Available online at www.elixirpublishers.com (Elixir International Journal)

Electrical Engineering

Elixir Elec. Engg. 93 (2016) 39692-39697

ISSN: 2229-712X

Reduction of Area and Power Using ECSFD in Wireless Sensor Networks

Sweetlineshamini A, B.Elizabeth Caroline, R.Shalini and R.Krishnaveni
Department of Electronics and Communication Engineering, J.J. College of Engineering and Technology, Trichy.

ARTICLE INFO

Article history:
Received: 25 July 2012;
Received in revised form:
14 April 2016;
Accepted: 20 April 2016;

Keywords
Distributed estimation,
Fusion,
Sensor fault detection,
VLSI architecture,
Wireless sensor networks.

ABSTRACT
Collaborative Sensor Fault Detection (CSFD) technique is used to detect the faults in the sensor nodes of a fusion center in a robust distribution estimation scheme. This technique can identify the faulty nodes efficiently and improve the accuracy of the estimates. ASIC implementation is used to design the fusion center based on their applications. In this approach there occurs some limitations such as computational complexity; hardware implementation is quite complex and high power consumption. In order to overcome these difficulties, we modify CSFD and propose an ECSFD scheme. ECSFD is simple and requires lower computational complexity, thus lower hardware cost and power consumption can be achieved. Furthermore, ECSFD achieves almost the same performance as CSFD. VLSI architecture is developed for hardware implementation.

Introduction

As Wireless Communications technology and microelectromechanical systems (MEMS) techniques have matured in recent years, wireless sensor networks (WSN) have emerged as a promising solution for a variety of remote sensing applications, including battlefield surveillance, environmental monitoring, intruder detection systems, weather forecasting, health care, agricultural technology, and so on. Irrespective of their purpose, all WSN are characterized by the requirement for energy efficiency, scalability, and fault tolerance [1]. These requirements are particularly crucial in sensor networks designed to perform an estimation function. The fusion center makes the distributed estimation based upon the information received from the local nodes. In such networks, the estimation performance is critically dependent upon the availability and reliability of the local information, and substantial errors are induced if the nodes become unavailable (e.g., as a result of consuming all their energy) or unreliable (e.g., as a result of intermittent malfunctions). Hence, the design of a robust distributed estimation for fusion center in WSN is essential.

The problem of distributed estimation systems have attracted significant interest in recent years [2]–[5]. The research focuses principally on the problem of developing energy-efficient and bandwidth-constrained designs. By contrast, the problem of enhancing the fault tolerance capability of decentralized estimation systems has attracted relatively little attention. In practical networks, fault tolerance is a critical concern since the sensor nodes are invariably battery-powered and randomly deployed, and are therefore not easily recharged or replaced. Furthermore, the sensors are generally deployed in outdoor or similarly harsh environments, and thus the occurrence of sensor failures or malfunctions is inevitable. To solve the problem, we have proposed a collaborative fault detection (CSFD) scheme [6] to detect the faulty nodes within the network such that their quantized messages can be excluded from the parameter estimation process.

Some related works about variants of enhancing the fault tolerant capability of decentralized estimation systems have been considered in the following literature. I. Rapoport *et al.* [7] addressed the problem of sensor fault detection and estimation in dynamic systems using an *a priori* sensor-fault model. Meanwhile, Delouille *et al.* [8] used an embedded subgraphs algorithm to design a robust distributed estimation scheme for sensor networks in which the sensors observe different physical phenomena. The scheme considers only temporary communication faults such as failing links and sleeping nodes, whereas the robust CSFD estimation scheme proposed considers all manner of possible sensor failures. Ishwar *et al.* [9] utilized a packet-erasure model to examine various aspects of distributed estimation inWSN, including its robustness toward sensor unreliability, its power-cycling characteristics, and the effects of uncertainties in the wireless transmissions. However, the estimation problem assumes that the fusion center requires the ability to discriminate between the local messages received from normally operating nodes and those messages received from faulty nodes.

In [6], CSFD takes the concept of collaborative signal processing to perform robust distributed estimation. Specifically, this work employs the homogeneity test [10] to implement CSFD scheme to detect the faulty nodes within the network such that their quantized messages can be excluded from the parameter estimation process. Utilizing the proposed CSFD mechanism, the fusion center identifies the faulty nodes with the WSN and then excludes theirs information when estimating the parameter of interest. With the aid of CSFD scheme, different sensor faults can be tolerated to improve the performance of estimating the parameter of interest. As predicted, CSFD performs better than the conventional approach in estimating theta in terms of different sensor faulty types and faulty number.

Tele:
E-mail address: rckrishnaveni@gmail.com

In the detecting process, CSFD requires such extensive computations as logarithm and division though it achieves very good performance. In many real-time WSN applications, the fusion center might be implemented with the ASIC and included in a standalone device, so a simple and good distributed estimation scheme of lower computational complexity is extremely desired.This motivation makes us modify CSFD and propose an efficient collaborative sensor fault detection (ECSFD) scheme and its VLSI architecture in this paper. Compared with CSFD, ECSFD performs slightly better and requires only about 55% of computations. Therefore, it does qualify as a good candidate for hardware implementation.

To our knowledge, ECSFD circuit is the first ASIC implementation for fault-tolerance fusion center for distributed estimation and no related state-of-the-art ASIC design exists in the literature.

Overview Of Csfd

Fig. 1 illustrates the basic structure of the distributed estimation network considered in the present work.The Bayesian formulation is considered here. LetS = { $s_1, ..., s_N$ } be a finite set corresponding to the N sensor nodes observing sensor measurement sequences generated from a common status of phenomenon $\theta \in \Theta$, the parameter under estimation.

It is assumed that the distribution of θ is known and is denoted by p(θ). The observation sequences taken by sensor s_n are denoted by $\{x_n^t\}_{t=1}^{\infty}$, where n is the node index and t is the time index. Every sensor node quantizes its own observations x_n^t to output m_n^t and send it to the fusion center. The local messages m_n^t are mapped to a binary signal vector b_n^t = ($b_{n1}^t, ..., b_{nG}^t$) where G =|$\log_2$ M| is the number of bits used to represent the local message and M is the number of partition levels at the local sensors.

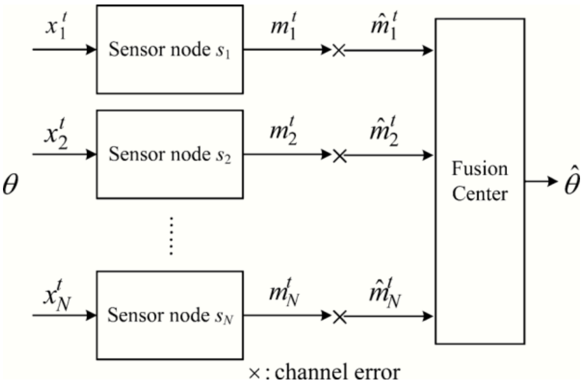


Fig 1: System Model For Defs

In the distributed estimation network shown in Fig. 1, two types of errors may affect the received quantized messages at the fusion center. The first error is caused by the faulty node. The considered WSN herein is very possible to contain faulty nodes because of random deployment in a harsh environment. The second error is the channel transmission error due to interference or noise. In this situation, the received $\hat{m}_n^t$ at the fusion center may not be equal to m_n^t and we denote Pr[$b_{nj}^t \neq \hat{b}_{nj}^t$] by ε for all t and n .

Consider the case where the fusion center estimates θ at some arbitrary time T . Note that in performing this estimation process, all the messages received from the local nodes up to time T. If sensor faults exist within the network, the estimated value of θ is liable to deviate significantly from the true value. To solve the problem, CSFD adopts the concept of collaborative signal processing to identify the faulty nodes.

In CSFD, the following sensor fault models are considered in order to include different misbehavior. Given M partition levels { q_k = 1, . . . ,M } for the quantizer, then we denote $p_{i|\theta}$ = Pr[$\hat{m}_n^t$ = $q_i|\theta$] when node s_n operates in a fault-free manner.
In one fault model, the output of local quantizers is independent of the parameter θ .

The process of CSFD can be divided into three stages. The first stage is to measure the faulty weights of all N nodes. Then, the faulty nodes are determined. The final distributed estimate is generated in the last stage. The detail of each stage is described as follows.

Measuring Faulty Weight: This stage consists of two steps and its aim is to decide the faulty weight of each node. The faulty weight is used to measure the deviation of a node. In the first step, we compute the number of q_i received from sensor s_n and denoted it as o_{ni}.

$$o_{ni} = \sum_{t=1}^{T} 1\{\hat{m}_n^t = q_i\} \quad (1)$$

Where 1{.} is the indicator function.

As mentioned in [17], the Kullback–Leibler (K-L) distance between distributions can be used to measure sensor-fault deviation. In CSFD, we use K-L distance to estimate the faulty weights of all sensors. According to the local decisions $\{\{\hat{m}_n^t\}_{t=1}^{T}\}_{n\in S}$, the K-L distance ED_n for node s_n is employed to measure the distribution distance from average sensor weight (1/N) $\sum_{n=1}^{N} \hat{m}_n^t$ to faulty sensor weight $\hat{m}_n^t$, and is defined as

$$ED_n\left(\hat{v}_i \,||\, \hat{r}_{ni}\right) = \sum_{i=1}^{M} \log \frac{\hat{r}_{ni}}{\hat{v}_i} \quad (2)$$

Where

$$\hat{r}_{ni} = \frac{o_{ni}}{T} \quad , \quad \hat{v}_i = \frac{\sum_{n=1}^{N} o_{ni}}{NT} \quad (3)$$

Determinining Faulty Nodes: The aim of this stage is to decide which sensor nodes are faulty, based on the faulty weights computed in the previous stage. First, all sensor nodes are sorted in descending order based on their magnitude of $\{EDn\}_{n=1}^{N}$ to get the faulty-weight-oriented sequence F = {$s_{(1)}$,$s_{(2)}$ . . . , $s_{(N)}$ }. After F is determined, we can obtain the candidate set of faulty sensors, denoted as $F_{(z)}$ = {$s_{(1)}$,$s_{(2)}$ . . . , $s_{(z)}$ } where z is the possible number of faulty nodes, and let $F_{(0)}$ represent the empty set Φ . In order to determine the value of , the following homogeneity testing problem can be formulated to test for the existence of a set of sensor nodes F_T at time T.

$$H_0 : \Pr[\hat{m}_n^t = q_i|\theta] = p_{i|\theta} \quad \text{for all } s_n \in S \backslash F_T; \quad (4)$$
H_1 : otherwise.

Then, the following statistics are utilized for homogenity testingto determine whether or not a candidate set of sensor nodes $\tilde{F}_T$ is F_T

$$H(S \backslash \tilde{F}_T) = \sum_{n=1}^{N} 1\left\{s_n \in S \backslash \tilde{F}_T\right\} \sum_{i=1}^{M} \frac{\left(o_{ni} - e_i\right)^2}{e_i} \quad (5)$$

Where

$$e_i = \frac{\sum_{n=1}^{N} 1\left\{s_n \in S \setminus \tilde{F}_T\right\} o_{ni}}{N - |\tilde{F}_T|}$$

Utilizing the statistic H(S\$\tilde{F}_T$), the binary hypothesis testing problem shown below

$$H_0 : H(S \setminus \tilde{F}_T) < \chi^2_{1-\alpha,(N-|\tilde{F}_T|-1)(M-1)}$$
$$H_1 : H(S \setminus \tilde{F}_T) > \chi^2_{1-\alpha,(N-|\tilde{F}_T|-1)(M-1)} \quad (6)$$

where $\chi^2_{1-\alpha,(N-|\tilde{F}_T|-1)(M-1)}$ is a threshold indicating the critical value of the chi-square distribution with (N- $\tilde{F}_T$ -1) (M-1) degrees of freedom at a significance level α .

Making Distributed Estimation: Once the set of faulty nodes $\hat{F}_{(z)}$ is determined, the fusion center removes the quantized messages of the faulty nodes and performs the parameter estimation. Then, the estimate obtained by minimum mean square error (MSE) criterion is adopted and is given by

$$\hat{\theta}^{(T)}_{(MSE)} = E\left[\theta \,|\, \left\{\left\{\hat{m}^t_n\right\}_{n\in S\setminus F_T}\right\}^T_{t=1}\right]$$

Efficient CSFD

CSFD performs better than the conventional approach with regard to fault tolerance. However, there are three difficulties to be overcome for implementing CSFD with a VLSI circuit.

The first one is that it requires some extensive and complex computations, such as logarithm and division in the detecting process (see (2)–(6)). The second difficulty is that the integration required for the estimate of in (7) is quite complex. The last difficulty is that the calculation of numerical integration needs many bits. In order to overcome these difficulties, we modify CSFD and propose an efficient collaborative sensor fault detection (ECSFD) scheme in this paper. ECSFD is simple and requires lower computational complexity, thus lower hardware cost and power consumption can be achieved. Furthermore, ECSFD achieves almost the same performance as CSFD. The details of ECSFD are described in the following.

A. Avoid the Logarithm and Division Operations

To avoid the logarithm and division operations required in (2), a simple and efficient sensor faulty weight estimate method is provided. We take advantage of collaborative signal processing to estimate the sensor faulty weight. More concretely, without knowing the true distribution of θ , most nodes in the networks can be reasonably assumed to normally report their decisions inferring the true distribution of θ to the fusion center. If the sensor behaviour $\hat{m}^t_n$ deviates from the average sensor behaviour (1/N) $\sum_{n=1}^{N} \hat{m}^t_n$ more obviously, the sensor s_n has larger faulty sensor weight. Hence, the faulty weight of the sensor nodes can be estimated by the sum of the absolute differences between $\hat{v}_i$ and $\hat{r}_{ni}$.

$$ED'_{(n)} = \sum_{i=1}^{M} |\hat{v}_i - \hat{r}_{ni}|$$
$$= \sum_{i=1}^{M} \left| \frac{\sum_{n=1}^{N} o_{ni}}{NT} - \frac{o_{ni}}{T} \right| \quad (8)$$

Besides, the final purpose of this stage is to calculate the $ED'_{(n)}$ for obtaining the faulty-weight-oriented sequence F . By multiplying all $ED'_{(n)}$ with a constant (NT) simultaneously, we can further reduce the computational complexity of $ED'_{(n)}$ Without $\hat{F}_{(z)}$ affecting the decided . Finally, the $ED'_{(n)}$ can be estimated with less computational complexity and is given as

$$ED''_{(n)} = \sum_{i=1}^{M} \left| \sum_{n=1}^{N} o_{ni} - N o_{ni} \right| \quad (9)$$

In order to overcome the problem of massive division , the hypothesis testing can be rewritten in the following formation by multiplying (6) with a constant:

$$H_0 : H(S \setminus \tilde{F}_T) \prod_{i=1}^{M} e_i < \chi^2_{1-\alpha,(N-|\tilde{F}_T|-1)(M-1)} \prod_{i=1}^{M} e_i$$
$$H_1 : H(S \setminus \tilde{F}_T) \prod_{i=1}^{M} e_i > \chi^2_{1-\alpha,(N-|\tilde{F}_T|-1)(M-1)} \prod_{i=1}^{M} e_i \quad (10)$$

Substituting (5) to (10) gives

$$H_0 : \sum_{n=1}^{N} 1\left\{s_n \in S \setminus \tilde{F}_T\right\} \sum_{i=1}^{M} \prod_{j=1, j\neq i}^{M} c_j (N' o_{ni} - c_i)^2$$
$$< \chi^2_{1-\alpha,(N-|\tilde{F}_T|-1)(M-1)} N' \prod_{i=1}^{M} c_i$$
$$H_1 : \sum_{n=1}^{N} 1\left\{s_n \in S \setminus \tilde{F}_T\right\} \sum_{i=1}^{M} \prod_{j=1, j\neq i}^{M} c_j (N' o_{ni} - c_i)^2$$
$$> \chi^2_{1-\alpha,(N-|\tilde{F}_T|-1)(M-1)} N' \prod_{i=1}^{M} c_i \quad (11)$$

Where N'=N-$|F_T|$ and $c_i = \sum_{n=1}^{N} 1\left\{s_n \in S \setminus \tilde{F}_T\right\} o_{ni}$

The required division operation in (5) is replaced with multiplication and the corresponding computational complexity cost can be reduced. Using (9) and (11), we can choose $\hat{F}_{(z)}$ the according to the step of determining faulty nodes of CSFD scheme listed in Section II.

B. Simplify the Integration

However, minimum MSE in (7) needs integral operation which is difficult for hardware implementation. Therefore, the numerical integration is used in the stage of making distributed estimation. (7) can be written in the following form:

$$\hat{\theta}^{(T)}_{(MSE)} = E[\theta \,|\, m^t_n \in S \setminus F_T]$$
$$= \int \theta P(\theta \,|\, m^t_n \in S \setminus F_T) d\theta$$

$$= \frac{\int_{\theta=-\infty}^{\infty} P(m_n^t \in S \setminus F_T \mid \theta) P(\theta) \theta d\theta}{\int_{\theta=-\infty}^{\infty} P(m_n^t \in S \setminus F_T \mid \theta) P(\theta) d\theta} \quad (12)$$

Where

$$P(m_n^t \in S \setminus F_T \mid \theta) = \prod_{m_n^t \in S \setminus F_T} f(m_n^t = q_i \mid \theta) \quad (13)$$

In the issue of wireless communication, the additional noise model can be reasonable assumed as a Gaussian function $(0, \sigma_\omega^2)$. Therefore $f(m_n^t = q_i \mid \theta)$, can be given as

$$f(m_n^t = q_i \mid \theta) = \int_{x_n^t \in L_i} \frac{1}{\sqrt{2\pi\sigma_\omega^2}} e^{\frac{-(x_n^t - \theta)^2}{2\sigma_\omega^2}} dx_n^t \quad (14)$$

Where L_i denotes the quantized range of x_n^t.

Let $f_i^\theta = f(m_n^t = q_i \mid \theta)$ and Q_i denote the number of q_i received from S $\setminus F_T$ in the fusion center . Then $\hat{\theta}_{(MSE)}^{(T)}$ can be calculated by the following equation :

$$\hat{\theta}_{(MSE)}^{(T)} = \frac{\int_{-\infty}^{\infty} \prod_{i=1}^{M} (f_i^\theta)^{Q_i} P(\theta) \theta}{\int_{-\infty}^{\infty} \prod_{i=1}^{M} (f_i^\theta)^{Q_i} P(\theta)} \quad (15)$$

Using the numerical integration , above equation can be approximated by integrating θ from –a to b with an interval ω

$$\hat{\theta}_{(MSE)}^{(T)} = \frac{\sum_{\theta=-a}^{b} \prod_{i=1}^{M} (f_i^\theta)^{Q_i} P(\theta) \theta}{\sum_{\theta=-a}^{b} \prod_{i=1}^{M} (f_i^\theta)^{Q_i} P(\theta)} \quad (16)$$

In addition, the value of a, b, c and ω are decided according to the prior distribution of θ.

C. Transform the Numerical Integration

However, the bit width required for the numerical representations of the numerator and the denominator in (16) are quite large when Q_i is large enough. With the aid of logarithm property, we transform $\sum_{\theta=-a}^{b} \prod_{i=1}^{M} (f_i^\theta)^{Q_i} P(\theta) \theta$ and $\sum_{\theta=-a}^{b} \prod_{i=1}^{M} (f_i^\theta)^{Q_i} P(\theta)$ to log $\sum_{\theta=-a}^{b} \prod_{i=1}^{M} (f_i^\theta)^{Q_i} P(\theta) \theta$ and log $\sum_{\theta=-a}^{b} \prod_{i=1}^{M} (f_i^\theta)^{Q_i} P(\theta)$, which need smaller bit width, respectively. Hence, (16) can be rewritten as

$$\hat{\theta}_{(MSE)}^{(T)} = \frac{\sum_{\theta=-a}^{b} 2 \sum_{i=1}^{M} (Q_i x \log_2 f_i^\theta) + \log_2 P(\theta) + \log_2 \theta}{\sum_{\theta=-a}^{b} 2 \sum_{i=1}^{M} (Q_i x \log_2 f_i^\theta) + \log_2 P(\theta)} \quad (17)$$

$$= \frac{2^{D_1} + 2^{D_2} + \ldots + 2^{D_i}}{2^{G_1} + 2^{G_2} + \ldots + 2^{G_i}} \quad (18)$$

Then, all the items of the numerators and denominators are sorted to find the one with the maximum exponent denoted as2^{Max}. According to the found value, all the other items which satisfy Max – D_i < 10 or Max – G_i < 10 $\hat{\theta}_{MSE}^{(T)}$ are selected to calculate the value of approximated. With the aid of the logarithm and the sorting process, the can be calculated efficiently.

Chip Architecture For Ecsfd

Observing the required operations in ECSFD, we develop a low-cost VLSI architecture for ECSFD where N and $|N_F|$ is setas 8 and 3, respectively, in the current implementation. This setting, as mentioned in [6], is suitable for general applications in WSN. Furthermore, the word length of signals is decided based on the following two considerations:

a) The performance of ECSFD circuit must be comparable to that of CSFD.

b) The hardware cost of ECSFD circuit must be minimized. After careful analysis and software simulation, we have chosen the 11-bit widths for representing different signals in the ECSFD circuit to meet the precision requirement and maintain the acceptable performance. The VLSI architecture of ECSFD consists of a logarithm unit, anti-logarithm unit, sort unit, register file, 11 11multiplier unit, comparator unit, and adder/subtractor unit connected to a shared bus. A top-level FSM coordinates the operations among these functional units.

A. Multiplication

Since the largest width of the signals in ECSFD is 11-bit, a basic 11x 11 multiplier is developed where the multiplier is denoted as A , the multiplicand is denoted as B , and the product is denoted as C . Many multiplication operations are required in ECSFD. Since the width of most signals is 11-bit, we need the 11x 11 multiplier. These multiplication operations are performed sequentially at different time instant, so we can apply the concept of hardware resource sharing and design special-purpose multipliers (11x 22, 22x 22, and 22x 33) to implement them. Hence, we utilized the 11 x11 multiplier to realize the four different multiplications where the multiplier, multiplicand and product are all realized with different bit widths of integer and fractional parts for respective precisions.

For most WSN applications, the cost issue is more important than timing performance in the design of fusion center. Hence, the 11x 22, 22x 22, and 22x 33 multipliers are realized with a normal 11 x11 multiplier circuit (multiplying two 11-bit operands to produce a 22-bit product) and a dedicated control circuit under multicycle implementation to reduce the hardware cost. With the help of the control circuit, the 11x 11 multiplier can implement all the required multiplication operations for different modes with multiple clock cycles.

B. Logarithm and Antilogarithm

As shown in (17), some logarithm and antilogarithm conversion operations are required in ECSFD. Let L and Y represent the input and output, thus the logarithm conversion can be denoted as $Y=\log_2(L)$ where L is the 22-bit input and Y is the converted 11-bit output. The reason of using $\log_2$ is to match the binary representation. Using a proper lookup table, we can implement the logarithm conversion with a dedicated control circuit.

In our implementation, the prior distribution of θ is a Gaussian function (0, 0.5), the range of the integration is from -3 to 3, the integral interval,ω, is set as 0.05, and M=4 .Hence, the lookup ROM table is constructed with 121x 6 entries. The antilogarithm conversion operations are also performed based on a lookup table. Let V and W represent the input and output, thus the antilogarithm conversion can be denoted as $V=2^W$, where W is the 10-bit input and V is the converted 22-bit output. The exponents of the selected items (D_i ,G_i) in (18) are normalized (subtracted by a common constant) to the range from 0.00 to 10.00 without affecting the approximated θ value.

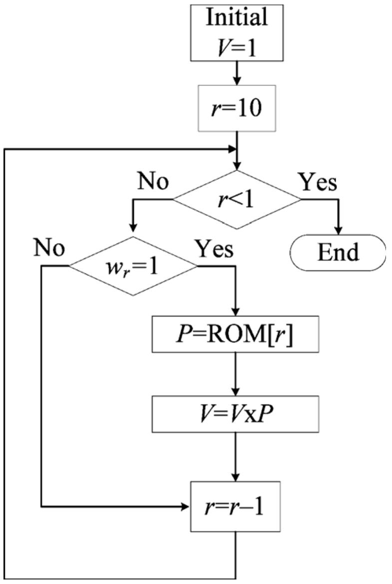


Fig 2: flow chart of antilogarithim conversion.

Fig.2 flow chart of antilogarithm computation in our design. We use 10-bit to represent W ($w_{10}w_9...w_2w_1$) . The lookup table is constructed with 10 entries and each stores the 17-bit value of $2^{2^r/100}$, where r is an integer to represent the position number and $0 \leq r \leq 100$. The 22 x22 multiplier is accessed 0 to 9 times to get the result of 22-bit . After getting the values of numerators and denominators in (18) through the antilogarithm module, can be calculated by a divider. Finally, the division required in (18) is replaced by repeated subtractions to reduce the hardware cost.

C. Sorting

In ECSFD, the faulty weights of sensors are represented as $ED_1'', ED_2, ...and'' ED_8''$.$|N_F|$ is 3 in the current implementation, so we need to find the three biggest values from these eight numbers and identify their node indexes for the following usage. The order of the other five smaller values is not important.

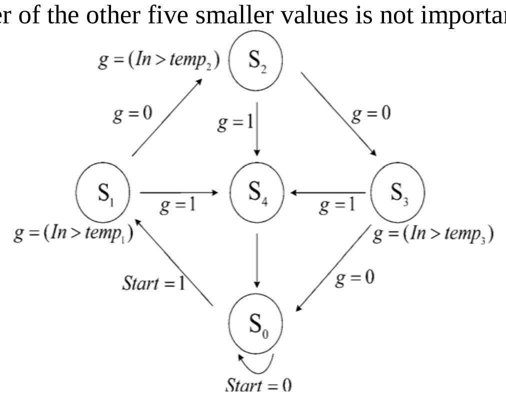


Fig 3: State Diagram Of Sort Module

Since $ED_1'', ED_2, ...and'' ED_8''$ are calculated and generated sequentially in the previous stage, we design a special purpose insertion sorting circuit which maintains both the three bigger values and their indexes through the whole sorting procedure. The state diagram of sort module is shown in Fig. 3.The calculated data are inputted to the sort module one by one in turn. The start signal will initialize the three registers with a small number and enable the sorting procedure. The current input is compared with the values in $\{temp_n\}^3_{n=1}$ one by one from S_1 to S_3 , and the control signal g will be set as 1 if the input is larger than $temp_n$. As soon as g=1 , the replacing procedure at S_4 is performed to save the current input to a proper register $temp_n$. Thus, the input value can be inserted to an appropriate position and $temp_1 \geq temp_2 \geq temp_3$ is satisfied. The sort module spends 1 to 3 clock cycles to find the appropriate position for inserting the input, and the replacing operation needs another 1 clock cycle. The corresponding sensor numbers are $\{temp_{(n)}\}^3_{n=1}$ recorded in the three registers and will be finally outputed.

Implementation Results and Evaluation

The proposed VLSI architecture of ECSFD was implemented by using Verilog HDL.

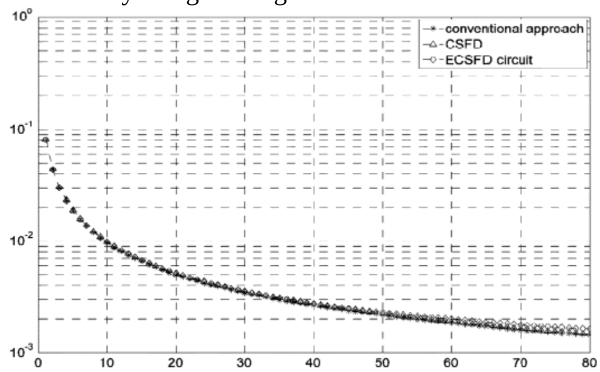


FIG 4: Performance comparison of CSFD, ECSFD circuit, and the conventional approach in a fault-free WSN.

Fig. 4 compares the estimation performance of CSFD, ECSFD, and the conventional scheme for the case in which all of the sensors within the network are fault-free. It is evident that the MSE values of CSFD and ECSFD are virtually identical to those of the conventional scheme, implying that CSFD and ECSFD have exceedingly small possibility to remove the normally operating nodes.

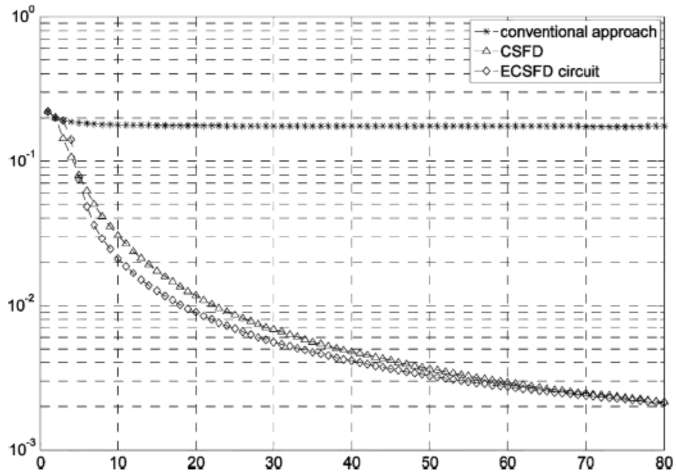
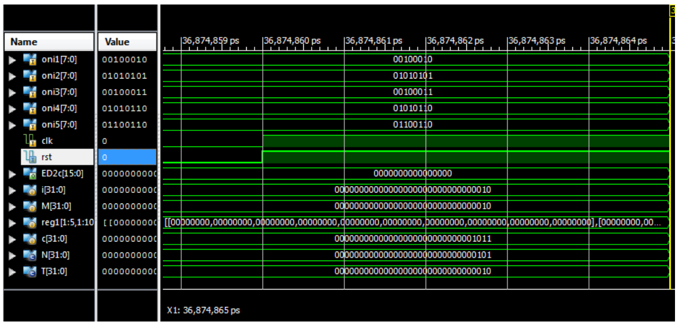
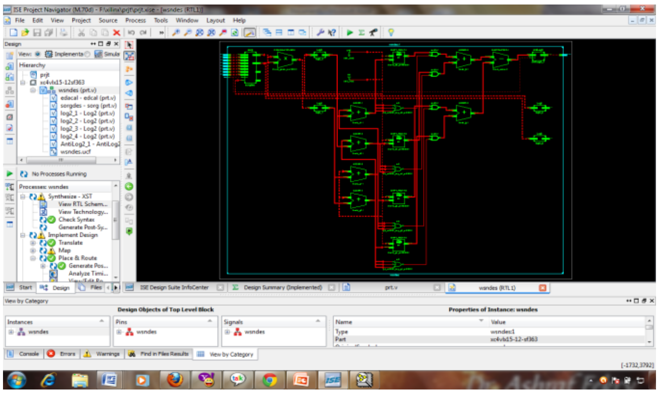
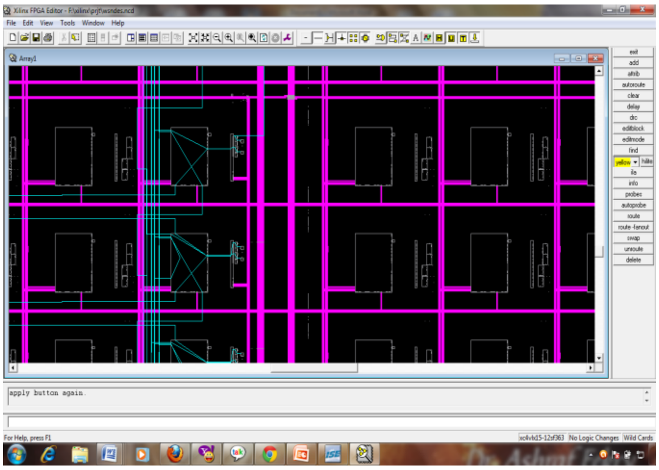


FIG 5: Performance comparison of CSFD, ECSFD circuit, and the conventional approach in a WSN with two sensors with stuck-at faults.

Fig. 5 compares the estimation performance of the three schemes when two of the nodes within the WSN experience stuck-at-zero faults, that the two faulty nodes are drawn uniformly from the eight nodes within the network. The results confirm that both robust estimation schemes result in a significantly lower MSE than that obtained using the conventional approach. Moreover, ECSFD performs slight better than CSFD.







Conclusion

The ECSFD is designed in order to reduce the computational complexity required for CSFD in this paper. Based on ECSFD, a low cost VLSI architecture is proposed for fault-tolerant fusion center in WSN. With the multicycle structure, the proposed VLSI architecture can work fast enough to provide the real-time operation but only needs a low hardware cost. According to the performance evaluation, the VLSI architecture for ECSFD can work better than the conventional approach and its performance is close to that of CSFD.

Reference:

[1] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, “A survey on sensor networks,” *IEEE Commun. Mag.*, vol. 40, no. 8, pp. 102–114, Aug. 2002.

[2] Y. Zhu, E. Song, J. Zhou, and Z. You, “Optimal dimensionality reduction of sensor data in multisensor estimation fusion,” *IEEE Trans. Signal Process.*, vol. 53, no. 5, pp. 1631–1639, May 2005.

[3] R. Niu and P. K. Varshney, “Target location estimation in sensor networks with quantized data,” *IEEE Trans. Signal Process.*, vol. 54, no. 12, pp. 4519–4528, Dec. 2006.

[4] A. Ribeiro and G. B. Giannakis, “Bandwidth-constrained distributed estimation for wireless sensor networks—Part I: Gaussian case,” *IEEE Trans. Signal Process.*, vol. 54, no. 7, pp. 2784–2796, Jul. 2006.

[5] A. Ribeiro and G. B. Giannakis, “Bandwidth-constrained distributed estimation for wireless sensor networks—Part II: Unknown probability density function,” *IEEE Trans. Signal Process.*, vol. 54, no. 7, pp. 2784–2796, Jul. 2006.

[6] T.-Y. Wang, L.-Y. Chang, and P.-Y. Chen, “A collaborative sensorfault detection scheme for robust distributed estimation in sensor networks,” *IEEE Trans. Commun..*, vol. 57, no. 10, pp. 3045–3058, Oct.

[7] I. Rapoport and Y. Oshman, “A new estimation error lower bound for interruption indicators in systems with uncertain measurements,” *IEEE Trans. Inf. Theory*, vol. 50, no. 12, pp. 3375–3384, Dec. 2004.

[8] V. Delouille, R. N. Neelamani, and R. G. Baraniuk, “Robust distributed estimation using the embedded subgraphs algorithm,” *IEEE Trans. Signal Process.*, vol. 54, no. 8, pp. 2998–3010, Aug. 2006.

[9] P. Ishwar, R. Puri, K. Ramchandran, and S. S. Pradhan, “On rateconstrained distributed estimation in unreliable sensor networks,” *IEEE J. Sel. Areas Commun.*, vol. 23, no. 4, pp. 765–775, Apr. 2005.

[10] R. C. Elandt-Johnson, *Probability Models and Statistical Methods in Genetics*. New York: Wiley, 1971.

[11] S. Diao, Y. Zheng, and C.-H. Heng, “A CMOS ultra low-power and highly efficient UWB-IR transmitter for WPAN applications,” *IEEE. Trans. Circuits Syst. II, Exp. Briefs*, vol. 56, no. 3, pp. 200–204, 2009.

[12] M. Verhelst andW. Dehaene, “Analysis of the QAC IR-UWB receiver for low energy, low data-rate communications,” *IEEE. Trans. Circuits Syst. I, Reg. Papers*, vol. 55, no. 8, pp. 2423–2432, 2008.