Diverse Sorting Algorithm Analysis for ACSFD in Wireless Sensor Networks

Jose Anand, K. Sivachandar

Abstract - The progression in wireless communications and electronics has emerged the expansion of low-cost sensor networks. Sensor networks exploit large number of wireless sensor nodes to collect information from their sensing terrain. The gathered information will undergo in-network process and send to the remote sink. Sensor networks have wide range of applications including medical, military, home, etc. In the sensor networks, a fault-tolerant distributed decision fusion will occur due to the presence of sensor faults. For this fault detection, Collaborative Sensor Fault Detection (CSFD) scheme was used and this fault detection scheme is very difficult to implement because of the extensive computations. So an Approximated Collaborative Sensor Fault Detection (ACSFD) scheme was developed, which is less cost and utilizes less power than CSFD and has the same performance of CSFD. The important blocks present in the architecture of ACSFD consist of multipliers, logarithms, and sorting. In this paper, analysis has been done with various sorting algorithms and concluded the best sorting technique that can be used in ACSFD scheme to improve the performance of the fault detection scheme in the wireless sensor network.

Keywords – ACSFD, fault detection, sorting algorithms, wireless sensor networks.

I. INTRODUCTION

Wireless Sensor Networks (WSNs) employ enormous amount of wireless sensor nodes to collect information from their sensing terrain. The gathered information will undergo in-network process and are communicated to the remote sink [6]. The wireless sensor nodes in the WSN are compact, light-weighted, and are battery-powered devices that can be used in almost any environment. With this data, simple computations are carried out and communication with other sensor nodes or controlling authorities in the network will take place [8]. WSN have recently engrossed in plenty of applications which include environment monitoring such as temperature, sound, pressure, vibration, pollutants, oxygen content, carbon dioxide content, sulphur dioxide content, etc. at different locations, mobile object tracking, and navigation applications. All these applications consist of many inexpensive wireless sensor nodes that are proficient of collecting, processing and storing various information in a prompt manner. In wireless sensor applications all the sensor nodes will periodically report the collected information to a single sink node which realizes a many-to-one communication network model [3]. The sensor nodes are organized as a cooperative network and the structure of a node is shown in figure 1.

Manuscript Received on February, 2013.

Jose Anand, Associate Professor, Department of Electronics and Communication Engineering, K C G College of Technology, Karapakkam, Chennai, Tamil Nadu, India.

K. Sivachandar, Associate Professor, Department of Electronics and Communication Engineering, K C G College of Technology, Karapakkam, Chennai, Tamil Nadu, India.

Each node has controllers for processing, multiple types of memories like program, data, and flash, RF transceivers, power source, and various sensors and actuators. The sensor nodes communicate wirelessly and often self-organize after being deployed in an ad hoc fashion.

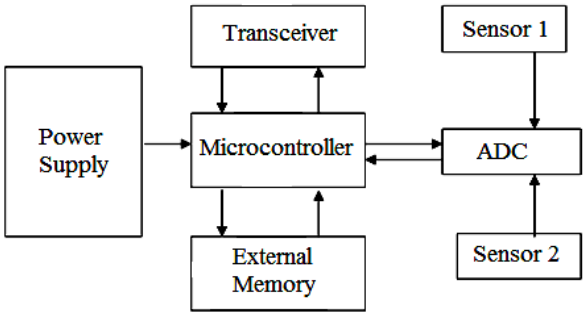

Figure 1 Structure of Sensor Node

Packets transmitted in the WSN contain useful information, which can be utilized through packet-based computation and to reduce the sensor fault rate [6]. The WSN packet computation has small packet forwarding rate and the forwarding computation capability is limited. Mostly the sensor nodes are modeled with limited energy, as a result the sensor nodes lacks recharging issues. But still wireless nodes packet-based computation is preferred since it is generally known that the computation utilizes reduced energy than the communication [1]. To attain the Quality of Service (QoS) requirements, the network resources should be used in a fair and efficient manner. Moreover, techniques such as data compression, data fusion and aggregation become very useful in maintaining robustness. Due to the changes in node mobility and wireless channel or node failure, the WSN seems to be unreliable in nature. In order to efficiently use the WSN for real-time applications the issues related to the wireless protocols are reduced [6]. As a result, an efficient design of distributed fault-tolerant fusion center is important in the WSN.

The sequential testing to judge the fusion center for WSN requires error probability and decisions are made on the comparison with the threshold value on the requirement [2]. The decisions taken by the fusion center are delayed and are reported at regular time interval to select the correct strategy. CSFD scheme is used for isolating the faulty nodes in the WSN and to eliminate the unreliable local decisions [7].

The CSFD scheme will make decision fusion in the WSN at regular time interval and meet the real-time requirements. The scheme identifies the faulty sensor node but the failure on the dynamic system cannot be detected. The collaborative signal processing improves the performance of decision-making process. That is, the CSFD scheme determines the faulty nodes each time with minimum error bound. The performance of CSFD scheme is better than the conventional scheme in making the decision fusion to identify the sensor faults in the WSN.

The fusion center is implemented as an ASIC for real-time WSN applications. The CSFD scheme architecture has complex computations such as exponent, multiplication, and division. This results with hardware complexity in the implementation and with more energy consumption. To overcome these drawbacks an ACSFD scheme is developed using the VLSI architecture [5]. ACSFD requires only one-third computations compared with CSFD. The ACSFD circuit consists of 9265 gates and needs a core size of 368 x 358 μm^2 operating at a clock frequency of 102 MHz and having a power consumption of 2.516 mW.

The rest of the paper is organized as follows. Section 2 describes the architecture of ACSFD scheme in a detailed way with necessary diagrams and section 3 gives an analysis of diverse sorting algorithms like quick sort, merge sort, bubble sort, counting sort, insertion sort, selection sort, and shell sort used in the ACSFD scheme for finding the faulty sensor node in the WSN. Finally conclusion is given in section 4.

II. ACSFD SCHEME

In the conventional method, to make decision fusion for WSN cannot be done suddenly because the fusion centers have to report their decisions at periodic interval. To avoid this CSDF scheme was developed [7], but the fusion centers for WSN are implemented in VLSI. The VLSI architecture of CSFD scheme is complex due to the enormous computations. To reduce the computations ACSFD scheme was proposed [5].

A wake-up detector for an acoustic surveillance sensor network was also implemented using VLSI [4]. The CSFD scheme for isolating the faulty nodes will eliminate the unreliable local decisions while distributed decision fusion [7]. Figure 2 shows the structure of a parallel fusion network model. The model has N sensor nodes used to observe the parameters. The observation sequences are represented by x^n_t, where t denotes the time index and n corresponds to the sensor node. Based on this value a binary decision is made and is denoted by u^n_t. Binary decision from the entire individual sensor node is sent to the fusion center. Two types of errors may affect the received local decisions at the fusion center, and are caused by either the faulty node which are randomly deployed or the channel transmission which is caused due to interference. The performance of fusion center will be degraded by the faulty nodes [5].

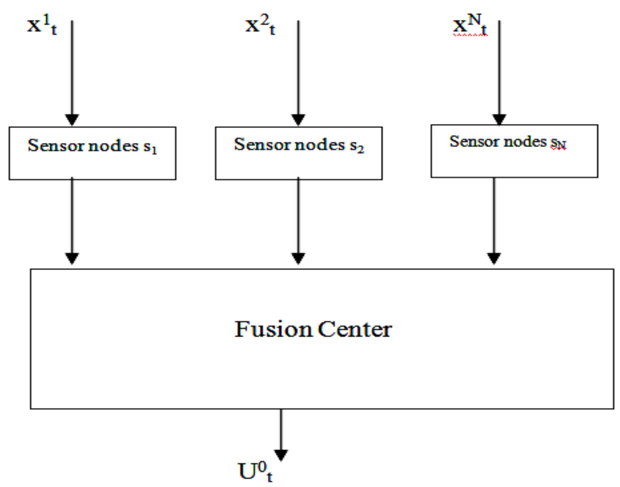

Figure 2 Parallel Fusion Network Model

The CSFD architecture for finding the faulty node in the WSN will be implemented using VLSI circuit. This implementation is very complex due to the following two reasons.

i. CSFD requires complex computations like exponent, multiplication and division.
ii. The bit width required during computation is very large.

To avoid these two difficulties ACSFD scheme [5] was developed. This scheme requires only less computation than CSFD scheme, which results in less hardware requirements and power consumption than CSFD scheme. The architecture of ACSFD scheme consists of the following units connected to a shared common bus and a top-level Finite State Machine (FSM) will coordinate the various operations between these units.

a. Logarithm unit
b. Sorting unit
c. Register file
d. Multiplier unit
e. Comparator unit
f. Adder/subtractor unit

From the above units the multiplier unit, logarithm unit and sorting unit has an important role in the ACSFD scheme. In this paper, importance has been given to the sorting unit. Based on the weights of the faulty sensor, sorting is needed to find the biggest values from the weights and identify their node index for further usages. In [5], insertion sorting scheme is used for finding the bigger values and their indexes through the entire sorting rule. The state diagram of the sorting scheme is shown in figure 3.

This has six states from 000 to 011 or S_0 to S_5. 'g' is the control signal, and will be set to '1' if the input is greater than the temporary value. The sorting algorithm places the faulty weights of the sensor nodes in a numerical order. This algorithm is used in different applications based upon its computational complexity, memory usage recursion, stability, and based on adaptability. Selection of efficient and reduced time complexity sorting algorithm will improve the performance of ACSFD scheme. For the analysis of finding the time complexity in diverse sorting algorithm, different number of faulty sensor nodes in the WSN is considered.

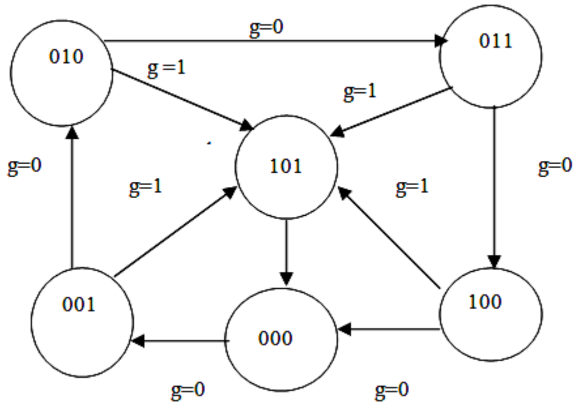

Figure 3 sorting module state diagrams

III. ANALYSIS OF SORTING

Analysis of diverse sorting algorithms has been carried on an Intel Core 2 Duo CPU system with 2.10 GHz on a 32-bit Windows 7 Ultimate Operating System. Analysis

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication

has been carried out for the following sorting algorithms quick sort, merge sort, bubble sort, counting sort, insertion sort, selection sort, and shell sort. Each sorting algorithms has different logic in sorting. Simulation has been performed for five different scenarios. For first scenario twenty weights of the sensor nodes with fault are considered. For second scenario forty weights of sensor nodes with fault, third scenario sixty weights of sensor nodes with fault, fourth scenario eighty weights of sensor nodes with fault, and fifth scenario hundred weights of sensor nodes with fault are considered. Table 1 show the time consumed in seconds for the various scenarios used with diverse sorting algorithms.

Table 1 Time consumed for diverse sorting algorithms under various scenarios

Type of sorting algorithm used	Number of sensor nodes in the WSN with faulty nodes				
	20	40	60	80	100
Quick sort	0.004375	0.002927	0.004309	0.003391	0.007358
Merge sort	0.006550	0.006351	0.009390	0.010554	0.010524
Bubble sort	0.190745	0.066123	0.063572	0.090835	0.068407
Counting sort	0.000317	0.001418	0.001344	0.001406	0.002271
Insertion sort	0.014858	0.001114	0.000528	0.000606	0.000711
Selection sort	0.006257	0.008256	0.006824	0.008447	0.006671
Shell sort	0.004703	0.000300	0.000525	0.000609	0.001595

Table 1 describes the time consumed in seconds for sorting the weights of the sensor nodes with faulty nodes using diverse sorting algorithms under five different scenarios, each having weights of 20, 40, 60, 80, and 100 sensor nodes. The proposed ASCFD scheme in [5] uses the insertion sort. From table 1, it is clear that shell sort requires minimum computational time for sorting compared with the other six sorting algorithms. Insertion sort algorithm is very simple and is mainly used with small number of values to be sorted. The operation of insertion sort is that, the numbers from the list are taken one by one and is inserted in the appropriate place in a new list. This is more time consuming than shell sort algorithm. Shell sort algorithm is a kind of insertion sort algorithm and is used to sort large number of values. In shell sort more than one value is placed in the new position at a time. This will speed up the sorting process and the shell sort algorithm can be used to increase the speed of computation of the ASCFD scheme which is implemented in VLSI to detect the faulty sensor node for the WSN.

IV. CONCLUSION

This paper presents the analysis of diverse sorting algorithms for the ACSFD scheme to detect the faulty sensor nodes in the WSN. ACSFD scheme is implemented to minimize the computational complexity while the embedding the architecture into a VLSI chip. Due to the complexity in implementing CSFD scheme into a VLSI chip for finding the faulty sensor nodes in the WSN, ACSFD scheme is used to detect the faulty sensor nodes in the WSN because ACSFD scheme has only one-third computations than the CSFD scheme. Multiplier, logarithm and sorting are the major computations needed in the ACSFD scheme. For sorting insertion sort was used conventionally in ACSFD scheme. In this paper, an analysis has been carried out with different sorting algorithms and observed that the shell sort requires less computational time than the other sorting algorithms. Therefore the performance of the fault detection method using ACSFD scheme for the WSN can be improved by using the shell sort algorithm in the sorting part of the ACSFD architecture.

REFERENCES

1. Akyildiz I. F., Su W., Sankarasubramaniam Y., and Cayirci E., "Wireless sensor networks: A survey", Journal on Computer Networks, Vol. 38, pp. 393-422, 2002.
2. Cheng Q., Varshney P. K., Mehrotra K. G., and Mohan C. K., "Bandwidth management in distributed sequential detection", IEEE Transactions on Information Theory, Vol. 51, No. 8, pp. 2954-2961, Aug. 2005.
3. Chih-Min Chao, and Yi-Wei Lee, "A Quorum-Based Energy-Saving MAC Protocol Design for Wireless Sensor Networks", IEEE Transactions on Vehicular Technology, Vol. 59, No. 2, pp. 813-822, Feb. 2010.
4. Goldberg D. H., Andreou A. G., Julian P., Pouliquen P. O., Riddle L., and Rosasco R., "A wake-up detector for an acoustic surveillance sensor network: Algorithm and VLSI implementation", Proceedings of the IPSN'04, pp. 134-141, 2004.
5. Pei-Yin Chen, Li-Yuan Chang, and Tsang-Yi Wang, "A Low-Cost VLSI Architecture for Fault-Tolerant Fusion Center in Wireless Sensor Networks", IEEE Transactions on Circuits and Systems, Vol. 57, No. 4, pp. 803-813, April 2010.
6. Vijaya Raja V., Rani Hemamalini R., and Jose Anand, "Performance Evaluation of Agent-based Upstream Congestion Control Approach in WSN" International Journal of Computer Technology and Applications (IJCTA), Vol. 2 (6), pp. 1939-1944, (ISSN: 2229-6093), Nov./Dec. 2011.
7. Wang T. Y., Chang L. Y., Duh D. R., Wu J. Y., "Fault-tolerant decision fusion via collaborative sensor fault detection in wireless sensor networks", IEEE Transactions on Wireless Communication, Vol. 7, No. 2, pp. 756-768, Feb. 2008.
8. Yuanzhu Peter Chen, Arthur L. Liestman, and Jiangchuan Liu, "A Hierarchical Energy-Efficient Framework for Data Aggregation in Wireless Sensor Networks", IEEE Transactions on Vehicular Technology, Vol. 55, No. 3, pp. 789-796, May 2006.

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication