COLLABORATIVE COMMUNICATION PROTOCOLS FOR DISTRIBUTED MICROSENSOR NETWORK SYSTEMS

Wootae Jeong[1], Shimon Y. Nof[1]

PRISM Center, School of Industrial Engineering,
Purdue University, West Lafayette, IN 47907, USA

Abstract: Current wireless microsensor network systems with low energy and cost have provided extended capabilities for manufacturing, environmental monitoring, transportation, biomedical care, and security and safety. Because of their miniaturized size, however, microsensors have serious energy constraints and limited computing power. Therefore, microsensor network systems require constraint specific communication protocols to deliver accurate, timely readings despite frequent communication failures. In order to achieve the desired accuracy of data, timeliness, and reliability, this research addresses the importance of fault tolerant and constraints adaptive protocols, and emerging trends in wireless microsensor network systems.

Keywords: Collaborative Control, Fault-Tolerance, Multisensor Integration, Microsensor Fusion, Sensor Networks, Protocols, Communication Protocols, Time-Out.

1. INTRODUCTION

Before the advent of micro-miniaturization technology, single sensor systems played an important role in a variety of practical applications because they were relatively easy to construct and analyze. However, single sensor systems offer an acceptable solution only when there is critical limitation of space to implement them. They are also relatively limited in their reliability. In order to overcome these critical disadvantages in most realistic applications, multisensor network systems have been designed with replicated microsensors to deliver accurate and robust information for various conditions. It is now expected that current microsensor communication architectures of Distributed Sensor Networks (DSN) should provide seamless connectivity and reliability under different constraints needed in a distributed system environment. In general, a DSN should provide main features as follows:

- *Fault-tolerance*: Multiple sensors increase reliability in case of sensor errors and network link failures. In case the microsensor node or link fail, a DSN enables enough redundancy with the data from different routes or nodes that the system may still produce acceptable quality information.
- *Accuracy Improvement*: Redundancy of information can reduce overall uncertainty and increase the accuracy with which events are perceived. Since nodes located close to each other are combining information about the same event, fused data improve the quality of the event information.
- *Timeliness*: DSN can provide the processing parallelism that may be needed to achieve as part of the integration process, either at actual speed that a single sensor provides, or even at faster operation speed.
- *Lower cost*: Although there is redundancy, a distributed microsensor system can obtain useful information at a lower cost compared with the equivalent information expected from a single sensor. The reason: It does not require additional cost of functions to obtain information at the same reliability and accuracy levels.

[1]Partially supported by Purdue Research Foundation, the PRISM Center and Indiana 21ST Century for Science & Technology

Therefore, a distributed network of MicroSensor Arrays (MSA) can yield more accurate and reliable results based on built-in redundancy. Recent developments of flexible and robust protocols with improved fault-tolerance will not only meet essential requirements in distributed systems, but will also provide advanced features needed in specific applications. In this paper, Section 2 describes structures of distributed microsensor networks and related research about network architecture, protocol and sensor integration algorithms. In Section 3, a Fault-Tolerant Time-out Protocol (FTTP [2]) is described including architecture and advantages of time-out control schemes for DSN. Section 4 explains new challenges for integration and future implementation for various applications.

2. DISTRIBUTED MICROSENSOR NETWORK

Since the importance of distributed microsensor systems was highlighted in applications of various surveillance systems with respect to monitoring and communications, studies about DSN have been developed over the last decade. An effective microsensor network system should contain seamless connection with various applications in terms of the architecture of distributed microsensor network, communication protocols, multi-sensor fusion, and integration, as illustrated in Fig. 1.

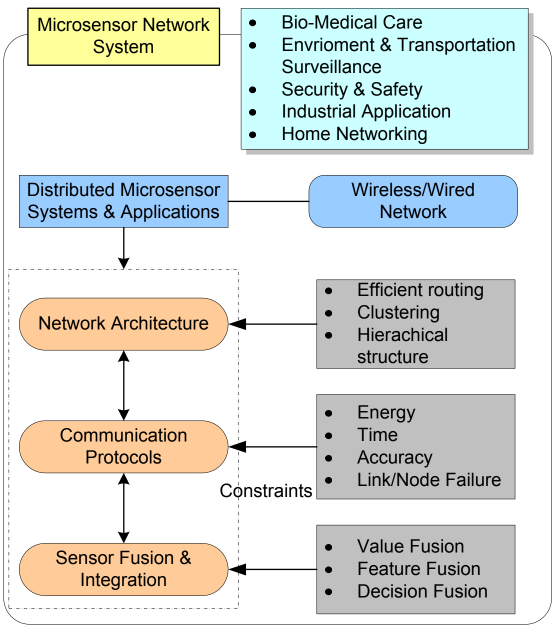

Fig. 1. Applications and development of distributed microsensor network systems

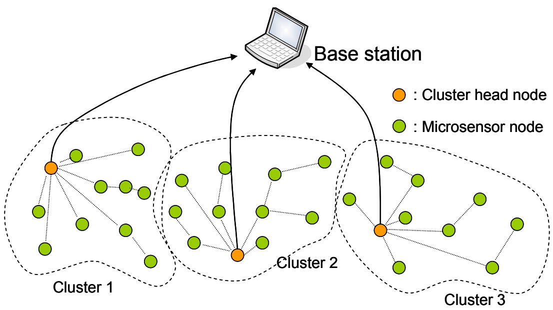

Fig. 2. General architecture of distributed microsensor communication network systems

2.1 Architecture

Various architectures have been proposed and developed to improve the performance of systems and fault tolerance functionality of complex networks depending on their applications. A general DSN structure was first discussed by Wesson et al. (1981) for a multi sensor system. Iyengar et al. (1994) improved and developed new architectures for the distributed sensor integration. As shown in Fig. 2, a DSN consists of a set of sensor nodes, a set of cluster-head nodes (CH), and communication network interconnecting the nodes (Iyengar et al., 1994, Jayasimha, 1996). In general, one sensor node communicates with more than one CH; a set of nodes communicating with a CH is defined as a cluster. Clustering architecture can increase system capacity and enables better resource allocation (Lin and Gerla, 1997; Ghiasi et al., 2002). Data are integrated at a cluster-head by receiving values from associated (not necessarily all) sensors of the cluster. In the clustering architecture, CHs can interact not only with other CHs, but also with higher level CHs, or with a fusion center; any sensor node can assume the role of the cluster CH, in case of CH failure in its cluster.

In recent years, with the advancement of wireless mobile communication technology, an ad hoc wireless sensor networks (AWSN) became important. The architecture of AWSN is fully flexible and dynamic. That is, a mobile ad hoc network represents a system of wireless nodes that can freely organize into temporary networks and allow nodes to communicate in areas with no existing infrastructure, thus interconnection between nodes can be dynamically changed and the network is set up only for a short time of communication (Ilyas, 2002). In applications where there is no given pattern of sensor deployment, such as battlefield surveillance or environmental monitoring, this approach can provide efficient sensor networking. In the dynamic network environment of AWSN, dynamic adaptation by self-organizing sensor networks is used to control the system (for instance, Lim, 2001).

In order to route information in an energy efficient way, directed diffusion routing protocol based on the localized computation model (Intanagonwiwat et al., 2000), has been discussed for robust communication. The consumer of data will initiate requests for data

[2] FTTP is a patent pending protocol of PRISM Center at Purdue University, USA.

with certain attributes. Nodes will then diffuse the requests towards producers via a sequence of local interactions. This process sets up gradients in the network which channel the delivery of data. Even though the network status is dynamic, the impact of dynamics can be localized.

A mobile-agent-based DSN (Qi and Snyder, 2000) utilizes a formal concept of agent to reduce network bandwidth requirement. Mobile agent is a floating processor migrating from node to node in the DSN and performing data processing autonomously. Each mobile agent carries a partially integrated data which will be fused at the final CH with other agents' information. However, to save time and energy, if certain requirements of a network are satisfied in the middle of its tour, the mobile agent returns to the base station without having to visit other nodes on the way. This logic reduces network load, overcoming network latency, and improves fault-tolerant performance.

2.2 Communication Protocols

Communication protocols for the distributed microsensor network provide systems with better network capability and performance, by creating efficient paths and accomplishing effective communication between the sensor nodes.

The point-to-point protocol (PTP) is the simplest communication protocol that transmits data only to one of its neighbors as illustrated in Fig. 3(a). But PTP is not appropriate for a DSN because there is no communication path in case of failure of nodes or links. In the flooding protocol (FP), the information sent out by the sender node is addressed to all of its neighbors as shown in Fig. 3(b). It disseminates data quickly in a network where bandwidth is not limited and links are not loss-prone. However, since a node always sends data to its neighbors, regardless of whether or not the neighbor has already received the data from another source, it leads to the implosion problem and wastes resources by sending duplicate copies of data to the same node.

Gossiping protocol (GP) (Hedetniemi et al., 1988) is an alternative to the classic flooding protocol, in which instead of indiscriminately sending information to all its neighboring nodes, each sensor node only forwards the data to one randomly selected Neighbour, as depicted in Fig. 3(c). While the GP distributes information more slowly than FP, it dissipates resources, such as energy, at a relatively lower rate. In addition, it is not as robust relative to link failures as BP, because a node can only rely on one other node to re-send the information for it, in case a link failure happens. In order to solve the problem of implosion and overlap, Heinzelman et al. (1999) proposed the Sensor Protocol for Information via Negotiation (SPIN). SPIN nodes negotiate with each other before transmitting data, which helps ensure that only useful transmission of information will be executed.

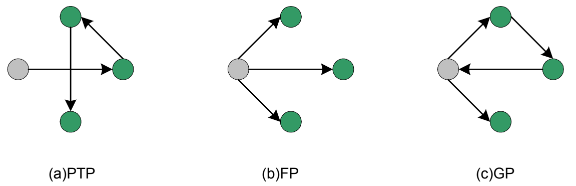

Fig. 3. Basic communication Protocols; (a) Point-to-Point Protocol (PTP), (b) Flooding Protocol (FP), (c) Gossiping Protocol (GP); Source: (Liu, 2001).

Under a relatively large sensor network, a clustering architecture with a local cluster-head (CH) is necessary. Heinzelman (2000) proposed the Low-Energy Adaptive Clustering Hierarchy (LEACH), which is a clustering-based protocol that utilizes randomized rotation of local cluster base stations to evenly distribute the energy load of sensors in DSN. The cluster-heads in the local cluster aggregate the information from each sensor node. In order to distribute the energy load among the cluster, LEACH elects a different cluster-head at different time intervals, which depends on the amount of energy left at the node. Thus, LEACH should be extended in the event driven network system.

A data-centric protocol was developed by Intanagonwiwat et al. (2000). They proposed the Directed Diffusion (DD) protocol, data dissemination paradigm for sensor networks. A DD has some novel features: data-centric dissemination, reinforcement-based adaptation to the empirically best path, and in-network data aggregation and caching. These features can enable highly energy-efficient and robust dissemination in dynamic sensor networks, while at the same time minimizing the per-node configuration that is characteristic of modern sensor networks.

Capabilities of communication protocols are compared in Table 1.

Table 1 Functional comparison of communication protocol capability for microsensor networks

	Simplicity	Fault Tolerance	Wireless Applicability	Networking Speed
PTP	●	□	□	●
FP	▲	●	□	□
GP	▲	▲	□	●
LEACH	□	▲	●	▲
DD	□	●	●	▲
FTTP	▲	●	●	●

●: Excellent, ▲: Good, □: No/Not available

2.3 Distributed Sensor Fusion/Integration

Multisensor integration or fusion is not only the process of combining inputs from sensors with information from other sensors, but also the logical procedure of inducing optimal output from the multi-inputs with one representative format (Luo and Kay, 1989). In the fusion of large-size distributed sensor network, the main advantage of Multi-Sensor Integration (MSI) is to obtain more fault-tolerant information. The fault tolerance is based on redundant sensory information that compensates faulty or erroneous readings of sensors. There are several types of multi-sensor fusion and integration methods, depending on the types of sensors and their deployment (Iyengar et al., 1995). This topic has received increasing interest in recent years because of the sensibility of networks built with many low-cost, micro- and nano-sensors.

The concept of abstract sensor (AS) was introduced by Marzullo (1990), who considered a physical value of sensor node as a continuous interval estimate that is a bounded and connected subset of the real measured value. Relative to AS, a concrete sensor (CS) is defined as a sensor that measures the physical variable of interest from the environment. An AS is derived from the point value received from the corresponding CS. From the interval of sensory readings, M function is defined to return the smallest interval that contains all the intersections of $(n - f)$ intervals, where n is number of microsensor nodes in a cluster and f is number of faulty sensors. However, it provides a single interval from all the sensors without fault detection.

The M function was extended by Jayasimha (1996) to provide all the $(n - i)$-intervals ($0 \leq i \leq f$), but in many applications, single output results are preferred. Another integration algorithm called F function was proposed to deliver stable outputs with respect to the slight change of input intervals (Schmid and Schossmaier, 2000).

A recent improvement of the Fault-Tolerance Sensor Integration Algorithm, FTSIA, by Liu and Nof, (2001, 2004) is that it not only detects the possibly faulty sensors and widely faulty sensors, but also generates a final data interval estimate from the correct sensors after removing readings of those faulty sensors. Steps of FTSIA algorithm for the example in Fig. 4 are given as follows:

Step 1: Find out the distinct 6-interval from abstract sensor I_1 to I_6

Step 2: Ignore I_1 because its higher bound is the smallest among the first six abstract sensors and find the distinct 6-interval from abstract sensors I_2 to I_7

Step 3: Combine distinct intervals and classify different levels of distinct intervals

Step 4: Compare the distinct intervals with all the abstract sensors and find out the possibly faulty sensors: if the result of step 2 contains the true value, I_1 is tamely faulty and also I_8 must be faulty.

Step 5: Ignore readings from possibly faulty and faulty sensors, find out the final output interval from the rest of the correct sensors.

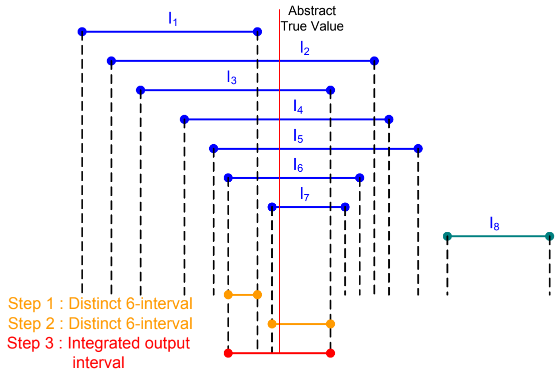

Fig. 4. Example of FTSIA execution with 8 sensory values, one tamely faulty sensor, and one widely faulty sensor, where I_i is interval reading from the i^{th} AS.

3. FAULT-TOLERANT TIMEOUT PROTOCOL

In order to overcome vulnerability of protocols in case of link failure, Fault-Tolerant Time-out Protocol (FTTP) has been recently developed by Liu and Nof (2004) at the PRISM Center to handle fault-tolerance effectively with a timeout control. FTTP is motivated by the combination of time-out and task coordination protocols (Esfarjani and Nof, 1998). It is also based on clustering architecture and knowledge base. It is robust per the failures of links by defining three stages of messages to communicate, in case of any types of communication link failures: Rerouting Task Message (RTM), Task Offer Message (TOM), and Backup Data Message (BDM). According to the time-out protocol, the base station will stop waiting for the information from the sensor node if a certain amount of time, T, has passed, where T is the predetermined time-out period and its value depends on the application. A base station will then announce the rerouting task message to its sibling nodes. The FTTP provides two alternatives according to the type of base station: FTTP with a blind base station and FTTP with a smart base station.

3.1 FTTP with a Blind Base Station

In this model, the base station (BS) that acts as the control node of the respective cluster will broadcast a data transmission request to all the sensor nodes in the cluster. In order to increase the system tolerance to link failure, each sensor node sends information not only to the base station, but also to one or more other sibling nodes that are defined as its backup nodes. When the link of the sensor node fails, the base station will broadcast the rerouting task message (RTM) to its all sibling nodes. Receiving RTM, its backup node generates a bid value. The base station then evaluates the priorities of the backup nodes based on their bid values, and sends the task offer message to the backup node with the highest priority (preferred bid value). Finally, the selected backup node with priority sends the required backup data message to the base station. Thus, this model is robust under the various failures of communication link and tolerates them well. It requires, however, certain additional communication and increase in communication traffic.

3.2 *FTTP with a Smart Base Station*

This protocol uses an active knowledge base to minimize the added communication traffic occurring with the blind base station protocol. Instead of sending RTM to all the sibling nodes of a failed link node, the base station transmits task offer message only to the node which has high priority by evaluating corresponding backup nodes. The priority is decided by the time when the backup node can finish the requested task. In comparison with the FTTP with a blind base station, the FTTP with a smart base station requires less communication, and has the advantage of being able to keep track of the updated transmission rate of each sensor node. However, this protocol also requires a base station with relatively more complex structure, because the knowledge base must actively keep track of all the information for each node, such as backup nodes assignments, variable transmission rates, and the number of tasks offered.

4. CHALLENGES OF FTTP

Protocol architectures for microsensor networks should consider a long system lifetime with low energy dissipation, high quality and timeliness of data, and fault tolerance. However, there is always a trade-off between quality of service and energy consumption. Fig. 5 illustrates wireless micro-node model of FTTP. Most of the energy dissipation occurs in transmitting/exchanging data among nodes, and optimization algorithms with time-out control and generation of routing paths in case of link failure also require certain amount of computation energy. Therefore, simplified algorithm and architecture in the extended FTTP have been designed to reduce complexity of network and energy dissipation of system through fast communication control with minimal wear of sensors.

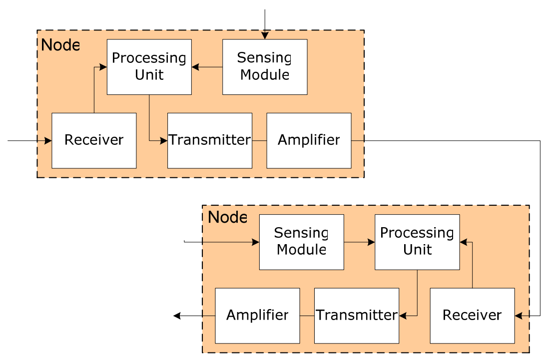

Fig. 5. Wireless micro-node model. Each node has sensing module, processing unit, and communication elements

4.1 *Fast communication*

Many applications require speedy network process both in network setup phase and in real data communication phase, to minimize overall communication time and delay. In addition, in order to obtain fast response in network communication, time synchronization (Lamport, 1987) and delay of packet transmission between nodes should be considered in the design of protocol architecture, and in analysis with network simulation. Therefore, time-out control and detailed time scheduling process in a packet transmission of the FTTP can improve the network communication speed between microsensor nodes. The extension and mathematical rationalization of the time-out control will be determined during the development of speedy network communication protocol.

4.2 *Low energy consumption*

In general, distributed microsensors have limited energy or battery for performing internal computation processes and transmitting information. This situation is especially true in a wireless network environment. Therefore, minimizing energy dissipation of a network for longer life time of the microsensor nodes is an essential criterion for microsensor network systems. Simplified control using time-out control and protocol architecture of FTTP can deliver better extension for future, emerging microsensor network applications with wireless communications.

4.3 *High accuracy of data*

To obtain high accuracy of output with minimal wear of sensors is another important issue in the design of microsensor network protocol. Intelligent or sophisticated network protocol can provide precise results and good fault-tolerance functionality, but it makes the entire network communication protocols more complicated. Therefore, there always exists a trade-off between high accuracy of information and low energy network system. In FTTP, well orchestrated protocol scheme between communication time management, and fault-tolerance provides a useful model to design microsensor network systems.

5. CONCLUSION

In the design of protocol architecture for wireless microsensor networks, recent needs for speedy and reliable communication motivated the development of FTTP. Time-out control and robust fault-tolerant function of the FTTP enable seamless connections among wireless microsensors in applications requiring high security and fast response, such as home/office security systems and transportation/ environment surveillance systems. The dynamic and flexible protocol design based on FTTP provides the expected high performance under various constraints in wireless network systems.

REFERENCES

Esfarjani, k. and S.Y. Nof (1998). A client-server model for integration and collaboration in production testing. *International Journal of Production Research* **36**(12), 3295-3321.

Ghiasi, S., A. Srivastava, X. Yang and M. Sarrafzadeh (2002). Optimal energy aware clustering in sensor networks. *Sensors* **2**, 258-269.

Hedetniemi, S.M., S.T. Hedetniemi and A.L. Liestman (1988). A survey of gossiping and broadcasting in communication networks. *Networks* **18**, 319-349.

Heinzelman, W.B. (2000). Application-Specific Protocol Architectures for Wireless Networks. Doctor of philosophy. Massachusetts Institute of Technology.

Heinzelman, W.R., J. Kulikand, H. Balakrishnan (1999). Adaptive protocols for information dissemination in wireless sensor networks. *Proceedings of the 1999 5th Annual ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom'99)*. ACM Press. Seattle, Washington, United States. pp. 174 -185.

Ilyas, Mohammad (2002). The Handbook of Adhoc Wireless Networks. CRC Press.

Intanagonwiwat, C., R. Govindanand, D. Estrin (2000). Directed diffusion: A scalable and robust communication paradigm for sensor networks. *In Proceedings of the Sixth Annual International Conference on Mobile Computing and Networking (MobiCOM'00)*. Boston, Massachusetts.

Iyengar, S.S., D.N. Jayasimha and D. Nadig (1994). A versatile architecture for the distributed sensor integration problem, *IEEE Transactions on Computers*, **43**(2), 175-185.

Iyengar, S.S., L. Prasad and H. Min (1995). Advances In Distributed Sensor Integration: Application and Theory. *Vol.7 of Environmental and Intelligent Manufacturing Systems*. Prentice Hall PTR.

Jayasimha, D.N. (1996). Fault tolerance in multi sensor networks. *IEEE Transactions on Reliability* **45**(2),308-315.

Lamport, L. (1987). Synchronizing time servers. Technical report. Digital Systems Research Center.

Lim, A. (2001). Distributed services for information dissemination in self-organizing sensor networks. *Journal of the Franklin Institute* **338**(6), 707-727.

Lin, C.R. and M. Gerla (1997). Adaptive clustering for mobile wireless networks, *IEEE Journal on Selected Areas in Communications* **5**(7), 1265-1275.

Liu, Y. (2001). Distributed Micro Flow Sensor Arrays and Networks: Design of Architecture and Fault-Tolerant Integration. M.S. Thesis, Purdue University.

Liu, Y. and S.Y. Nof (August 2004). Distributed micro flow-sensor arrays and networks: Design of architecture and communication protocols. *International Journal of Production Research* **42**(15), 3101-3115.

Luo, R.C. and M.G. Kay (1989). Multisensor integration and fusion in intelligent systems, *IEEE Transactions on Systems, Man and Cybernetics* **19**(5), 901-931.

Marzullo, K. (1990). Tolerating failures of continuous valued sensors. *ACM Transactions on Computer Systems* **8**(4), 284-304.

Qi, H. and W.E. Snyder (2000). Conditioning analysis of missing data estimation for large sensor arrays. *In Proceedings of IEEE Conference on Computer Vision and Pattern Recognition* **2,** 565-570.

Schmid, U. and K. Schossmaier (2000). How to reconcile fault-tolerant interval intersection with the lipschitz condition. *Distributed Computing* **14**(2), 101-111.

Wesson, R., F. Hayes-Roth, J.W. Brurge, C. Stasz and C.A. Sunshine (1981). Network structures for distributed situation assessment. *IEEE Transactions on Systems Man and Cybernetics* SMC-**11**(1), 5-23.