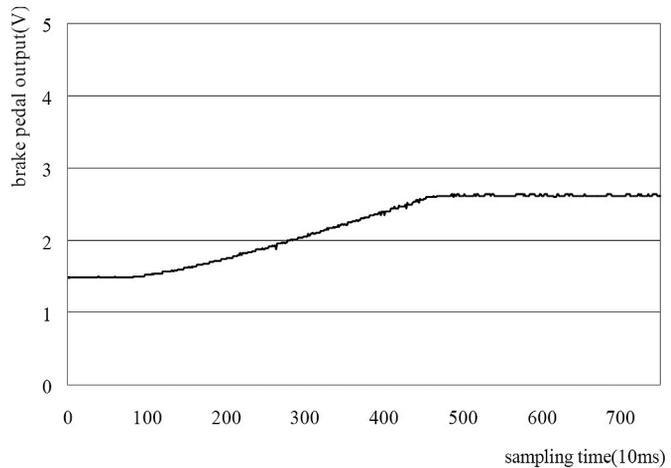
(a) pedal signal output of average voter



(b) brake pedal signal from a faulty potentiometer



(b) pedal signal output of median voter



(c) pedal signal output of Kalman predictive redundancy algorithm

Fig. 7. Examples of brake pedal signals from a potentiometer.

Fig. 8 shows the brake pedal signal of the average voter, median voter, and Kalman predictive redundancy algorithm with 4000 injected faults. Fig. 8(a) shows that after processing by the average voter using three different signals with faults, the output signal still included a great deal of instantaneous fluctuation compared to the original signal. This was because the average voter always used all the signals, even when a signal contained noise. Fig. 8(b) shows that the median voter partly eliminated the faults. The IAE performance index for the median voter (14.7) was better than that of the average voter (18.67), as shown in Fig. 9, but many faults still affected the output when two or more values were faulty. This experimental result shows that these two voters may not be suitable for fault tolerance of safety-critical systems like x-by-wire systems.

Fig. 8(c) shows the signal output of the Kalman predictive redundancy system. It appears almost identical to the original signal. As expected, the IAE performance index was 0.27, as shown in Fig. 9, which is far better than the values produced by the average and median voters. This experimental result indicates that the Kalman predictive 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.

Fig. 8. Brake pedal signal using average voter, median voter, and Kalman predictive redundancy algorithm for 4000 injected faults.

Fig. 9 shows the IAE performance indices of the average voter, the median voter, and the Kalman predictive redundancy algorithm for various numbers of injected faults. The IAE performance indexes of the average and median voters increased linearly to a maximum of 18.67 and 14.7, respectively. However, the IAE performance index of the Kalman predictive redundancy system remained relatively low, and reached a max-

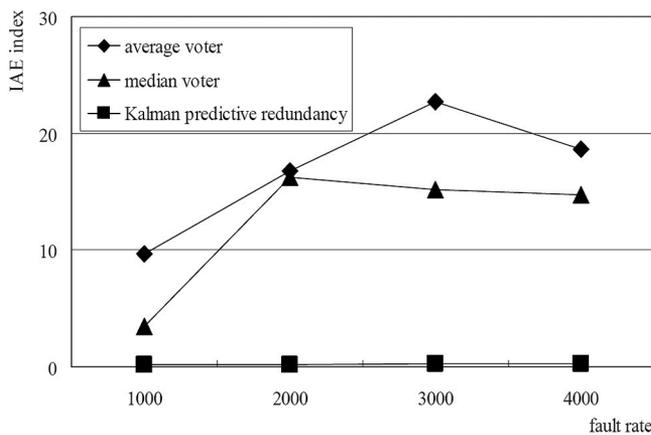


Fig. 9. IAE performance index for various numbers of injected faults.

imum of only 0.27. These results indicate that the performance of the Kalman predictive redundancy algorithm was superior to those of the average and median voters. In addition, it was possible to implement a redundant system more economically since the average and median voters used three sensors while only two sensors were necessary for the Kalman predictive redundancy. This may mean an overall reduction of the system cost because a sensor may cost more than the microprocessor required to execute the algorithm.

IV. SUMMARY AND CONCLUSION

This paper presents the Kalman predictive redundancy along with a fault detection algorithm using a Kalman filter for safety-critical systems such as the x-by-wire system in intelligent vehicles. To verify the feasibility of Kalman predictive redundancy, we developed an experimental Kalman predictive redundancy system using an embedded microcontroller unit with an experimental setup to simulate a redundant brake pedal signal, and compared the performance of the Kalman predictive redundancy with those of the average and median voters. The conclusions derived from this research are as follows.

First, the experimental results showed that the Kalman predictive redundancy using the Kalman filter could eliminate faults much better than the general voting method used in a triple modular redundancy system. The fault-masked signal was very similar to the original signal without faults because the exception handler determined a feasible output for safe operation even if both inputs were temporarily faulty. The experiment demonstrated that Kalman predictive redundancy could be very effective in safety-critical systems.

Second, since general voting methods such as average and median voters require at least three sensors, while the Kalman predictive redundancy system requires only two, it may be possible to implement a more flexible redundancy system more cost effectively. With inexpensive microcontrollers and better implementation technology, Kalman predictive redundancy can be used in various industrial systems that are sensitive to cost.

The natural extension of this research is to compare the performance of Kalman predictive redundancy with that of other

dynamic or hybrid methods, along with experimental demonstration. Also, other methods to handle prolonged faults in both sensors should be developed because the proposed method may be inadequate in such situations. In addition, it is necessary to implement overall redundancy system including sensor, actuator, and communication network for evaluating the performance of x-by-wire by using the Kalman predictive redundancy system.

REFERENCES

- [1] G. Cena, A. Valenzano, and S. Vitturi, "Advances in automotive digital communications," *Comput. Standards & Interfaces*, vol. 27, no. 6, pp. 665–678, 2005.
- [2] K. C. Lee, M. H. Kim, S. Lee, and H. H. Lee, "IEEE-1451-based smart module for in-vehicle networking systems of intelligent vehicles," *IEEE Trans. Ind. Electron.*, vol. 51, pp. 1150–1158, 2002.
- [3] A. Broggi, "Intelligent vehicle applications worldwide," *IEEE Intell. Syst.*, vol. 15, no. 1, pp. 78–81, 2000.
- [4] G. Leen and D. Heffernan, "Expanding automotive electronic systems," *Computer*, vol. 35, no. 1, pp. 88–93, 2002.
- [5] R. Isermann, R. Schwarz, and S. Stolz, "Fault-tolerant drive-by-wire systems," *IEEE Control Syst. Mag.*, vol. 22, no. 5, pp. 64–81, 2002.
- [6] M. H. Kim, S. Lee, and K. C. Lee, "Predictive hybrid redundancy using exponential smoothing method for safety critical systems," *Int. J. Control, Autom. Syst.*, vol. 6, no. 1, pp. 126–134, 2008.
- [7] O. Rooksa, M. Armbrusterb, A. Sulzmann, G. Spiegelberg, and U. Kiencke, "Duo duplex drive-by-wire computer system," *Reliability Eng. Syst. Safety*, vol. 89, no. 1, pp. 71–80, 2005.
- [8] H. K. Kim, H. T. Lee, and K. S. Lee, "The design and analysis of AVTMR (all voting triple modular redundancy) and dual-duplex system," *Reliability Eng. Syst. Safety*, vol. 88, no. 3, pp. 291–300, 2005.
- [9] K. G. Popstojanova and A. Grnarov, "N version programming with majority voting decision: Dependability modeling and evaluation," *Microprocess. Microprogram.*, vol. 38, no. 1, pp. 811–818, 1993.
- [10] M. D. Krstic, M. K. Stojcev, G. L. Djordjevic, and I. D. Andrejic, "A mid-value select voter," *Microelectron. Reliability*, vol. 45, no. 3, pp. 733–738, 2005.
- [11] G. L. Shabgahi, "A novel algorithm for weighted average voting used in fault tolerant computing systems," *Microprocess. Microsyst.*, vol. 28, no. 7, pp. 357–361, 2004.
- [12] G. Levitin, "Weighted voting systems: Reliability versus rapidity," *Reliability Eng. Syst. Safety*, vol. 89, no. 2, pp. 177–184, 2005.
- [13] B. W. Johnson, *Design and Analysis of Fault-Tolerant Digital Systems*. Reading, MA: Addison-Wesley, 1989.
- [14] G. Welch and G. Bishop, *An Introduction to the Kalman Filter*. Univ. North Carolina, Chapel Hill, NC, TR 95-041, 2006.
- [15] R. E. Kalman, "A new approach to linear filtering and prediction problems," *Trans. ASME-J. Basic Eng.*, pp. 35–45, 1960.
- [16] T. G. Lee, "Centralized Kalman filter with adaptive measurement fusion: Its application to a GPS/SDINS integration system with an additional sensor," *Int. J. Control, Autom. Syst.*, vol. 1, no. 4, pp. 444–452, 2003.
- [17] O. Ondel, E. Boutleux, G. Clerc, and E. Blanco, "FDI based on pattern recognition using Kalman prediction: Application to an induction machine," *Eng. Appl. Artif. Intell.*, vol. 21, no. 7, pp. 961–973, 2008.
- [18] Z. Zhou, C. Hu, H. Fan, and J. Li, "Fault prediction of the nonlinear systems with uncertainty," *Simulation Modeling Practice and Theory*, vol. 16, no. 6, pp. 690–703, 2008.
- [19] S. J. Kwon, "Kalman predictive filtering with perturbation estimation process for uncertain systems," *J. Control, Autom., Syst. Eng.*, vol. 12, no. 3, pp. 201–207, 2006.
- [20] S. Anwer and L. Chen, "An analytical redundancy-based fault detection and isolation algorithm for a road-wheel control subsystem in a steer-by-wire system," *IEEE Trans. Veh. Technol.*, vol. 56, no. 5, pp. 2859–2869, 2007.
- [21] K. Astrom and T. Haqqlund, *PID Controller: Theory, Design and Tuning*. Research Triangle Park, NC: International Society for Measurement and Control, 1995.



Man Ho Kim received the B.S. degree from Donga University, Busan, Korea, in 2001, and the M.S. and Ph.D. degrees from Pusan National University, Busan, in 2003 and 2008, respectively.

Since 2006, he has been working at the Daegu Gyeongbuk Institute of Science and Technology (DGIST) as a Researcher. His research interests include fault-tolerant system with hardware redundancy structure, in-vehicle networking system, embedded microcontroller system design, and driving behavior and workload analysis of driving vehicle.

Dr. Kim is a member of the Korean Society of Automotive Engineers, the Korean Society of Precision Engineers, and the Institute of Control, Robot, and Systems Engineers.



Suk Lee (M'92) received the B.S. degree from Seoul National University, Seoul, Korea, in 1984, and the M.S. and Ph.D. degrees from The Pennsylvania State University, University Park, in 1985 and 1990, respectively.

He is a Professor at the School of Mechanical Engineering, Pusan National University, Busan, Korea. Prior to joining Pusan National University, he was a Research Assistant Professor in the Center for Advanced Manufacturing Systems, University of Cincinnati, Cincinnati, OH. His research interests

are industrial network, in-vehicle network, and home network.

Dr. Lee is a member of the Korean Society of Mechanical Engineers, the Korean Society of Precision Engineers, and the Institute of Control, Robot, and Systems Engineers.



Kyung Chang Lee (S'00–M'04) received the B.S., M.S., and Ph.D. degrees from Pusan National University, Busan, Korea, in 1996, 1998, and 2003, respectively.

He is an Assistant Professor in the Department of Control and Automation Engineering, Pukyong National University, Busan. Prior to joining Pukyong National University, he was a Research Associate in the Network-Based Automation Research Center, University of Ulsan, Ulsan, Korea, from 2003 to 2004. From 1997 to 2003, he was a Research As-

sistant in the Mechanical Engineering and Technology Research Information Center, Busan. His research interests are embedded network systems, industrial networks, in-vehicle network, home network, wireless sensor network, and networked control systems.

Dr. Lee is a member of the Korean Society of Automotive Engineers and the Institute of Control, Robot, and Systems Engineers.