

# Supplemental Material for the Paper “Deploying Wireless Sensor Networks with Fault-Tolerance for Structural Health Monitoring”

Md Zakirul Alam Bhuiyan, *Member, IEEE*, Guojun Wang, *Member, IEEE*, Jiannong Cao, *Senior Member, IEEE*, and Jie Wu *Fellow, IEEE*



## APPENDIX A RELATED WORK ON DESIGNING WSN-BASED SHM SYSTEMS

Designing WSN systems for SHM is an interesting interdisciplinary topic, which is rarely combined by engineering and computer science (CS) communities deeply for their aspects. To date, most of the real SHM systems utilize wired sensor network systems to collect information about structures for SHM. While WSNs are gradually receiving attention from the engineering domains as an attractive tool, due to the WSN on-board processing, easy deployment, and relatively low capital and maintenance costs, SHM is becoming a potential application from the CS domain. During the past few years, both communities have started developing the analytical system to detect and quantify the structural health status under wireless sensor technologies [1], [2], [3], [4], [5], [6], [7], [8], [9], [10], [11], [12], [13], [14], [15], [16].

However, these approaches are generally proposed to support traditional global SHM, without much consideration to the WSN requirements and constraints. A state-of-the-art monitoring system has been deployed at the Golden Gate bridge (GGB) [10], which represents one of the first large-scale deployments of WSNs for SHM purposes. This system has a large latency, arising from the fact that the underlying SHM method (physical aspects, e.g., damage occurrence) was designed separately from the WSN (cyber aspects, e.g., communication).

Distributed frameworks [7], [9], [8] are presented for damage detection, where the final decision is made at the BS. A cyber-physical system (CPS) approach has recently been suggested in [11], considering both the constraints of the underlying WSNs (the *cyber* components) and the SHM requirements (the *physical* components). In this approach, a distributed damage detection algorithm is suggested under a hierarchical WSN. Although it offers a trade-off between the computation and communication capacities, it does not reveal the details, such as connectivity or coverage, in the hierarchical architecture.

A hierarchical cluster-based SHM (C-SHM for short) considers a fundamental problem in SHM:

mode shape analysis in clusters [9]. In each cluster, the vibration characteristics are identified at high vibration frequencies and then are assembled together. However, a heuristic-based clustering approach is used, which may not fit a real SHM. Although the clustering in the C-SHM needs to meet extra requirements of modal analysis, the quality of SHM may be affected. This is because the modal analysis can be different at the same cluster at different times. Moreover, the system performance through the hierarchical architecture of C-SHM in terms of connectivity, communication, lifetime, etc., is not discussed.

Although the approaches above are considered to be effective for monitoring, they suffer from at least two major limitations.

- They require high detection latency during damage detection, and *uninteresting* data that are collected from a large number of sensors without a reasonable time frame. For example, GGB requires 9 hours to collect a single round of data from 64 sensors. Such systems are inadequate for identifying health status (timely or online) during an extreme event. Thus, the resulting systems may inherently suffer from high energy consumption and sensor faults/failures.
- Allowing for the limitation of multi-hop transmission in WSNs, a huge amount of *interesting* data transmitted towards the BS may be lost. Thus, the reliability and quality of SHM is low.

FTSHM attempts to mitigate the limitations of the existing algorithms and approaches to some extent.

## APPENDIX B LINK-QUALITY MODEL

Besides the power required for communication between adjacent sensors, a communication metric of importance is the quality of a wireless link. Packet-loss is typically dependent on  $d_{uv}$ , with losses being higher for sensors with larger separation or communication distance. In FTSHM, since each sensor is enabled to transmit a local mode shape rather than a complete set, we must ensure that each data packet is successfully transmitted from a sensor to its CH, and the CH to the BS. A packet-loss means the mode shape

(which may contain *interesting data* about damage) is lost. Thus, every single packet-loss enforces a packet retransmission.

Since the sensor placement for an SHM approach is characterized by finding the optimal locations so as to fulfill the engineering requirements, the placement affects not only the routing protocols employed for data collection, but also the reliable collection of the transmitted data at the BS. The situation becomes more serious if all the recorded vibration signals (e.g., in case of damage) are needed to send. We estimate the link quality between any two nodes  $u$  and  $v$ , denoted by  $c_T(\{u, v\})$ , using an existing link model [17]. Such a model is suitable for our needs in deploying a WSN system for SHM, as it allows for the estimation of the predictive energy cost and the communication cost of optimal locations. The estimation of  $c_T(\{u, v\})$  helps to find high communication-efficient locations for sensor placement. This is very similar to the sensor placement on structures.

The total communication cost of two sensors  $u$  and  $v$ , placed at locations  $l_u$  and  $l_v$ , can be given as:

$$c_T(\{u, v\}) = \rho \int_{\alpha} \frac{1}{\alpha_{u,v}} \zeta d\alpha_{u,v} \quad (\text{B.1})$$

In (B.1),  $\alpha_{u,v}$  is the probability for a successful transmission between the locations of any two sensors  $u$  and  $v$ , and  $(\alpha_{u,v})^{-1}$  is the expected number of retransmissions, since the success packet transmission probability between any two locations depends on the network density  $\zeta$  instead of a fixed value for  $\alpha_{u,v}$ .

Using the formula in (B.1), we obtain the link quality between any two sensor locations  $u$  and  $v$ . When analyzing the link quality for sensor locations  $u$  and  $v$ , we set  $c_T(\{u, v\})'$  as an acceptable link quality. If an obtained link quality between any two sensors  $u$  and  $v$  is higher than  $c_T(\{u, v\})'$ , the link is considered a high quality (or reliable) link. If there are inconsistencies in the routing paths during the network function for SHM, FTSHM uses a data path validation tool [18], by which the routing layer can dynamically repair the topology with a low overhead, and forwards the packet normally.

## APPENDIX C PEAK-PICKING FREQUENCY METHOD

The Peak-Picking (PP) method is the simplest known technique for estimating the modal properties of a structure from structural system output data. This method, like many other output-only techniques, assumes that the immeasurable excitation input can be characterized as zero-mean Gaussian white noise. In the engineering applications, this type of excitation is generally achieved by using ambient vibration, forced vibration, or loading conditions. Assuming that a structure is excited with a white Gaussian noise, the FFT of a single sensor measured response corresponds

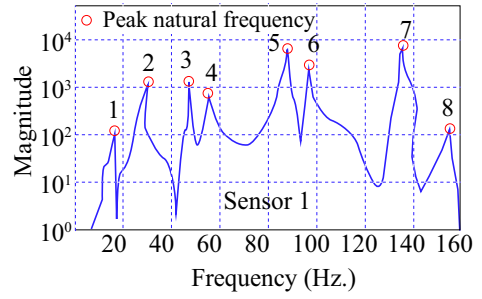


Fig. C.1. Picking peak frequency.

to the frequency response of the structure for that sensor. Since response's peaks are located in correspondence to the natural frequencies, as shown in Fig. C.1, a local analysis of calculated peak frequencies can lead to a fairly accurate estimate of natural frequencies, themselves. Significant changes in the natural frequencies may indicate the occurrence of structural damage. Each sensor is enabled to scan the frequency response set  $F_i$  and pick peak frequencies.

## APPENDIX D PROOF-OF-CONCEPT EXPERIMENTS

### D.0.1 Deployment Objectives

As a proof-of-concept experiment, we have conducted a field experiment to compute real mode shapes of the *Lee Shao Kee* (LSK) building located on our university campus (see Fig. 7). The objective of this deployment is to validate the feasibility of placement performance, fault tolerance, and decentralized data processing in SHM.

### D.0.2 System Deployment

The experiments are conducted three times on three different days. This is because the structural ambient excitation varies day to day, and the results could also be varied. On the 2nd and 3rd days, we conduct experiments under sensor faults. Each day, the network is monitored for 8 hours. The structure is instrumented by deploying motes, called *SHM mote*, integrated by off-the-self Intel Imote2 sensors [19]. According to the proposed algorithms, on the first day of deployment, we first select 22 locations for 22 SHM motes. They seem to capture the overall vibration frequency and mode shape without any sensor fault. There are at least 220 locations ( $=M$ ) on the structure computed by the FEM model of the structure [10], [20]. On the 2nd day, we first place 16 SHM motes ( $=N$ ) according to the EFI method. We find location coordinates of 6 RPs on the structure, and find appropriate locations out of the remaining locations (204 locations) for the 6 backup sensors ( $=R$ ).

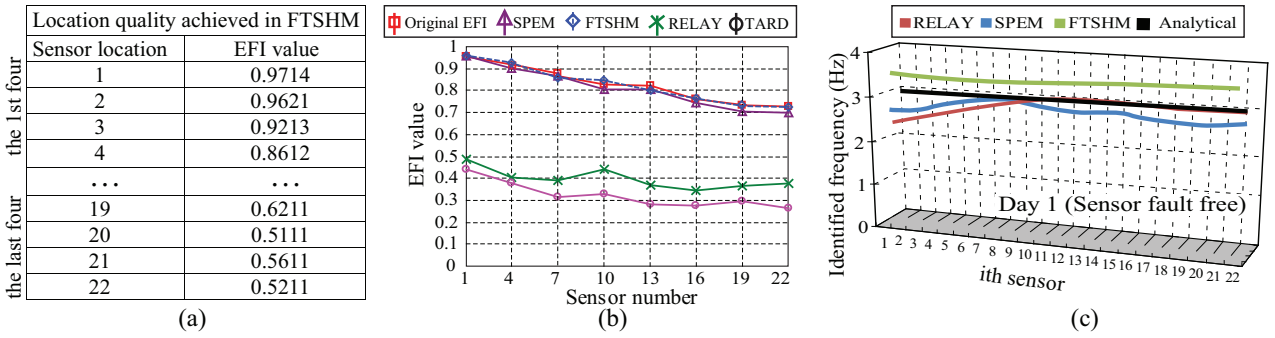


Fig. D.1. Experimental performance: (a) the location quality for the first four and the last four sensor locations; (b) location quality compared to other approaches; (c) identified mode shape under the sensor fault-free condition.

### D.0.3 Platforms

TinyOS 2.1 [21] runs on the Imote2. Each mote’s main board combines a low power PXA271 XScale processor with an 802.15.4 radio and an antenna using 2.4 GHz. It also offers 256 KB of integrated SRAM and 32 MB of external SDRAM, although our approach does not require such a large space. In FTSHM, the final mode shape results remain in the memory after computation that needs about 15 to 22 bytes. When a mote gets an acknowledgment of data packet reception, this data is also removed from the memory. On average, 3% – 5% of memory space has been seen to be used during each round of monitoring. The PXA271 is a 32-bit processor and is capable of computing mode shape independently. In our design, 32-bit is processed at a time so that 32-bit data is aligned; otherwise, we get weird results. Each packet is of a 14 byte header, plus a 28 byte payload. However, we make an option to increase the payload up to 116 bytes. To obtain simplified decisions and estimates of mode shapes for clusters, high-precision time synchronization is required to avoid phase differences in the sensors’ collected data. That is to say, the synchronization error should be less than 5 ms between the motes. We use a modified FTSP (flooding time synchronization protocol) [22] to fulfill the needs of time synchronization in a hierarchy (in the CHs and then sensors of the network).

An extra Imote2 as a BS is located around 80m away from the LSK building. There is a PC as a command center for the BS mote and data visualization that sets parameters for the WSN. The BS mote receives data packets from CH motes through wireless transmission, and relays the data to the PC over a USB cable. This implementation enables the motes to be organized into clusters within a single-hop to multi-hop, and transmits data to CHs. Two neighboring clusters are overlapped up to 40%. Each mote runs a program (implemented in nesC language) to fully process data acquired from on-board accelerometers. Digital acceleration data, acquired within frames of 2048 ( $\rightarrow x$ ) points, is then collected and processed

at each mote independently through the *Damage-Indicator* algorithm.

**Sensor Fault Injections.** We inject three kinds of faults. (i) 2 sensors, placed on the 5th floor and 7th floor, are given limited power, so that they can run up to 4 hours and then fail. (ii) At the around the 2nd hour, we remove one sensor from the 11th floor, and we think that a sensor may fail or be lost due to many reasons. (iii) Sensor debonding fault by loosening the attachment between a sensor and the building on the 13th floor, thus, it is supposed to provide “abnormal results.”

### D.0.4 Experimental Results

We first analyze the sensor placement performance achieved by our experiments. Fig. D.1(a) shows the location quality of the 1st four sensors and the last four sensors (mainly the backup sensors). We can see that the location quality achieved for the backup sensors is still high ( $> 0.5$ ).

In Fig. D.1(b), compared to other approaches, we found that FTSHM can fully satisfy the civil engineering requirements; even in several cases, it still outperforms SPEM (e.g., 4-sensor, 10-sensor, 16-sensor, and 19-sensor cases) in finding locations.

Fig. D.1(c) illustrates the results of the identified mode shape of the LSK building under the sensor fault free condition and undamaged structural condition, which are obtained on the first day of deployment. We can observe that, although there are no physical sensor faults, the mode shape identification is still affected to some extent in SPEM and RELAY. Particularly, SPEM has deviated from the analytical performance by more than 6%. This is mainly caused by communication errors. A large number of retransmissions is needed for the packet-loss. It is often found that a number of packets are not received at all at the BS.

Next, we analyze the performance of SHM under the sensor fault conditions in the WSN. We determine the mode shapes on the 2nd day. Fig. D.2(a) and Fig. D.2(b) demonstrate the mode shapes in all of the

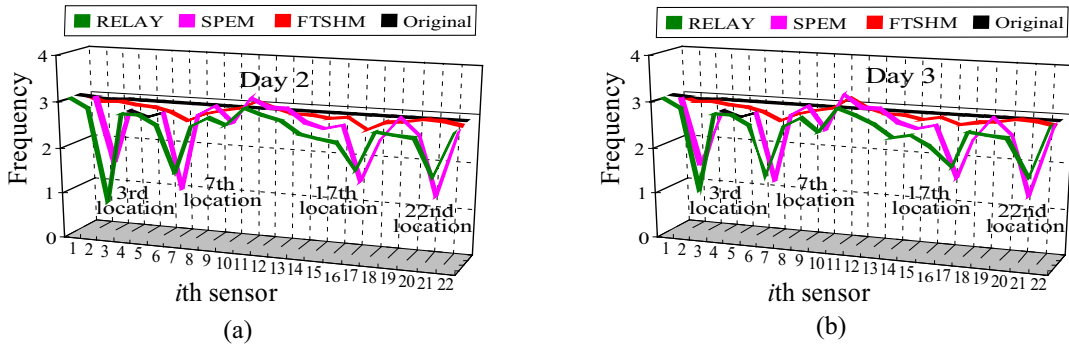


Fig. D.2. Impact of sensor faults in SHM performance: (a) identified affected mode shape under sensor fault on Day 2; (b) identified affected mode shape under sensor fault on Day 3.

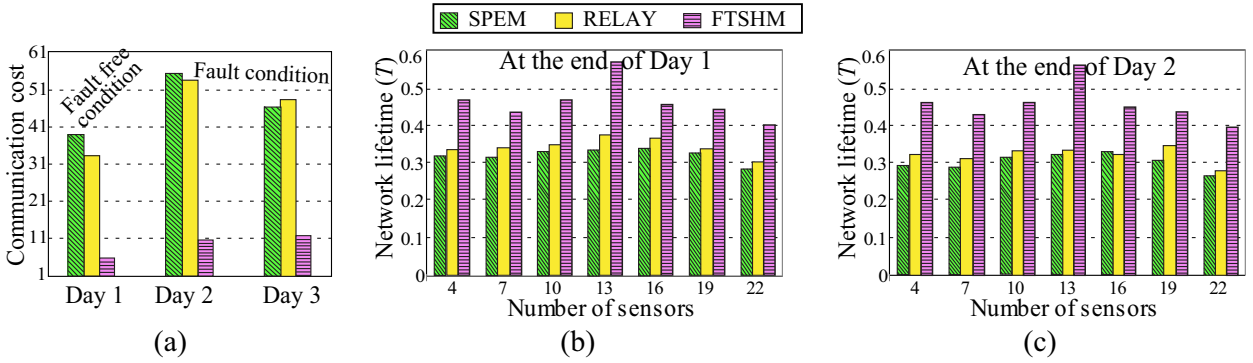


Fig. D.3. Impact of sensor faults in SHM performance: (a) the total communication cost on different days; (b)  $T$  vs. the number of sensors (at the end of Day 1); (c)  $T$  vs. the number of sensors (at the end of Day 2).

approaches. We reveal severe changes in both SPEM and RELAY. We discover that such changes are not actually affected by damage, but by the sensor faults in the WSN—where there are no backup sensors available at the locations of the failed sensors, and no recovery solutions from the situations. The changes indicate that, in the case of the WSN, the engineering deployment methods bring more chances for WSNs to be prone to faults, where the communication distance varies greatly and the distance-based link does not provide good link quality. The changes are more than 35% in SPEM.

In contrast, RELAY has better performance at the faulty sensor locations, but worse performance at some other locations. This is due to random deployment. We note that the distorted mode shapes in both SPEM and RELAY may lead to *indication* for damage detection (although this is false-negative detection). A closer look reveals that the identified mode shape is slightly distorted ( $< 6\%$ ) in FTSHM on both Day 1 and Day 2, which does not have much of an impact in SHM. This is because a sensor needs to adjust its communication range or sensing range, and keeps  $k$ -connectivity to cover the failed sensor location.

During the sensor deployment, we confirm achieving the high link quality in backup sensor placement. In FTSHM, we do not allow neighboring sensor data

aggregation, so that all sensor data is not influenced. For a sensor failure, 0 (zero) is considered as a identified frequency at the location of a failed sensor. The results in Fig. D.2 depict that FTSHM is able to overcome the situations of faults in the network, and prove the high performance in monitoring the health of the structure.

Finally, we discuss the network performance under the sensor fault condition in the WSN. Fig. D.3(a) depicts that a higher communication cost is needed in SPEM and RELAY for maintaining network connectivity after failure of each sensor. The communication cost is the highest on Day 2, since all of the faults are injected on Day 2. The communication cost is still higher on Day 3 than on Day 1. This is because, the WSN still strives to provide the coverage for the failed sensor locations, and connectivity between the neighboring nodes around the failed sensors.

Fig. D.3(b) and D.3(c) illustrate  $T$  under sensor faults at the end of Day 2 and Day 3, respectively. We found that  $T$  decreases (around 4% in SPEM, and around 6% in RELAY) after each sensor fault or failure. Besides the energy consumption for the communication cost (in case of the recovery from sensor faults), they also require a higher energy consumption for the data transmission. As a result, the reduction on  $T$  is larger in SPEM and RELAY than in FTSHM.

There is a higher reduction on  $T$  in both SPEM and RELAY on the end of Day 3 than on the end of Day 2. This implies that the reduction on  $T$  in both SPEM and RELAY becomes larger as time goes on.

## REFERENCES

- [1] C. A. Peckens and J. P. Lynch, "Embedded linear classifiers on wireless sensor networks for damage detection," in *Sensors and Smart Structures Technologies for Civil, Mechanical, and Aerospace Systems*, 2013.
- [2] K. Anshuman, J. Ratneshwar, W. Matthew, and J. Kerop, "Damage detection in an experimental bridge model using HilbertHuang transform of transient vibrations," *Structural Control and Health Monitoring*, vol. 20, no. 1, pp. 1–15, 2013.
- [3] S. Beygzadeh, E. Salajegheh, P. Torkzadeh, J. Salajegheh, and S. Naseralavi, "Optimal sensor placement for damage detection based on a new geometrical viewpoint," *International Journal of Optimization in Civil Engineering*, vol. 3, no. 1, pp. 1–21, 2013.
- [4] B. Li, Z. Sun, K. Mechitov, C. Lu, S. J. Dyke, G. Agha, and B. F. Spencer, "Realistic case studies of wireless structural control," in *Proceedings ACM/IEEE 4th International Conference on Cyber-Physical Systems (ICCPs'13)*, 2013.
- [5] Z. Xing and A. Mita, "A substructure approach to local damage detection of shear structure," *Structural Control and Health Monitoring*, vol. 19, no. 2, pp. 309–318, 2012.
- [6] A. Jindal and M. Liu, "Networked computing in wireless sensor networks for structural health monitoring," *IEEE/ACM Transactions on Networking*, vol. 20, no. 4, pp. 1203–1216, 2012.
- [7] D. Musiani, K. Lin, and T. S. Rosing, "Active sensing platform for wireless structural health monitoring," in *Proc. of ACM IPSN*, 2007, pp. 390–399.
- [8] G. Hackmann, F. Sun, N. Castaneda, C. Lu, and S. Dyke, "A holistic approach to decentralized structural damage localization using wireless sensor networks," *Computer Communications*, vol. 36, no. 1, pp. 29–41, 2012.
- [9] X. Liu, J. Cao, S. Lai, C. Yang, H. Wu, and Y. Xu, "Energy efficient clustering for WSN-based structural health monitoring," in *Proc. of IEEE INFOCOM*, 2011, pp. 2768–2776.
- [10] S. Kim, S. Pakzad, D. Culler, J. Demmel, G. Fennes, S. Glaser, and M. Turon, "Health monitoring of civil infrastructures using wireless sensor networks," in *Proc. of ACM IPSN*, 2007, pp. 254–263.
- [11] G. Hackmann, W. Guo, G. Yan, Z. Sun, C. Lu, and S. Dyke, "Cyber-physical codesign of distributed structural health monitoring with wireless sensor networks," *IEEE Transactions on Parallel and Distributed Systems*, p. <http://doi.ieeecomputersociety.org/10.1109/TPDS.2013.30>, 2013.
- [12] M. Z. A. Bhuiyan, G. Wang, J. Cao, and J. Wu, "Energy and bandwidth-efficient wireless sensor networks for monitoring high-frequency events," in *Proc. of IEEE SECON*, 2013.
- [13] M. Z. A. Bhuiyan, G. Wang, and J. Cao, "Sensor placement with multiple objectives for structural health monitoring in WSNs," in *Proc. of IEEE HPCC*, 2012, pp. 699–706.
- [14] M. Z. A. Bhuiyan, J. Cao, G. Wang, and X. Liu, "Energy-efficient and fault-tolerant structural health monitoring in wireless sensor networks," in *Proc. of IEEE SRDS*, 2012, pp. 301–310.
- [15] X. Liu, J. Cao, M. Z. A. Bhuiyan, S. Lai, H. Wu, and G. Wang, "Fault tolerant WSN-based structural health monitoring," in *Proceedings of the IEEE/IFIP 41st International Conference on Dependable Systems & Networks (DSN)*, 2011, pp. 37–48.
- [16] M. Z. A. Bhuiyan, J. Cao, and G. Wang, "Backup sensor placement with guaranteed fault tolerance for structural health monitoring," in *Proc. of 6th Beijing-Hong Kong International Doctoral Forum (BJ-HK PhD Forum)*, 2011, pp. 1–4.
- [17] A. Krause, C. Guestrin, A. Gupta, and J. Kleinberg, "Robust sensor placements at informative and communication-efficient locations," *ACM Transactions on Sensor Networks*, vol. 20, no. 7, pp. 1–33, 2011.
- [18] O. Gnawali, R. Fonseca, K. Jamieson, D. Moss, and P. Levis, "Collection tree protocol," in *Proceedings of the 7th ACM Conference on Embedded Networked Sensor Systems (SenSys'09)*, 2009, pp. 1–14.
- [19] Crossbow Technology Inc., Imote2 hardware reference manual.
- [20] J. A. Rice and B. F. Spencer, "Flexible smart sensor framework for autonomous full-scale structural health monitoring," *NSEL Report No. 18*, University of Illinois at Urbana-Champaign, 2009.
- [21] TinyOS reference manual. [Online]. Available: <http://www.tinyos.net>
- [22] M. Maroti, B. Kusy, G. Simon, and A. Ledeczi, "The flooding time synchronization protocol," in *Proceedings of the 2nd international conference on Embedded networked sensor systems (SenSys'04)*, 2004, pp. 39–49.