J Intell Manuf (2008) 19:335–345
DOI 10.1007/s10845-008-0086-4

Performance evaluation of wireless sensor network protocols for industrial applications

Wootae Jeong · Shimon Y. Nof

Received: 1 December 2006 / Accepted: 1 September 2007 / Published online: 25 January 2008

Abstract Recently, distributed wireless microsensor systems have provided more flexible leverage to emerging industrial applications. The tiny distributed wireless microsensor network systems, however, should be designed to overcome various constraints such as limited energy, bandwidth limit, and unexpected failure of communication under disturbances. In addition, their network topologies need to be managed with designated communication protocols. Thus, design of microsensor network protocols still needs to be application-specific. It should be also evaluated through designated tools at each level of networking characteristics. This research describes essential factors that affect the performance of sensor network systems in the design of wireless microsensor network protocols, and presents effective time-based network protocol and performance evaluation tool which are applicable for various protocols in industrial applications. The developed network evaluation tool, called TIE/MEMS, also includes functional comparison with recent protocols proposed for wireless microsensor networks, and provides design guidelines for multi-sensor network systems needed for emerging industrial applications.

Keywords Communication protocols · Sensor integration · Wireless communication · Microsensor networks · Network protocols · Teamwork Integration Evaluation (TIE) · Task administration

W. Jeong
Korea Railroad Research Institute,
360-1 Woulam-dong, Uiwang, Gyeonggi-do, Korea
e-mail: wjeong@krri.re.kr

S. Y. Nof (✉)
PRISM Center, School of Industrial Engineering, Purdue University, 315 N. Grant St, West Lafayette, IN 47907, USA
e-mail: nof@purdue.edu

Introduction

Design of distributed sensor network and integrating multiple sensors into manufacturing systems enable adaptive and flexible automation and enable better process adaptability and quality control. Remote sensing application, including weather prediction, agricultural application, and monitoring of compliance with nuclear disarmament treaties require large-scale sensor networking systems (Culler et al. 2004; Willig et al. 2005; Sundararajan et al. 2005). Arrayed microsensors and their networked systems are envisioned to provide a seamless link between the physical world and the global information infrastructure. They can produce widely accessible, reliable and accurate information for decisions about physical environments.

Although the field of wireless sensor networks has rapidly evolved, the targeted applications are environment surveillance and military applications. In order to be used in industrial applications, however, multi-sensor network systems should be designed with application-specific communication and task administration protocols and operating algorithms (Yu et al. 2005). Nevertheless, current application-specific wireless implementations have not been investigated with a generic system building approach for industrial applications. In addition, the protocols and algorithms should be selected through evaluation procedures to deliver accurate and robust information under the dynamic environment and frequently changing objectives. Therefore, in the design of wired or wireless communication network system, the architectures of Distributed Sensor Networks (DSN) should provide seamless connectivity and reliability under different constraints needed for the distributed system environment. In general, a DSN should provide essential features as follows:

- *Fault-tolerance*: Multiple, redundant sensors increase reliability when sensor errors and network link failures occur. In case a microsensor node or link fail, a DSN needs to enable sufficient redundancy so that data from different routes and nodes may still yield 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 an effective integration process, either at actual speed that a single sensor could provide, 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, localization, and accuracy levels.
- *Network topology*: A high number of nodes deployed throughout the sensory field should be maintained by carefully designed topology because any changes in sensor nodes and their deployments affect the overall performance of DSN. Therefore, a flexible and simple topology is usually preferred.
- *Energy consumption*: Since each wireless sensor node is working with a limited power source, the design of power-saving protocols and algorithms is a significant issue for providing longer lifetime of sensor network systems.
- *Scalability*: A coverage area of sensor network system depends on the transmission range of each node and density of deployed sensors. The density of deployed nodes has to be designed carefully to match the topology with the specific application requirements. The network density μ can be expressed according to (Cheng and Ansari 2003) as

$$\mu(R) = (N \cdot \pi \cdot R^2)/A \quad (1)$$

where N is the number of scattered sensor nodes in area A, and R is the wireless transmission range with circular propagation.

In order to provide the optimal solution to meet these design criteria in sensor networks, researchers have studied various protocols and algorithms. Because the design of sensor network system has typically been application-specific solutions, these studies have not satisfied all the general design considerations described above. Therefore, the main objective of the research described in this article is to provide a flexible sensor network protocol for industrial applications such as automated manufacturing systems by applying a time-based control. This article also addresses the importance of network performance analysis and evaluation by developing an efficient sensor network simulation tool.

Microsensor network architecture

A well designed distributed network with 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. They can produce widely accessible, reliable and accurate information about physical environments.

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 distributed sensor integration. As shown in Fig. 1, a DSN consists of a set of sensor nodes, a set of cluster-head nodes (CH), and communication network interconnecting the nodes (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 enable better resource allocation (Ghiasi et al. 2002; Lin and Gerla 1997). Data are integrated at a CH

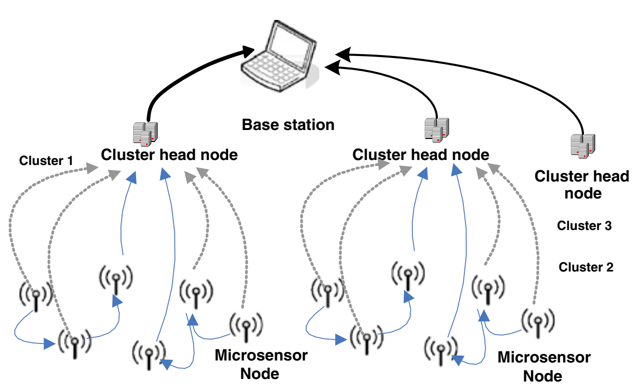


Fig. 1 General architecture of distributed microsensor communication network systems

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 centre; 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 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.

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 neighbours, as illustrated in Fig. 2a. 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 neighbours as shown in Fig. 2b. 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 neighbours, regardless of whether or not the neighbour 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.

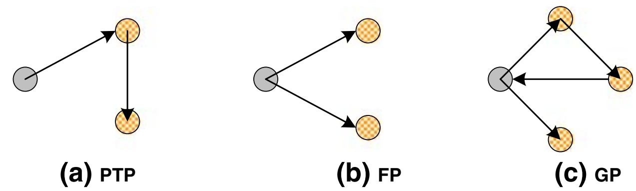


Fig. 2 Three basic communication Protocols; (**a**) Point-to-Point Protocol (PTP), (**b**) Flooding Protocol (FP), (**c**) Gossiping Protocol (GP); Source: (Jeong and Nof 2005)

Gossiping protocol (GP) (Hedetniemi et al. 1998) is an alternative to the classic flooding protocol, in which instead of indiscriminately sending information to all its neighbouring nodes, each sensor node only forwards the data to one randomly selected neighbour, as depicted in Fig. 2c. 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 a broadcasting protocol, 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.

Under a relatively large sensor network, a clustering architecture with a local cluster-head (CH) is necessary. 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 was developed by Heinzelman (2000). 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.

Energy minimizing routing protocols have also been developed to extend the lifetime of the sensing nodes in a wireless network. For example, a Minimum Transmission Energy (MTE) routing protocol (Ettus 1998) chooses intermediate nodes such that the sum of squared distances is minimized by assuming a square of distance power loss between two nodes. However, this protocol results in unbalanced death of nodes with respect to the entire network. 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.

The design of industrial open protocols for mostly wired communication known as field buses like DeviceNet and ControlNet have been also evolved to provide open data exchange and messaging framework (OPC HAD Specifications 2003). Further development for wireless has been investigated in asset monitoring and maintenance on an open communication protocol such as ZigBee (ZigBee Alliance 2004).

Distributed sensor integration

Multi-sensor 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 M integrating function 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. 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), 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, (Liu and Nof 2004; Nof et al. 2003) 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.

Network performance evaluation

In the microsensor communication network, it is impossible to analytically model the interactions between all the nodes because a large number of nodes are involved and complexity of reality makes theoretical analysis impossible. In addition, a simulation should be cost effective especially for large-scale microsensor network experiments. Thus, numerical simulation is necessary to validate the suggested sensor network protocol design.

A network simulator, *ns*, has been developed at the University of California, Berkeley, to simulate network protocols (Fall and Varadhan 1998). It has been written in C++ with a communication interface using an OTcl engine. The simulator has been extended by adding several features to support extensive simulation of wireless network protocol by Heinzelman (2000). Many other simulation tools are under development to test specifically designed network architectures.

Another approach to verify the performance of sensor network is to evaluate the measures by formulating a cost function theoretically under various constraints (Mhatre et al. 2005). In general, the cost can be a function of energy in networked sensors, or length of path to route the signals according to the topology of the specific network protocol. However, time can be another critical factor to evaluate performance of sensor network. This paper addresses the importance of time factor by proposing time-based network protocol for speedy communication.

Time-based network protocol

In general, the objective of a time-based protocol is to ensure that when any tasks keep the resource idle for too long, their exclusive service by the resource is disabled. That is, the time-based control protocol is intended to provide a rational collaboration rule among tasks and resources in the networked system (Liu and Nof 2004). Here, slow sensors will delay timely response and other sensors may need to consume extra energy. The patented FTTP[1] uses the basic concept of time-out scheme effectively in a microsensor communication control.

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

The terms that are frequently used in this network communication for the time-based protocol are described as follows.

- *Set-up time*. This is the time required to prepare both cluster head and its member nodes for network communication. The average setup time is defined as T_s.
- *Cluster head waiting time*. This is the time for the cluster head node to get all responses from its member nodes. The average cluster head waiting time without time-out is defined as T_w.
- *Communication time*. This is the time to transmit and receive signals among sensor nodes after waiting time T_w. Data integration in a cluster head is processed during this communication time. The sensor communication time is defined as T_c.
- *Time-out threshold* (T_o). This is the time limit to obtain sufficient responses for required accuracy of the data in a cluster head.

In a microsensor network system, a time-based protocol is executed as follows: after set-up time, the cluster head waits for the responses from its member nodes until obtaining sufficient responses. When the time that the cluster head waits for requiring responses reaches the allowance time, called the threshold (T_o), the cluster head stops waiting for other responses from the cluster members. Therefore, the overall time that the network is reserved and completes the given process is:

$$T_{t/o} = T_s + T_o \quad (2)$$

where T_o is a time threshold to obtain k responses, assuming k is a minimum number of sensor nodes to obtain sufficient accuracy of data at the cluster head node. The network communication time can be minimized by utilizing sensing nodes with the shortest response time order.

Finally, failed links can be rerouted and all signals are transferred to the cluster head during the communication time. The rerouted communication from the failed links is illustrated in Fig. 8.

Sensor network performance evaluation

A wireless microsensor

The wireless microsensor node consists of a sensing module, a processing element, and communication elements. The sensing module is an electrical part detecting physical variable from the environment. The processing unit (a tiny microprocessor) performs signal processing functions, i.e. integrating data and computation required in the processing of information. The communication elements consist of a receiver, a transmitter, and an amplifier if needed (see Fig. 3).

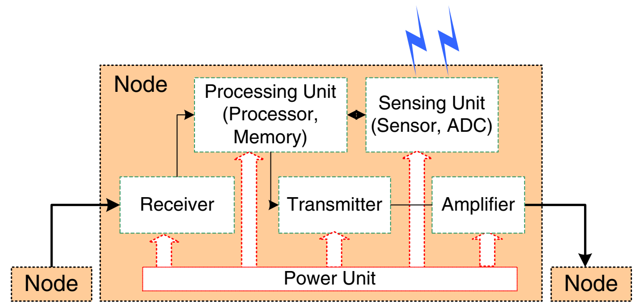


Fig. 3 Wireless micro-node model. Each node has sensing unit, processing unit, power unit, and communication modules

Basically, all individual sensor nodes are operated by a limited battery, but a base station node as a final data collecting center can be modelled with an unlimited energy source.

Communication energy model

In addition to the wireless micro-node model, it is important to adopt good communication model to use the energy efficiently in the microsensor network. In this research, the communicational and computational energy models among nodes are based on the models used in (Heinzelman 2000). In this model, the power attenuation is dependent on the distance, d, between the transmitter and receiver in the microsensor nodes. For a relatively short distance, the propagation energy loss is inversely proportional to d^2 while for the long distance, it is inversely proportional to d^4. Thus, each transmitting microsensor node should amplify the power to ensure the signal at the receiving node. Assuming that node A transmits a k-bit data packet a distance d to the node B, node A dissipates energy as follows:

$$\begin{aligned} Et &= Et_{elec} + Et_{amp} \\ &= \begin{cases} E_{elec} \times k + e_{amp} \times k \times d^2 \text{ (for a short distance)} \\ E_{elec} \times k + e_{amp} \times k \times d^4 \text{ (for a long distance)} \end{cases} \end{aligned} \quad (3)$$

For the receiving node B, it should expend energy as follows:

$$Er = Er_{elec} = E_{elec} \times k \quad (4)$$

where the electronics energy, E_{elec}, depends on factors such as the digital coding, modulation, and filtering of the signal before it is sent to the transmit amplifier. The parameter, e_{amp}, depends on the required receiver sensitivity and the receiver noise ratio.

The communication energy mode adopted here takes important role not only in generating clusters in the sensor network, but also in deciding the routing path under the failure of communication link, because priority of routing node is a function of response time from nodes in this time-based control protocol.

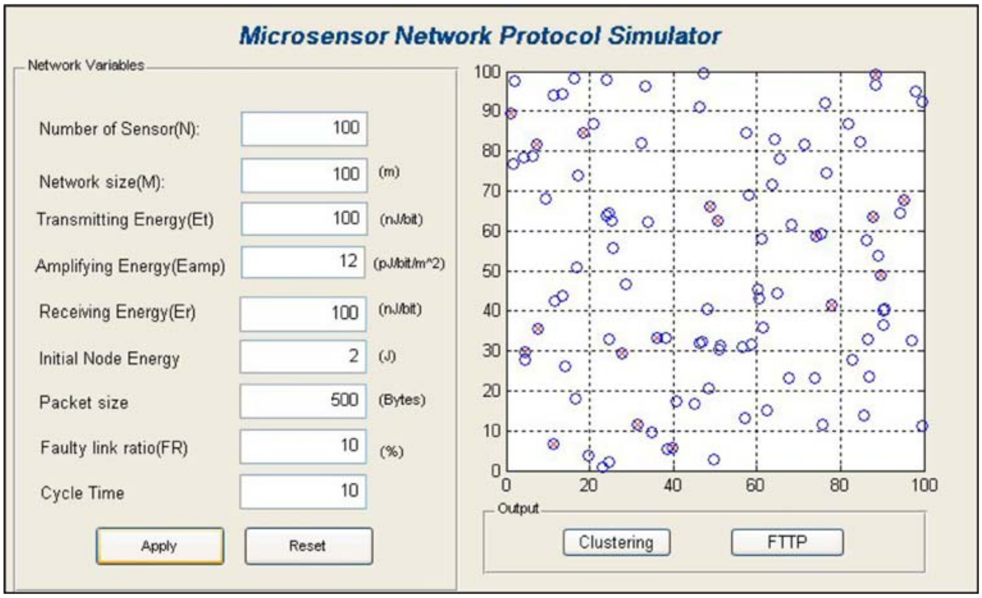


Fig. 4 Microsensor network simulator with a Matlab tool

Microsensor network simulator

The time-based network protocol was evaluated by Teamwork Integration Evaluator (TIE) using parallel programming (Jeong and Nof 2005; Liu and Nof 2001). TIE has been improved with a concrete time-based communication control and energy model (Jeong and Nof 2008). Figure 4 illustrates the enhanced simulator programmed by a Matlab tool.

The procedure of simulation consists of two phases, i.e., network setup phase and data transmission phase. The setup phase includes cluster formation, detection of sensor location, and cluster head node selection through minimal overhead information exchanges. The data transmission phase consists of a steady-state data communication and integration with a time-based control. The detailed procedure can be described as follows:

1. *Network setup*: A network environment variables and parameters must be specified before executing the network communication protocol. Constants used in the network should be checked as well. Constants and variables required in the time-based network simulation are summarized in Table 1. Note that mistakes in units of variables and inputs will result in unexpected results or errors in performance evaluation.
2. *Setup stage*: Microsensors exchange their minimal size of overhead data. The network is organized and executes main functions as follows:
   - From the given network size and sensors, number of clusters and CH nodes in each cluster are decided. Each cluster can have only one CH for managing cluster member nodes.
   - Failed communication links are detected and the faulty link information is updated to the base station.
   - Minimum number of sensors for time-based control is calculated and informed to the cluster head nodes.
   - Decide rerouting path for faulty link via a prior back-up node.
3. *Time-based control*: An initial control time ($T_{t/o}$) is decided by the number of sensor nodes required to obtain accuracy of data with existence of faulty nodes. Reducing the $T_{t/o}$ from the responses of sensor nodes will improve speed of a network system and minimize the energy consumption of the network.
4. *Steady-state stage*: Data are transmitted from each node to cluster-head node or base station. Instead of sending all the readings of sensors, the minimum number of sensory values required to obtain sufficient accuracy is transmitted according to the time-based control.
5. *Data integration*: Transmitted readings from each cluster member node to the cluster head node are integrated with a fault-tolerance sensor integration algorithm. The algorithm is described in (Jeong 2006). Integrated data in a cluster head are sent to the base station for final analysis.
6. *Repeat the cycle*: One cycle consisting of setup and steady state phase is repeated periodically.
7. *Analyze results*: Final Results at the base station are analyzed for the purpose of further utilization.

Evaluation of protocols

To evaluate the efficiency of basic network architectures and communication protocols discussed in this paper, experiments

Table 1 Summary of parameters in the microsensor network simulator

Network constant	Variables
Number of nodes (N)	Time-out value ($T_{t/o}$)
Network size (M)	Setup time (T_s)
Initial energy for each node (Joule)	Threshold time (T_o)
Number of possibly faulty sensors (f)	Communication time (T_c)
Transmitting energy (E_{elec})	
Amplifying energy (e_{amp})	

Table 2 Functional comparison of communication protocol capability for microsensor networks

	Simplicity	Fault tolerance	Mobile applicability	Networking speed
PTP	●	☐	☐	●
FP	●	●	☐	☐
GP	▲	▲	☐	●
LEACH	☐	▲	●	▲
MTE	☐	●	●	▲
FTTP[a]	▲	●	●	●

●: Excellent, ▲: Good, ☐: No/Not available [a] FTTP is a patent pending protocol of the PRISM Center at Purdue University, USA

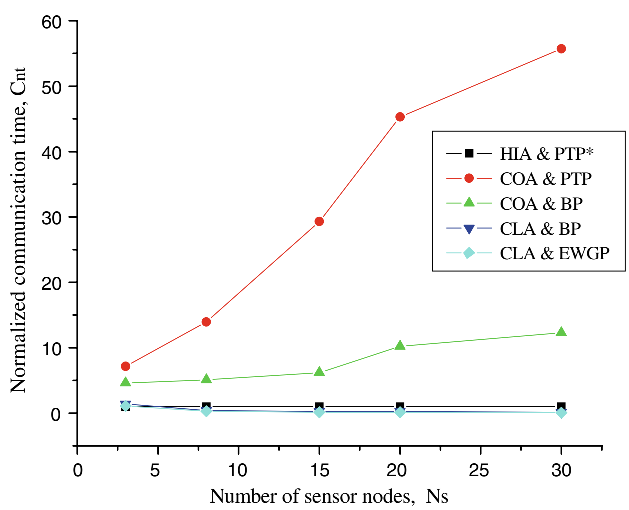


Fig. 5 Normalized communication time (*Cnt*) of five combinations of network architectures and communication protocols vs. number of sensor nodes (*Ns*), (*: Used as normalization reference)

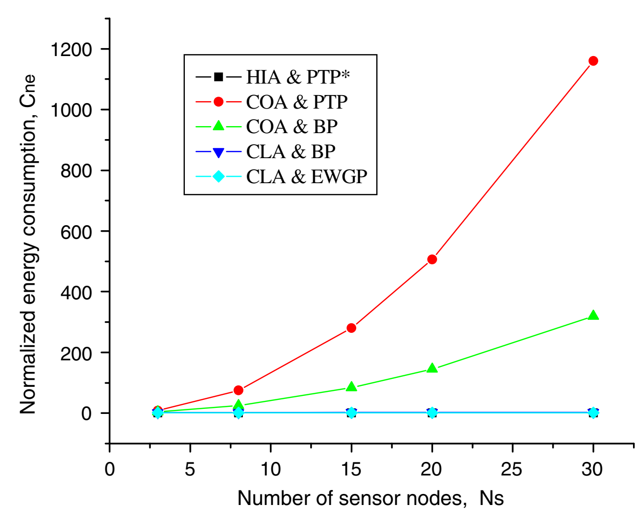
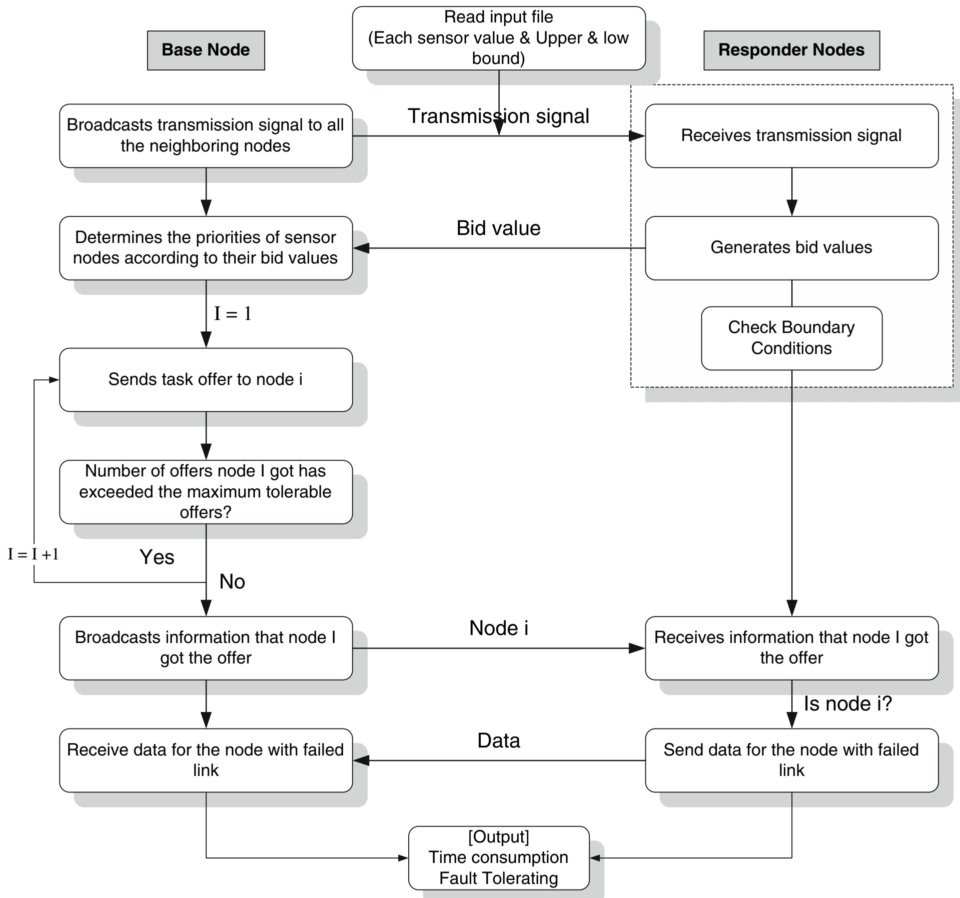


Fig. 6 Normalized energy consumption (*Cne*) of five combinations of network architectures and communication protocols vs. number of sensor nodes (*Ns*); Normalization by the ratio to the energy consumption of the leaf node in HIA and PTP (*: Used as normalization reference)

have been conducted through the TIE/MEMS parallel simulator. First of all, five combinations of network architectures and communication protocols were selected for comparison, and the recently developed WSN protocols are compared with FTTP according to their functional capability, as shown in Table 2.

The five selected combinations are: (1) Hierarchical Architecture (HIA) and PTP, (2) Committee Architecture (COA) and PTP, (3) COA and BP, (4) Cluster Architecture (CLA) and BP, and (5) CLA and Evenly Wide Gossiping Protocol (EWGP) with the number of backup nodes, N_b, equal to one. Thirty experiments were run for each combination. Output of the experiments is communication time (*Ct*) and communication energy (*Ce*) per communication round. The results of *Ct* vs. number or sensors (*Ns*) and the normalized communication time, *Cnt*, vs. *Ns* (normalization by the ratio to *Ct* of HIA and PTP) are shown in Fig. 5. Assuming energy transmitting cost of 600 mW per message transmission and energy receiving cost of 200 mW per message reception, overall energy consumption of each combination model is compared in Fig. 6. Results are expressed as the normalized energy consumption, *Cne*, vs. *Ns* normalization by the ratio to *Ce* of the leaf node of in HIA. Programming structure of the TIE/MEMS parallel simulator is illustrated as shown in Fig. 7.

Fig. 7 Flow diagram of TIE/MEMS parallel programming structure with a timeout control



The experiment results indicate that when *Ns* is small, there is relatively small difference in both communication time and energy cost between the five combinations of network architectures and protocols. In this case, COA and CLA are preferred because of their fault-tolerant characteristics. As *Ns* increases, however, communication time and energy cost of the COA network, especially, the COA with PTP, increase dramatically. Therefore, HIA and CLA are recommended for large sensor networks, and clustering architecture used in the FTTP can be a good choice in either case.

In addition to network architecture comparison, functional capabilities of recent communication protocols are compared as summarized in Table 2. From the summary, the FTTP has excellent networking speed and fault-tolerance capability. By rationalizing the structure of the FTTP, its relatively higher complexity can be improved. Especially, the functions for wireless applicability are extended with the timeout control procedure by proposing a new timeout-based information forwarding scheme (Jeong and Nof 2005).

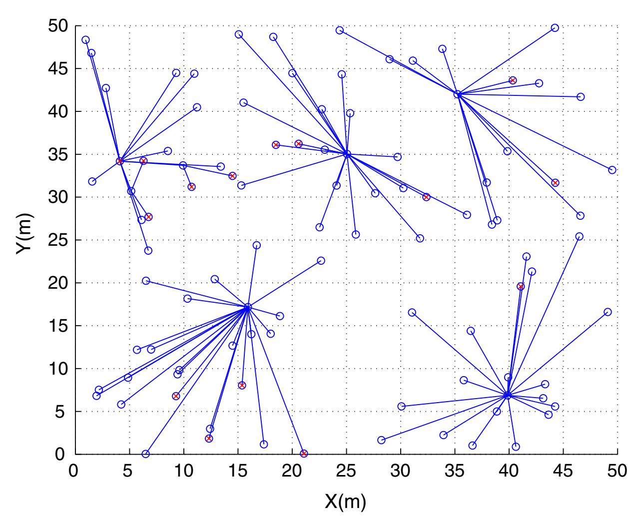


Fig. 8 Faulty sensor routed communication by a time-based control. Five node clusters, each with one cluster head, are shown. The lines between nodes indicate communication links

Time-based protocol evaluation and results

In order to evaluate the performance of the time-based control protocol, wireless microsensor network simulation using a Matlab tool has been conducted by adding, as described, several functionalities into the protocol. Other protocol architectures are also included for the purpose of comparison.

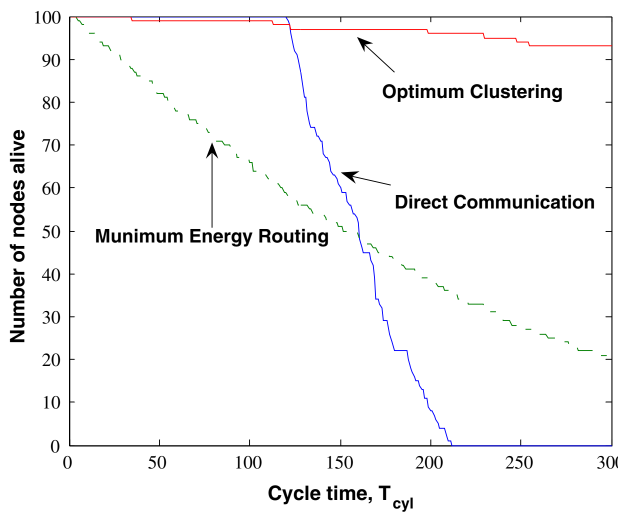


Fig. 9 Number of nodes alive (i.e., not faulty and having sufficient level of energy) after 300 cycle times (Cycle time in number of iterations)

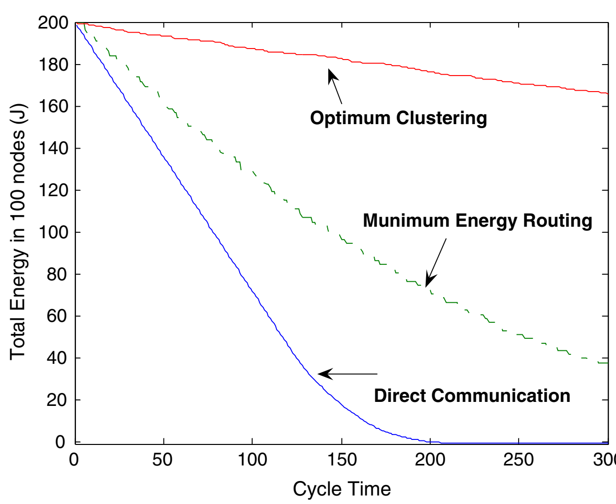


Fig. 10 Total remaining energy after each cycle time (Cycle time in number of iterations)

In this simulation, randomly distributed 100 microsensor nodes were used for this experiment as shown in Fig. 8. Overall network size was set to 50 $m \times 50\,m$ and the amplifying energy (e_{amp}) was set to $10 pJ/bit/m^2$; the transmitting energy (E_{elec}) was set to $50 nJ/bit$. The base station to collect final data from all the nodes was placed at (50 m, 150 m). The transmitting data among nodes was set to 500 byte long. Figure 8 shows that the network consists of five clusters, which are formed by a heuristic optimization algorithm, e.g., a simulated annealing algorithm (Kirkpatrick et al. 1983). Each cluster has one cluster head node, and the cluster heads are collecting data from their member nodes, finally transmitting data to the base station. The optimal number of clusters has been calculated following (Heinzelman 2000).

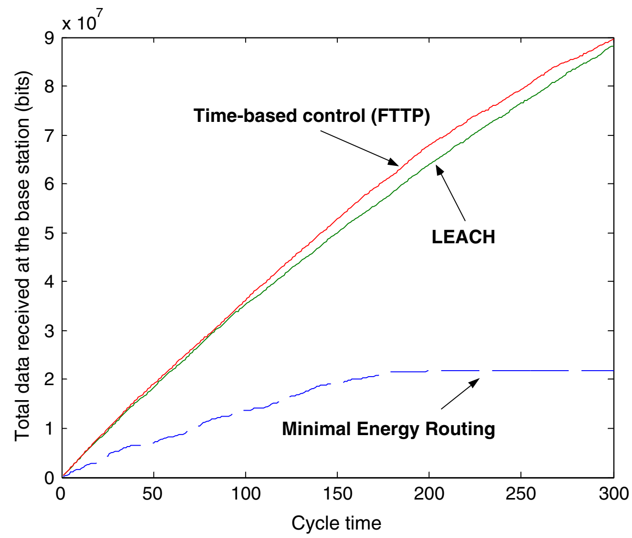


Fig. 11 Time-based control algorithm vs. other algorithms when 15% communication link failure is assumed (Cycle time in number of iterations)

In the simulation, the FTTP, i.e., a clustering protocol, and two non-clustering protocols are compared as shown in Figs. 9 and 10. The FTTP with clustering algorithm minimizes the total energy consumption in comparison with the non-clustering algorithms because clusters are reconstructed every cycle time by evenly distributing the cluster-head-nodes throughout the network. Relatively, nodes close to the base station, however, tend to drain out their energy earlier in the minimum energy routing protocol as shown in Fig. 10.

Based on the given network parameters and algorithm developed, total quantity of data received at the base station is analyzed as shown in Fig. 11. The timeout-based communication algorithm used in this analysis contains the energy model and systematic design steps described in the previous section; detailed code has been omitted. Assuming 15 percent of distributed nodes are possibly faulty in their communication links, two other sensor network protocols, i.e., LEACH and Minimal Energy Routing protocol, fail to deliver all data from sensor nodes, but the timeout-based control protocol (FTTP) is relatively robust in collecting as much data as needed, under existing failed links.

Applicable industrial solutions

Initially, distributed microsensor networks have been mostly applied for military applications. A recent trend of sensor networks, however, is to apply the technology for various industrial solutions. Focus of industrial applications involves system monitoring, system diagnosis and instrumentation, and middleware design to support specific applications. For the instrumentation applications, all the system components require real-time performance and fault-tolerance capability as shown in the timeout control scheme. Monitoring and diagnosing applications include reporting the state of system/facility/equipment, and take rational reactions if abnormal states are detected. Especially, networked sensor applications used in production facilities need to consider geometrical and dynamical characteristics of the system at the design step of network architecture. Those facility sensor networks (FSN) have to consider optimal deployment of wireless infrastructure, interoperability between legacy networks and sensor network, and extendibility to different application needs (Jeong and Nof 2007). Therefore, network protocol selection through performance evaluation addressed in this research becomes significant for industrial automation and production applications. It is also essential to use fault-tolerant network protocols to aggregate very weak signals without losing any critical signals. Specifically designed sensor network solutions can also be applicable for an intelligent transportation system, monitoring material flow, and home/office network systems.

Conclusion

For the design of microsensor network protocols, detailed specifications of application should be considered not only because general architecture of a sensor network cannot meet all requirements of each application, but also because implemented microsensor nodes have different characteristics. Through the proposed microsensor network protocol evaluation tool, TIE/MEMS, performance of various network communication models and architectures can be effectively evaluated by using the node response time, which provides timely and reliable communication. It also helps system designers to evaluate fault-tolerance capability of the networked systems. In particular, it can measure robustness over errors or failures due to both communication links and nodes. Based on this evaluation, it is possible to design a protocol that also enables seamless connections among wireless microsensors in applications requiring high security and fast response, such as home/office security network systems and transportation/environment surveillance solutions. Beyond the scope of this research, the authors are recently applying sensor networks with the timeout-based control to a monitoring/diagnosing application in automated manufacturing facilities. The timeout-based protocol evaluation tool has guided in selecting optimal network protocols to minimize inter-communication time and energy consumption. As analyzed and illustrated in previous sections, the dynamic and flexible protocol design with time-based control can provide expected levels of high performance under various constraints in wireless network systems.

Acknowledgements This work was partially supported by the PRISM Center; Indiana 21st Century Fund for Science & Technology; and Purdue Research Foundation.

References

Cheng, G., & Ansari, N. (2003). A theoretical framework for selecting the cost function for source routing. *IEEE International Conference on Communications, 1*, 631–635.

Culler, D., Estrin, D., & Srivastava, M. (2004). Overview of sensor networks, *IEEE Computer, 37*(8), 41–49.

Ettus, M. (1998). System capacity, latency, and power consumption in multihop-routed SS-CDMA wireless networks. In *Radio and Wireless Conference (RAWCON '98)*, pp. 55–58.

Fall, K., & Varadhan, K. (1998). The network simulator ns-2. http://www.isi.edu/nsnam/ns.

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

Hedetniemi, S. M., Hedetniemi, S. T., & Liestman, A. L. (1998). A survey of gossiping and broadcasting in communication networks. *Networks, 18*, 319–349.

Heinzelman, W. B. (2000). Application-specific protocol architectures for wireless networks. Dissertation, Massachusetts Institute of Technology.

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

Ilyas, M. (2002). *The handbook of adhoc wireless networks*. CRC Press.

Intanagonwiwat, C., Govindanand, R., & Estrin, D. (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., Jayasimha, D. N., & Nadig, D. (1994). A versatile architecture for the distributed sensor integration problem, *IEEE Transactions on Computers, 43*(2), 175–185.

Iyengar, S. S., Prasad, L., & Min H. (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.

Jeong, W. (2006). Fault-tolerant timeout communication protocols for distributed micro-sensor network systems, Dissertation, Purdue University.

Jeong, W., & Nof, S. Y. (2005). Design of protocols for wireless MEMS sensor network systems in industrial applications. In *18th International Conference on Production Research*, Italy, July.

Jeong, W., & Nof, S. Y. (2007). Optimal deployment of collaborative wireless microsensors in manufacturing facilities. In *19th International Conference on Production Research*, Chile, July.

Jeong, W., & Nof, S. Y. (2008). A collaborative sensor network middleware for automated production systems. *Special issue on collaborative e-Work networks in industrial engineering in International Journal of Computers and Industrial Engineering*, in print.

Kirkpatrick, S., Gerlatt, C. D. Jr., & Vecchi, M. P. (1983). Optimization by simulated annealing. *Science, 220*, 671–680.

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., & Gerla, M. (1997). Adaptive clustering for mobile wireless networks, *IEEE Journal on Selected Areas in Communications, 5*(7), 1265–1275.

Liu, Y., & Nof, S. Y. (2001). *TIE/MEMS: Modeling and analysis of micro sensor networks, user manual, Research Memorandum*, Industrial Engineering, Purdue University.

Liu, Y., & Nof, S. Y. (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., & Kay, M. G. (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.

Mhatre, V. P., Rosenberg, C., Kofman, D., Mazumdar, R., & Shroff, N. (2005). A minimum cost heterogeneous sensor network with a lifetime constraint. *IEEE Transaction on Mobile Computing, 4*(1), 4–15.

Nof, S., Liu, Y., & Jeong, W. (2003). Fault-tolerant time-out communication protocol and sensor apparatus for using same, Patent pending, PurdueUniversity.

OPC HAD Specifications (2003). Version 1.20.1.00 edition, http://www.opcfoundation.org.

Qi, H., & Snyder, W. E. (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., & Schossmaier, K. (2000). How to reconcile fault-tolerant interval intersection with the lipschitz condition. *Distributed Computing, 14*(2), 101–111.

Sundararajan, V., Redfern, A., Schneider, M., Wright, P., & Evans, J. (2005). Wireless sensor networks for machinery monitoring. *ASME International Mechanical Engineering Congress and Exposition, IMECE'05*, Nov., Orlando, United States.

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

Willig, A., Matheus, K., & Wolisz, A. (2005). Wireless technology in industrial networks. *Proceedings of the IEEE, 9*(6)1130–1151.

Yu, Y., Hong, B., & Prasanna, V. K. (2005). Communication Models for Algorithm Design in Wireless Sensor Networks. *Proceedings of the 19th IEEE International Parallel and Distributed Processing Symposium*.

ZigBee Alliance (2004). Network Specification, Version 1.0.