NSEL Report Series
Report No. NSEL-038
June 2015

Decentralized Identification and Multimetric Monitoring of Civil Infrastructure using Smart Sensors

Sung-Han Sim

and

Billie F. Spencer, Jr.

Department of Civil and Environmental Engineering
University of Illinois at Urbana-Champaign

UILU-ENG-2015-1804

ISSN: 1940-9826

The Newmark Structural Engineering Laboratory (NSEL) of the Department of Civil and Environmental Engineering at the University of Illinois at Urbana-Champaign has a long history of excellence in research and education that has contributed greatly to the state-of-the-art in civil engineering. Completed in 1967 and extended in 1971, the structural testing area of the laboratory has a versatile strong-floor/wall and a three-story clear height that can be used to carry out a wide range of tests of building materials, models, and structural systems. The laboratory is named for Dr. Nathan M. Newmark, an internationally known educator and engineer, who was the Head of the Department of Civil Engineering at the University of Illinois [1956-73] and the Chair of the Digital Computing Laboratory [1947-57]. He developed simple, yet powerful and widely used, methods for analyzing complex structures and assemblages subjected to a variety of static, dynamic, blast, and earthquake loadings. Dr. Newmark received numerous honors and awards for his achievements, including the prestigious National Medal of Science awarded in 1968 by President Lyndon B. Johnson. He was also one of the founding members of the National Academy of Engineering.

Contact:
Prof. B.F. Spencer, Jr.
Director, Newmark Structural Engineering Laboratory
2213 NCEL, MC-250
205 North Mathews Ave.
Urbana, IL 61801
Telephone (217) 333-8630
E-mail: bfs@illinois.edu

This technical report is based on the first author's doctoral dissertation of the same title, which was completed in May 2011. The second author served as the dissertation advisor for this work.

Financial support for this research was provided in part by the National Science Foundation under NSF Grants No. CMS-060043, CMMI-0724172, CMMI-0928886, and CNS-1035573. In addition, we would like to thank the numerous collaborators on this work, including Gul Agha, Kirill Mechitov, Ryan Giles, Tomonori Nagayama, Sheng-Lin Lin, Manuel Ruiz-Sandoval, Hongki Jo, Chia-Ming Chang, Brian Phillips, Robin Kim, Parya Moinzadeh, Lauren Linderman, Jian Li, Nicholas Wierschem, Shinae Jang, Jong-Woong Park, and Soojin Cho.

The cover photographs are used with permission. The Trans-Alaska Pipeline photograph was provided by Terra Galleria Photography (http://www.terragalleria.com/).

ABSTRACT

Wireless Smart Sensor Networks (WSSNs) facilitates a new paradigm to structural identification and monitoring for civil infrastructure. Conventionally, wired sensors and central data acquisition systems have been used to characterize the state of the structure, which is quite challenging due to difficulties in cabling, long setup time, and high equipment and maintenance costs. WSSNs offer a unique opportunity to overcome such difficulties. Recent advances in sensor technology have realized low-cost, smart sensors with on-board computation and wireless communication capabilities, making deployment of a dense array of sensors on large civil structures both feasible and economical. However, as opposed to wired sensor networks in which centralized data acquisition and processing are common practice, WSSNs require decentralized algorithms due to the limitation associated with wireless communication; to date such algorithms are limited. This research develops new decentralized algorithms for structural identification and monitoring of civil infrastructure. To increase performance, flexibility, and versatility of the WSSN, the following issues are considered specifically: (1) decentralized modal analysis, (2) efficient decentralized system identification in the WSSN, and (3) multimetric sensing. Numerical simulation and laboratory testing are conducted to verify the efficacy of the proposed approaches. The performance of the decentralized approaches and their software implementations are validated through full-scale applications at the Irwin Indoor Practice Field in the University of Illinois at Urbana-Champaign and the Jindo Bridge, a 484 meter-long cable-stayed bridge located in South Korea. This research provides a strong foundation on which to further develop long-term monitoring employing a dense array of smart sensors. The software developed in this research is opensource and is available at: http://shm.cs.uiuc.edu/.

CONTENTS

CHAPTER 1: INTRODUCTION 1

CHAPTER 2: BACKGROUND 3
2.1 Structural health monitoring 3
2.2 Modal analysis 5
2.3 Wireless smart sensor network 6
2.3.1 Smart sensor 6
2.3.2 Data processing schemes in wireless smart sensor network 7
2.3.3 System identification methods in WSSNs 10
2.4 Multimetric sensing 10
2.5 Summary 11

CHAPTER 3: DECENTRALIZED STRATEGIES FOR SHM USING WSSN 12
3.1 SHM application development using the ISHMP Services Toolsuite 12
3.1.1 ISHMP Services Toolsuite 12
3.1.2 Network-wide sensing service: *SensingUnit* 15
3.2 Decentralized independent processing 17
3.2.1 Independent processing application: *IndependentProcessingPSD* 17
3.2.2 Independent processing application: *CableTensionEstimation* 18
3.3 Decentralized coordinated processing 19
3.3.1 Network topology 20
3.3.2 Control flow 20
3.3.3 Fault tolerance 21
3.3.4 Data processing methods 22
3.4 Summary 22

CHAPTER 4: DECENTRALIZED MODAL ANALYSIS 24
4.1 Formulation 24
4.1.1 Local feature extraction 24
4.1.2 Determination of global modal properties 25
4.2 Examples 29
4.2.1 Plate model 29
4.2.2 Truss model 35
4.3 Summary 40

CHAPTER 5: DECENTRALIZED RANDOM DECREMENT TECHNIQUE 42
5.1 Random decrement technique 42
5.2 RDT-based decentralized data aggregation 44
5.3 Numerical validation 46
5.4 Experimental validation 51
5.4.1 Experimental setup 51
5.4.2 Data acquisition and in-network processing 52
5.4.3 Data processing at base station 53
5.5 Summary 55

CHAPTER 6: DAMAGE DETECTION USING MULTIMETRIC SENSING 56
6.1 Damage locating vector method.....56
6.2 Derivation.....57
6.2.1 Strain flexibility matrix.....57
6.2.2 Multimetric data in the stochastic DLV method.....58
6.2.3 DLV from multimetric data.....60
6.3 Numerical validation.....61
6.3.1 Overview of simulation.....61
6.3.2 Performance evaluation: false negative and positive detections.....63
6.4 Summary.....67

CHAPTER 7: FULL-SCALE VALIDATION 69
7.1 Decentralized modal analysis at the football field.....69
7.1.1 Irwin Indoor Practice Field.....69
7.1.2 Sensor deployment.....69
7.1.3 Decentralized data acquisition and processing.....71
7.1.4 Summary.....74
7.2 Jindo Bridge deployment.....75
7.2.1 Jindo Bridges.....75
7.2.2 WSSN deployment.....76
7.2.3 Software configuration.....80
7.2.4 Decentralized modal analysis.....82
7.2.5 Estimation of cable tension.....93
7.2.6 Summary.....99

CHAPTER 8: CONCLUSIONS AND FUTURE STUDIES 100
8.1 Conclusions.....100
8.2 Future studies.....103
8.2.1 Multimetric sensing.....103
8.2.2 Rare event monitoring.....104
8.2.3 Over-the-air programming.....104
8.2.4 Fatigue prediction.....105

REFERENCES.....106

APPENDIX A: COMMAND LINE USAGE OF APPLICATIONS.....114

APPENDIX B: *RemoteCommand* 122

APPENDIX C: DECENTRALIZED CABLE TENSION ESTIMATION.....125

Chapter 1

INTRODUCTION

The term 'civil infrastructure' collectively refers to systems (e.g., bridges, roads, railways, electrical power networks, pipelines, etc.,) which are required for our society to properly function. In particular, modern industrialized nations such as the United States highly rely on civil infrastructure and thus invest a huge amount of resources to construct and maintain them. Indeed, the five-year investment need is estimated 2.2 trillion dollars in the United States (ASCE Report Cards 2009). During the last century, thriving economy and technological advances have allowed an unprecedented amount of civil infrastructure to be built. Focus has been placed on construction, rather than maintenance of existing infrastructure; as a result, degrading infrastructural systems are rapidly increasing. Indeed, over 26 % of the bridges in the United States were rated as structurally deficient and functionally obsolete as of December 2008 by the Department of Transportation (ASCE Report Cards 2009). Unless appropriately handled, aging and deteriorating infrastructure systems can pose a significant threat to public safety as well as proper functioning of our society. Thus, monitoring the state of a structure to ensure timely maintenance is of importance to prolong the lifetime of our valuable assets and prevent structural failures that may occur otherwise. In addition to the long-term monitoring, prompt structural damage assessment and emergency evacuation plans after extreme events such as earthquakes, hurricanes, and tornadoes are vital to improve public safety. All these issues are addressed in Structural Health Monitoring (SHM).

SHM typically consists of (1) measuring data such structural responses (e.g., strain, displacement, velocity, and acceleration) and environmental variables (e.g., temperature, and wind velocity), (2) analyzing the sensor data for feature extraction, and (3) making decisions for further actions or maintenance plans. The research reported herein is focused on the first two issues.

Data acquisition in traditional SHM systems is based on wired sensors connected to a centralized data collection repository. All sensor data is aggregated at this central repository, where all data processing takes place to extract structural features and information. This centralized data acquisition and processing approach in the wired sensor network is common practice in traditional SHM systems; however, high cost and installation difficulties (Lynch *et al.* 2003) have prevented SHM from wider adoption in large-scale civil structures. For instance, imagine a SHM system of the Golden Gate Bridge; miles of cables would be required to connect the central base station to a dense array of sensor nodes distributed along the deck, towers, and cables; installation would be both costly and time-consuming, and maintenance would be challenging.

Wireless smart sensor networks provide a promising alternative to the traditional SHM approach. Smart sensors commonly refer to devices that are small, inexpensive, capable of wireless communication, and have on-board processing capabilities (Spencer *et al.* 2004). In the last decades, many academic and commercial smart sensors have been developed. Significant efforts have been devoted to various issues in Wireless Smart Sensor Networks (WSSNs), including data acquisition, processing, and damage detection. The majority of smart sensor research has focused on emulation of traditional

wired sensor networks employing centralized data acquisition and processing. Such approaches have proven to be intractable because transferring all sensor data saturate the limited bandwidth found in wireless communication and thus causes network congestion. To illustrate this point, Pakzad *et al.* (2008) reported that 10 hours were required to transmit 80 seconds of data at 1000 Hz for 56 wireless sensors back to the base station. Indeed, decentralized data acquisition and processing schemes are considered to be essential to ensure the scalability of WSSNs required to enable a dense array of sensors deployed on full-scale civil structures. Considering that structural damage is a local phenomenon, densely deployed sensor networks are expected to enhance the damage detection capability of SHM systems.

The advantage of smart sensors is not limited to SHM; modal analysis can be conducted more efficiently by employing smart sensors. Modal analysis and SHM have many common aspects in that measuring and processing structural response is involved. As in SHM systems, conventional modal analysis requires sensors, wires, and a data acquisition system. In particular, modal analysis is conducted in a short period of time to characterize dynamic properties of a structure (e.g., natural frequencies, mode shapes, and damping factors), whereas SHM is intended to track the structural health for a relatively longer time. Thus, installation convenience is more critical when conducting a modal survey; the benefit of the WSSN is evident. However, decentralized algorithms for modal analysis are sparse in the literature. A means to estimate global dynamic properties from the decentralized network is desired for the WSSN to be utilized effectively in conducting a modal survey.

While decentralized approaches enable scalable WSSNs for monitoring full-scale civil infrastructure, introducing multimetric sensor data to the WSSN can enhance the performance of WSSNs. As the human sensory system involves various perceptions, multimetric sensing can potentially enrich essential information for the condition of a structure, enabling more precise diagnosis. The same principle can be applied to SHM. Indeed, most monitoring systems for full-scale civil infrastructure takes advantage of multiple types of sensors. For example, sensors installed on the Akashi-Kaikyo Bridge include seismometers, anemometers, velocity gauges, GPS sensors, displacement gages, TMD displacement gauges, and thermometers (Fujino *et al.* 2000b). Integrating multimetric sensor data into the SHM process has the potential to enhance the capability to precisely characterize the structural state; however, this integration has yet to be fully explored. In particular, such approaches for the smart sensors are rarely found.

This research develops new decentralized algorithms for structural identification and multimetric monitoring of civil infrastructure. The following four features are considered to increase performance, flexibility and versatility of WSSNs: (1) decentralized modal analysis that reliably determines global modal properties based on the local modal information, (2) efficient system identification for the WSSN, and (3) multimetric sensing for damage detection. These WSSN applications are numerically and experimentally validated, including laboratory-scale testing using a truss bridge structure and full-scale experiments in the Irwin Indoor Practice Field at the University of Illinois at Urbana-Champaign and the Jindo Bridge located in the South Korea. The following chapters provide background on this study, implementation and development of decentralized approaches for the WSSN, experimental validation, conclusions and future research.

BACKGROUND

2.1 Structural health monitoring

SHM is often referred to as a process of implementing damage detection strategies for aerospace, mechanical and civil engineering infrastructure (Sohn *et al.* 2003); however, a broader definition can include a spectrum of applications such as design verification, assessment of structural state after catastrophic events, control of the construction process, and assisting with building and bridge maintenance. In the civil engineering field, SHM has become a prominent tool to as address problems associated with deteriorating civil infrastructure. For instance, the collapse of the Silver Bridge between West Virginia and Ohio in 1967 subsequently initiated the US government to develop and implement the National Bridge Inspection Standards (NBIS) (Small *et al.* 1999). In addition, the Korean government has been conducting bridge monitoring since the Sungsu Bridge over the Han River collapsed in 1994 resulted in 32 casualties. As such, assessing damage at an early stage and retrofitting or repairing structures in a timely manner is of paramount importance for public safety as well as to reduce maintenance cost.

Damage detection is one of the central objectives of SHM. In bridge inspections conducted biannually by the Federal Highway Administration (FHWA) in the United States, visual inspection and simple tap testing are common practice for damage assessment. However, one of the common features of all bridge failures, including the I-35 bridge collapse in Minneapolis, is that they have all undergone and passed regular inspection, leading to the conclusion that the current inspection approach is insufficient. Indeed, damage is often lurking in locations that are not accessible. The need for more accurate damage identification has driven the research community to pursue various approaches for damage detection that can be generally categorized as either local or global. Local and global methods can be used in combination: once global damage detection methods determine the presence and possible region of damage, local damage detection methods can find more specific damage locations (Chang *et al.* 2003). Non-Destructive Evaluation (NDE) techniques are typically used for local damage detection. With prior knowledge of damage location, the NDE approaches can be applied to find the exact location and extent of damage. More information regarding the NDE methods can be found in Chang and Liu (2002). SHM strategies in this study are focused on global damage detection approaches.

As damage detection has been extensively studied, a number of approaches have been developed as summarized by Doebling *et al.* (1996), Sohn *et al.* (2003), and Chang *et al.* (2003). The cited papers mostly utilize vibration responses to extract the structural feature change before and after damage. Doebling *et al.* (1996) categorized the vibration-based damage detection methods as:

- Natural frequency change
- Mode shape change
- Mode shape curvature/strain mode shape change

- Dynamically measured flexibility matrix
- Structural model updating

Early research on vibration-based damage detection considered changes in modal parameters (e.g., natural frequency and mode shape) before and after damage. However, the natural frequency and the mode shape, global properties of a structure, have been shown to be quite insensitive to localized damage (Begg *et al.* 1976; Nataraja 1983; Fox 1992; Srinivasan and Kot 1992); damage detection using the modal property change was considered to be impractical. Mode shape curvature was considered to find more damage sensitive quantities. Pandey *et al.* (1991) proposed the absolute change in mode shape curvature as a good damage indicator. Chance *et al.* (1994) showed numerically calculated curvature was erroneous, and thus used dynamic strains instead. However, the difficulty for methods employing dynamic strains lies in the fact that noise in dynamic strain measurements is in general higher than that in accelerations (Sohn and Law 2001).

Damage detection methods based on the dynamically measured flexibility matrix have also been developed. Unlike the stiffness matrix, the flexibility matrix is insensitive to higher frequency modes that are generally difficult to determine from measured data (Gao *et al.* 2004). This feature focused research efforts flexibility-based, rather than the stiffness-based, damage detection. Pandey and Biswas (1994) used changes in the measured flexibility matrix. Damage locations could be found from the first two lower modes. Bernal (2002) presented the Damage Locating Vector (DLV) method that utilizes changes in the flexibility matrix due to damage. The DLV method can find damage only with a limited number of DOFs in which the flexibility matrix is defined, if the DOFs are in the proximity of the localized damage. The DLV method was expanded to the Stochastic DLV (SDLV) method for output-only cases (Bernal 2006), and further generalized to take advantage of the transfer function matrix that can be thought as a flexibility matrix extended to the frequency domain (Bernal 2007a; Bernal 2007b).

Although researchers have continued to develop efficient damage identification methods, existing approaches have achieved only limited success, mostly through numerical simulations and laboratory experiments. Although there are some examples for damage detection of full-scale bridges (Maeck *et al.* 2001; Lee *et al.* 2005), full-scale implementation of damage detection approaches are not yet considered mature. Indeed, Brownjohn (2007) indicated that the SHM systems for civil infrastructure thus far were not necessarily intended for damage identification.

The advancement in digital computers and data acquisition systems has enabled SHM systems to be adopted in major civil infrastructure for various monitoring purposes during the last decade. In particular, the long-term monitoring system of the Yokohama Bay Bridge was used to investigate the dynamic behavior due to the seismic loadings (Siringoringo and Fujino 2006). Modal analysis results showed that there were unwanted deformation mechanisms that could potentially damage the bridge during an earthquake. The Republic Plaza in Singapore has been serving as an exemplary monitoring test bed. Brownjohn (1998) identified the natural mode of the Republic Plaza for design verification. The monitoring system was also used to characterize the earthquake loading spectrum for the building (Brownjohn *et al.* 2001). Furthermore, wind-induced lateral vibration was monitored using GPS sensors to inform local design codes (Ogaja *et al.* 2003; Brownjohn 2005). Other monitoring examples include the Akashi-Kaikyo Bridge in Japan (Fujino *et al.* 2000b), the Hakucho bridge in Japan (Fujino *et al.* 2000a), the

Tsingma Bridge in Hong Kong (Wong 2004), the Stonecutters Bridge in Hong Kong (Wong 2004), the Bill Emerson Bridge in the United States (Caicedo *et al.* 2002; Celebi *et al.* 2004), and the Punggol E26 Buildings in Singapore (Glisic *et al.* 2003). In addition, Ko and Ni (2005) summarized 20 bridge monitoring systems in China.

Although the full-scale SHM examples have demonstrated successful installation and monitoring to some extent, issues still remain. First of all, current SHM systems based on the traditional wired sensor system are quite expensive. For instance, the 300 sensors deployed for on the Tsing-ma Bridge cost approximately $7.8 million (Lynch and Loh 2006). This high cost is one of the most important issues that have hindered wider adoption of the SHM systems. In addition, wiring sensors poses difficulties in installation as well as significantly contributes to the cost. In particular, ease of installation is critical for monitoring in the construction stage. A more efficient means for SHM is desired for large-scale civil structures.

Smart sensors have emerged as a promising alternative to traditional SHM systems employing wired sensors. Recent advances in sensor technology have enabled low-cost smart sensors that have wireless communication, on-board processing, and multimetric sensing capabilities; even lower prices are expected once smart sensors are in mass production. These features offered by smart sensors provide the potential of realization of a dense array of sensors for a low cost. Knowing that damage is a local phenomenon, a densely deployed sensor array is essential for accurate damage localization. Thus, networks employing smart sensors have the potential to leverage SHM techniques.

2.2 Modal analysis

Modal surveys have been common practice to identify dynamic characteristics of a structure. Since research on structural vibration issues started in 1930s focusing on understanding the dynamic behavior of aircraft (Maia and Silver 2001), structural dynamic effects have been widely considered in various applications. Identified dynamic characteristics are often used for model updating, design verification, and improvement of serviceability. A typical example is the retrofit of the London Millennium Footbridge to address unexpected large lateral vibrations caused by pedestrians (Dallard *et al.* 2001). To control bridge vibrations, viscous fluid dampers and tuned mass dampers were installed on the bridge. In designing this damping system, understanding the dynamic characteristics of the as-built bridge was critical to ensuring efficient operation. Detailed theory and practice of modal surveys can be found in Ewins (1986).

Smart sensors provide a new paradigm for conducting modal surveys. As in the SHM systems, traditional modal surveys consist of a centralized data acquisition system with wired sensors that can be replaced with smart sensors. Wireless communication and cost effectiveness of smart sensors can address difficulties in wiring sensors, high cost, and long setup time. Considering that modal surveys are conducted in a relatively short period of time compared to monitoring of structures, installation convenience is critical in the modal survey. However, smart sensors are not widely adopted in modal surveys due to lack of appropriately designed software, which will be discussed in subsequent chapters.

2.3 Wireless smart sensor network

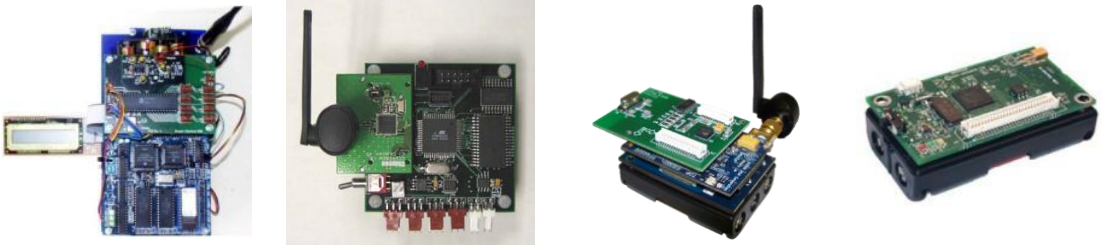
2.3.1 Smart sensor

Spencer *et al.* (2004) defined a *smart sensor* as having four features: (1) on-board Central Processing Unit (CPU), (2) small size, (3) wireless, and (4) the promise of being low-cost. The on-board microprocessor enables near real-time data processing in the network for identification of dynamic features, structural health diagnosis (e.g., damage assessment), and decision making. MEMS technology associated with sensing makes the smart sensor small, and wireless communication capability lowers the installation cost by removing the need for wires. By utilizing MEMS and microprocessor that are already in mass production and thus quite inexpensive, the cost of smart sensors is decreasing. These features offered by smart sensors can be utilized to resolve the difficulties that traditional wired monitoring systems have.

One of the earliest smart sensors developed for civil structures is the wireless modular monitoring system (WiMMS) by Straser and Kiremidjian (1998). WiMMS is made using commercial off-the-shelf (COTS) components: Motorola family of 68HC11 microprocessors, Proxim Proxlink MSU2 radio unit, and dimensions of 15 cm by 13 cm by 10 cm, which correspond to the features of smart sensors described earlier. Since then, many academic and commercial smart sensors have been developed as reviewed by Lynch and Loh (2006). Some of the smart sensors are compared in Table 2.1.

Table 2.1 Specification of smart sensors

	Straser and Kiremidjian (1998)	Swartz and Lynch (2006)	Imote2 (MEMSIC)	Mica2 (MEMSIC)
Category	Academic	Academic	Commercial	Commercial
Processor	Motorola 68HC11	Atmel ATmega128	XScalePXA271	Atmel ATmega128L
Clock Speed (MHz)	2.1	16	13 – 416	8
Bus size (bits)	8	8	32	8
Program Memory (bytes)	16K	128K	32K	128K
RAM (bytes)	32K	128K	256K+32M	4K
A/D Resolution (bits)	16	16	n/a	10
Power	9V battery	7.5V battery	3 x AAA	2 x AA

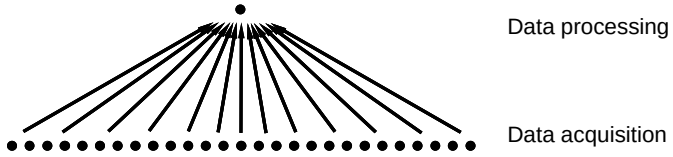


(a) Straser and Kiremidjian 1998 (b) Swartz and Lynch (2006) (c) Imote2 (d) Mica2

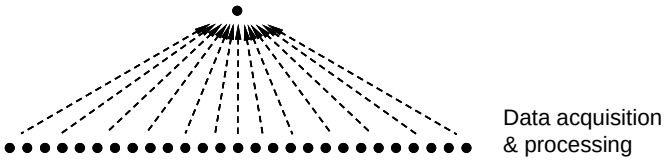
Figure 2.1 Smart sensors

2.3.2 Data processing schemes in wireless smart sensor network

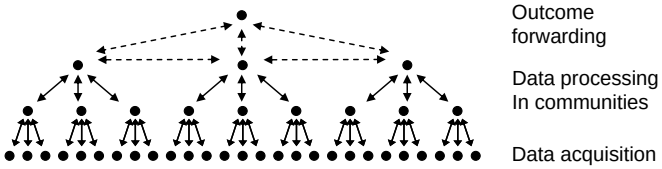
Although some challenges, such as power consumption and long term reliability, still need to be resolved for the smart sensors to be more widely adopted, available smart sensors are already quite capable and can be expected to replace traditional wired sensors for many applications. One particularly critical challenge relates to how the data is processed. Centralized data acquisition and processing schemes (see Figure 2.2a) that are commonly used in traditional wired sensor systems cannot be employed in WSSNs due to the limitation in wireless data communication speeds; bringing all data to a centralized location will result in severe data congestion in the WSSN. Indeed, sensor networks measuring dynamic response of a structure can produce a huge amount of data, which contains redundant information for characterizing the state of the structure. Thus, the lack of scalability in the centralized data acquisition and processing scheme limits wider adoption of WSSN systems. Considering WSSN for full-scale civil infrastructure directed toward a dense array of sensors, scalable sensor networks are essential in that the network scheme can be expanded to any size of networks within the capability of smart sensors.



(a) Centralized data collection.



(b) Decentralized independent processing.



(c) Decentralized coordinated processing.

Figure 2.2 Data acquisition and processing schemes.

To resolve this issue for the scalable WSSNs, a number of decentralized approaches have been proposed for conducting structural health monitoring and damage detection on networks of smart sensors. One of the decentralized approaches uses independent processing, as shown in Figure 2.2b (Tanner *et al.* 2003, Caffrey *et al.* 2004, Lynch *et al.* 2004a, Lynch *et al.* 2004b, and Nitta *et al.* 2005; Park 2009). Each sensor node processes measured data independently without communicating with other sensor nodes. The processed data, typically FFT or signature analysis, is then sent to the base station. The total amount of data transferred over the radio in the network is quite small. Although sensor networks based on this processing scheme is scalable, important spatial information (e.g. mode shape) cannot be extracted.

The decentralized approach proposed by Gao *et al.* (2005) employs a coordinated computing strategy, often called Distributed Computing Strategy (DCS), as shown in Figure 2.2c, which has the ability to capture local spatial information. The sensor network in this scheme is divided into hierarchical communities, in which sensor nodes within each community communicate with each other in processing data; communication between communities is conducted through each community's cluster-head. Because communication for sensor data takes place within each community consisting of a limited number of sensors, the total amount of data is nearly proportional to the network size and the network is therefore scalable. Gao *et al.* (2005) employed this processing scheme for damage detection of a truss structure in Figure 2.3 using the DLV method. Although a wired sensor system was used, each sensor community successfully localized damage, showing that the coordinated computing strategy was promising for WSSNs.

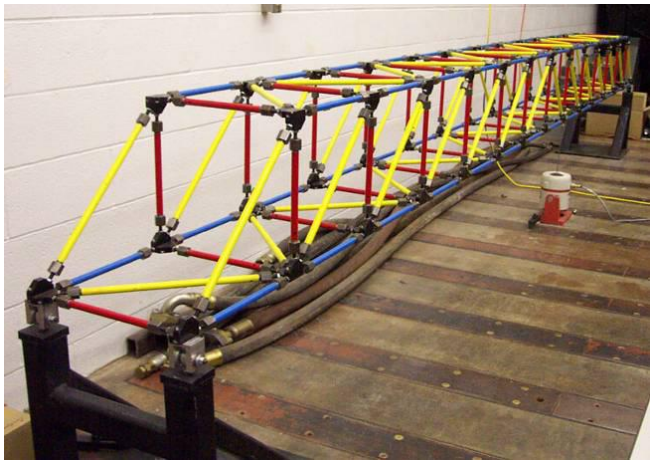
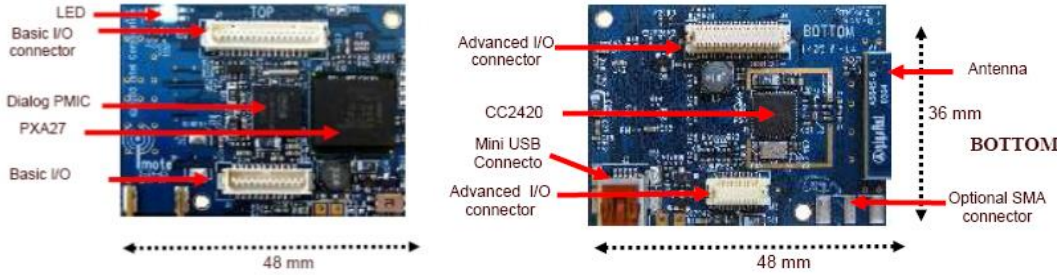


Figure 2.3 Truss structure in the Smart Structures Technology Laboratory.

Nagayama and Spencer (2007) implemented the DCS logic in a WSSN employing the Imote2 smart sensors. An output-only identification approach, the Natural Excitation Technique (NExT) (James *et al.* 1993), was employed in conjunction with Eigensystem Realization Algorithm (ERA) (Juang and Pappa 1985). The SDLV method in the distributed computing environment was programmed and embedded in each sensor node. Issues regarding time synchronization, synchronized sensing, and reliable communication were also addressed. The network topology as described in Figure 2.2c is based on the three types of sensors according to their function in the network: (a) gateway node, (b) cluster-head, and (c) leaf node. Both cluster-head and leaf node measure acceleration and

do data processing to get correlation functions using a sensor data from the cluster-head as a reference signal. The cluster-head collects the correlation functions for modal parameter estimation and damage localization. Only damage information is supposed to be sent to the gateway node. Although many issues such as battery life, sensing capability, and ease of expansion should be resolved, the sensor network based on the DCS logic was shown to be promising for monitoring full-scale civil infrastructure. More detailed information can be found in Nagayama and Spencer (2007).

(a) Imote2 with an external antenna.



(b) Top (left) and bottom (right) views.

Figure 2.4 Imote2 sensor platform.

While a number of decentralized algorithms for structural health monitoring and damage detection have been published, relatively little effort has been devoted to developing such approaches for modal analysis. Zimmerman *et al.* (2008) implemented a decentralized data processing scheme on a WSSN to identify the vibration characteristics of the balcony in a historic theater in metropolitan Detroit. In the WSSN, natural frequencies were determined at each node by the Peak Picking (PP) method (see Bendat and Piersol 1993 and Felber 1993) and sent to a central node where the final natural frequencies are determined. Based on the identified natural frequencies, phase differences between the responses of each pair of sensor nodes are determined in a sequential manner (see Figure 2.5), and collected centrally to assemble the global mode shapes. While this decentralized algorithm was successfully implemented and tested, the approach requires a linear network topology and may result in substantial accumulation of errors in the global mode shapes.

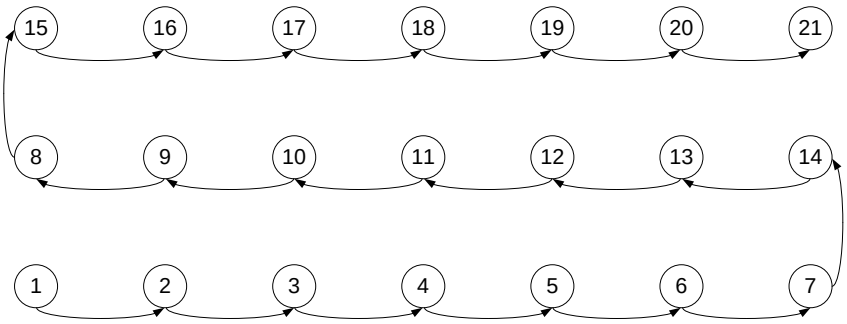


Figure 2.5 Network topology by Zimmerman *et al.* 2008.

2.3.3 System identification methods in WSSNs

As smart sensors typically are battery-powered, power management is critical for long-term monitoring. Although several approaches for power harvesting, e.g., solar power (Jang *et al.* 2010a), vibration power (Rahimi *et al.* 2003), have been reported, power consumption must still be appropriately managed. Reducing wireless communication, the most significant source of power consumption for smart sensors, can be achieved by employed embedded in-network data processing.

As previously described, Nagayama and Spencer (2007) implemented the NExT-based system identification. Measured time history data is locally processed to estimate the correlation function that is in general much smaller than the raw data. The use of NExT/ERA in the decentralized computing environment has been shown to be quite efficient from the data communication perspective.

The Random Decrement Technique (RDT) is an alternative output-only system identification method proposed by Cole (1968) that has several attractive features. The decentralized implementation of NExT by Nagayama *et al.* (2007) requires the complete time history data from the cluster-head in a group be transferred to the leaf nodes to calculate the correlation functions. In contrast, RDT only requires the trigger crossings be sent to the leaf nodes, which is typically of a much smaller size than the raw sensor data. The output of the RDT is the random decrement (RD) function, which can be used as input to system identification methods such as ERA, Ibrahim Time Domain (ITD) (Ibrahim and Mikulcik 1977; Ibrahim na d Pappa 1982), Stochastic Subspace Identification (SSI) (Peeters and De Roeck 2001; Van Overschee and De Moor 1996; Hermans and Van Der Auweraer 1999), Peak Picking (see Bendat and Piersol 1993 and Felber 1993), Frequency Domain Decomposition (FDD) (Brinker *et al.* 2001), Enhanced Frequency Domain Decomposition (EFDD) (Rodrigues *et al.* 2004), and Complex Mode Indicator Function (CMIF) (Shih *et al.* 1988; Gul and Catbas *et al.* 2008). Comparative study for the performance of various system identification methods are presented by Yi and Yun (2004).

2.4 Multimetric sensing

In SHM systems, acceleration and strain are commonly considered as one of the most important measurands; acceleration is frequently preferred due to the convenience of sensor installation and its relatively high signal-to-noise ratio. Acceleration signals are considered to contain information regarding the global behavior of structures; whereas

strain provides local information. However, because damage is a local phenomenon, damage localization using only global measurements has its limitations.

Multimetric sensing using a heterogeneous mix of measurands at various scales has the potential to provide for more accurate characterization of the state of a structure; however, to date, little effort has been devoted to the development of multimetric sensing strategies for damage detection. Studer and Peters (2004) presented a damage identification approach for composite structures using multimetric data, including: strain, integrated strain, and strain gradients measured from optical fiber sensors. Law *et al.* (2005) used a wavelet-based approach to combine acceleration and strain responses to obtain better damage detection results than using the two measurements separately. Motivated by these results, Kijewski-Correa *et al.* (2006) developed a decentralized SHM strategy that uses measured strain and acceleration and can be implemented on a wireless sensor network. A signal-based damage detection scheme based on the two-stage AR-ARX method developed at Los Alamos National Laboratory (Sohn and Farrar 2001) is employed in the sensor network. However, this method does not exploit spatial information that potentially improves damage detection and localization. Further effort is required to develop effective multimetric damage detection strategies that can utilize fully the spatial information in the inherent distributed computing environment offered by smart sensor networks.

2.5 Summary

Background on SHM and related research was reviewed in this chapter. While the necessity of SHM has been well recognized, monitoring systems are not widely used for civil infrastructure due to intrinsic limitations associated with traditional SHM systems, e.g., high cost and installation inconvenience. Smart sensors are considered as promising alternatives that have potential to leverage off SHM techniques, enabling full-scale application. In addition, modal surveys can be more efficiently conducted employing smart sensors. Because centralized data acquisition and processing schemes conventionally used for wired sensor systems are not suitable for the WSSN, decentralized processing approaches have been developed. The decentralized approaches in conjunction with appropriate system identification methods and multimetric sensing has the potential to enhance WSSN's capabilities to more effectively monitor civil infrastructure. In the following chapter, software implementations of the decentralized processing approaches on smart sensors are described.

DECENTRALIZED STRATEGIES FOR SHM USING WSSN

The decentralized data processing approaches mentioned in the previous chapter form a foundation of this study; implementation of such approaches on smart sensors is essential. Nagayama and Spencer (2007) have addressed many critical issues in software development such as network-wide time synchronization, synchronized sensing, and reliable communication, which are provided as middleware services the Illinois Structural Health Monitoring Project (ISHMP) Services Toolsuite Version 2 for the MEMSIC's Imote2 sensor platform (http://shm.cs.uiuc.edu). The decentralized approaches proposed herein are implemented on smart sensors based on the middleware services offered by the ISHMP Services Toolsuite. Following a brief overview of the ISHMP Services Toolsuite, implementations of the decentralized approaches are outlined, including:

- *SensingUnit*: a basic service component that allows WSSN applications to acquire either synchronized or unsynchronized data
- *IndependentProcessingPSD*: an application/service that calculates power spectrum of sensor data based on independent processing
- *CableTensionEstimation*: an application that estimates cable tensions based on independent processing
- *DecentralizedDataAggregation*: an application/service for data aggregation based on decentralized coordinated processing

3.1 SHM application development using the ISHMP Services Toolsuite

3.1.1 ISHMP Services Toolsuite

ISHMP is an interdisciplinary research project to develop hardware and software for monitoring civil infrastructure using wireless smart sensors. The ISHMP Services Toolsuite provides an open source library of services that are essential for developing SHM applications. These services can be used as building blocks to create customized applications to meet user needs. Some of the services and tools currently available include:

Foundation Services provide commonly used wireless sensor functionalities that are required to support higher-level applications. These services include basic communication and sensing functionalities.

- *Unified Sensing* is a TinyOS-based sensing interface for Imote2 that support for various sensor boards including MEMSIC's ITS400 and Illinois SHM-A. *Unified Sensing* service provides precise time stamping of sensor data that can be used to synchronize sensor data from different sensor nodes.

- *Time Synchronization* is a network-wide service for synchronizing the local clocks that sensor nodes in the network locally have.
- *Reliable Communication* ensures reliable data communication in a wireless sensor network. Data loss, a common problem in the wireless communication, is handled by the *ReliableComm* service.
- *Reliable Multi-Hop Communication* is the beta-implementation of any-to-any reliable multi-hop routing for WSSN applications. This ensures reliable communication in a WSSN for large structures, where all sensors cannot be reached by single-hop communication.
- *RemoteCommand* provides an efficient means for nodes to interact with each other. A command message is delivered to receiver nodes that process the designated tasks, and returns the list of responsive nodes and requested data to the sensor node. *RemoteCommand* allows the fault tolerant features to be easily implemented in sensor applications.

Application services provide the numerical algorithms necessary to implement SHM applications on the Imote2s and may also be used independently.

- *SyncSensing* resamples the sensor data, providing synchronized data. Although the local clocks are synchronized by the *Time Synchronization* service, the sensing start time and the sampling rate vary with sensor nodes with a certain amount of error. The unsynchronized sensor data is synchronized by resampling using the *SyncSensing* service.
- *CFE* estimates the correlation function between two arrays of synchronized data.
- *CPSD* estimates the cross power spectral density between two arrays of synchronized data.
- *RD* performs the Random Decrement method to estimate unscaled impulse response functions.
- *ERA* estimates modal properties (i.e., natural frequency, damping factor, and mode shapes) using the ERA. Natural Excitation Technique (NExT) (James *et al.* 1993) enables the correlation functions estimated from the *CFE* service to be used in the *ERA* service.
- *SSI* performs the covariance-driven Stochastic Subspace Identification (SSI) for modal property estimation (Hermans and Van Der Auweraer 1999).
- *FDD* performs the Frequency Domain Decomposition (FDD) algorithm for modal property estimation (Brincker *et al.* 2001).
- *SDLV* performs the Stochastic Damage Locating Vector (SDLV) method for damage localization (Bernal 2006).
- *SDDLV* performs the Stochastic Dynamic Damage Locating Vector (SDDLV) method for damage localization (Bernal 2007).

Continuous and Autonomous Monitoring services provide for continuous and autonomous WSSN operation while maintaining power efficiency.

- *SnoozeAlarm* provides sleep cycle functionality, which greatly reduces long-term power consumption. Sensors sleep for a period of time and then wake up for a relatively short period, during which they can interact with the network. The duty cycle is configurable by the user.

- *ThresholdSentry* allows a subset of the network to act as "sentry" nodes that are awakened periodically to sense data for a short period of time, determine if an interesting event is in progress, and notify the base station.
- *AutoMonitor* is a high-level network management application that coordinates each of its components in response to various events. It schedules sensing, data transfer, and *ThresholdSentry* operation according to a user-specified policy, allowing the network to operate unattended.

Tools and Utilities provide network testing and debugging capabilities that are necessary in any large-scale or long-term WSSN deployment. These tools facilitate evaluation of the network conditions at the structure to determine appropriate values of adjustable system parameters, and assess power consumption and longevity issues.

- *LocalSensing* collects sensor data from the single Imote2 connected to the PC. This tool is useful for testing new sensor board hardware as well as driver software.
- *autocomm* is a terminal program that provides an interface between the PC and the Imote2 through the IIB2400 interface board. This tool is frequently used to run the application in the ISHMP Services Toolsuite.
- *TestServices* that combines numerical services, CFE, ERA, and SDLV, performs damage detection from user-defined acceleration signals. *TestServices* is an example code that illustrates how the numerical services can be combined.
- *RemoteSensing* performs either synchronized or unsynchronized sensing in remote leaf nodes. Sensor data is stored in the flash memory of each leaf node and can be retrieved later to save power consumption by reducing waiting time in the process of the central data collection.
- *SensingUnit*, a service component that performs synchronized or unsynchronized sensing, is particularly useful for SHM application development.
- *IndependentProcessingPSD* is an implementation of the independent processing approach that decentrally estimates the power spectral densities in each leaf node.
- *CableTensionEstimation* calculates tension forces of cables based on estimated natural frequencies. It uses *IndependentProcessingPSD* to obtain the power spectrum that is subsequently used to determine natural frequencies of cables.
- *DecentralizedDataAggregation* is an application for the data acquisition and processing on decentralized hierarchical sensor network.
- *DecentralizedSysID* estimates modal characteristics using NExT in conjunction with ERA. *DecentralizedDataAggregation* provides the correlation functions for NExT.

More detailed information regarding the service-oriented architecture of the ISHMP Services Toolsuite can be found in Rice and Spencer (2009). The rest of this chapter describes development of the decentralized algorithms, focusing on *SensingUnit*, *IndependentProcessing-PSD*, *CableTensionEstimation*, *DecentralizedDataAggregation*, and *GlobalModesEstimation*. The command-line usage of these service and applications are found in Appendix A.

3.1.2 Network-wide sensing service: *SensingUnit*

The decentralized processing approaches can be efficiently implemented on Imote2 smart sensors utilizing the *SensingUnit* service. *SensingUnit* is a service component in the ISHMP Services Toolsuite that receives parameters specifying sensing information (i.e., sampling rate, data length, and sensing channels) and corresponding node ID numbers in the sensor network, performs synchronized or unsynchronized sensing, and outputs measured data.

To understand the utility of *SensingUnit*, consider an application that needs to estimate auto-correlation functions of the measured responses in each of the leaf nodes. Such a program can be developed by combining *SensingUnit* and *CFE* services: sensor data, the output of *SensingUnit*, is fed into *CFE* to estimate the auto-correlation. Considering that sensing is required in all SHM applications, *SensingUnit* is a good starting point for development of a broad array of independent processing applications. *RemoteSensing* is a typical example developed based on *SensingUnit* (Miller *et al.* 2010).

SensingUnit provides two different implementations for centralized and decentralized networks. In the centralized implementation, the gateway node sends command messages directly to all leaf nodes in the sensor network, while the decentralized implementation based on the hierarchical network employs the cluster-heads to convey data to each leaf node (see Figure 2.2c). Thus, the centralized and decentralized implementations of *SensingUnit* are well suited to independent and coordinated processing approaches, respectively.

To efficiently control the network-wide timing and flow in *SensingUnit*, a state machine is utilized. A state machine is an abstract model controlling the behavior of a program, with each state representing a particular configuration of variables and components of an application. This simplifies the control structure of complex applications by isolating data and control logic relevant to each state from the remainder of the application. States are defined based on the designated task of each node in a sensor network using *SensingUnit*; transitions between states are triggered by certain events, such as completion of sensing or sending/receiving data. For example, the default state of the leaf node in *SensingUnit*, **DISABLED**, is changed to **INIT** in the initialization, and subsequently to **SYNC** when the time synchronization starts if synchronized sensing is requested. The state is then changed to **SENSING** if the time synchronization is finished. In the state **SENSING**, sensor nodes measure data that can then be resampled if so specified. This overall process is illustrated in Figure 3.1 and Figure 3.2 for centralized and decentralized implementations, respectively.

RemoteCommand is utilized to reliably distribute command messages including sensing parameters to all sensor nodes in the network. *RemoteCommand* returns the list of the leaf nodes that correctly behave; the gateway node can subsequently remove misbehaving nodes and proceed. In addition, various commands and event handlers provided by *RemoteCommand* simplify the structure of *SensingUnit*. More information regarding *RemoteCommand* from an event-driven programming perspective is found in Appendix B.

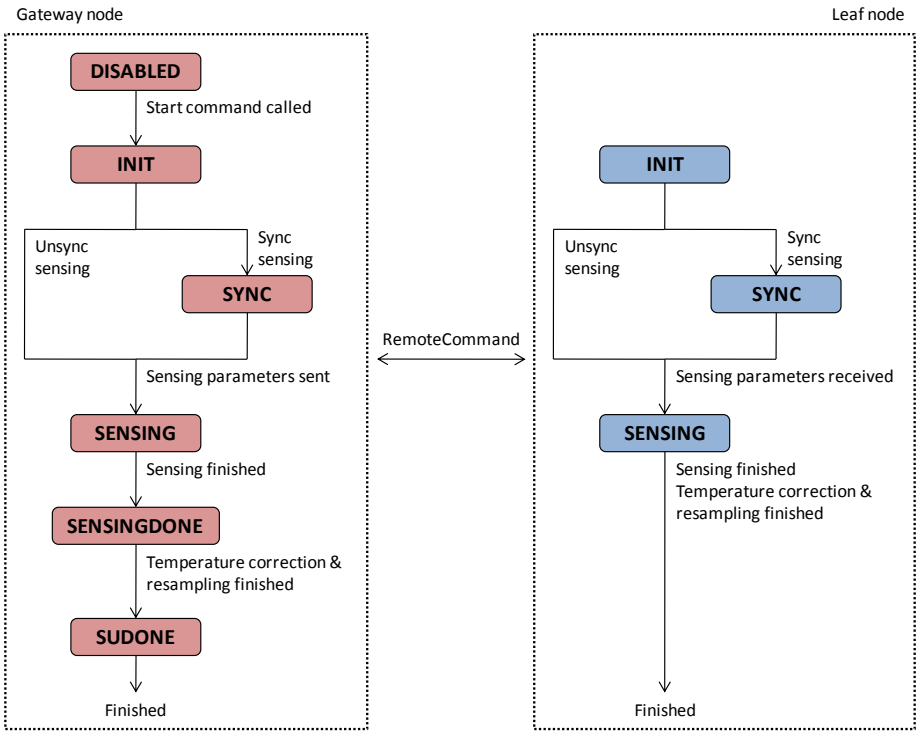


Figure 3.1 Flowchart of *SensingUnit*: centralized implementation.

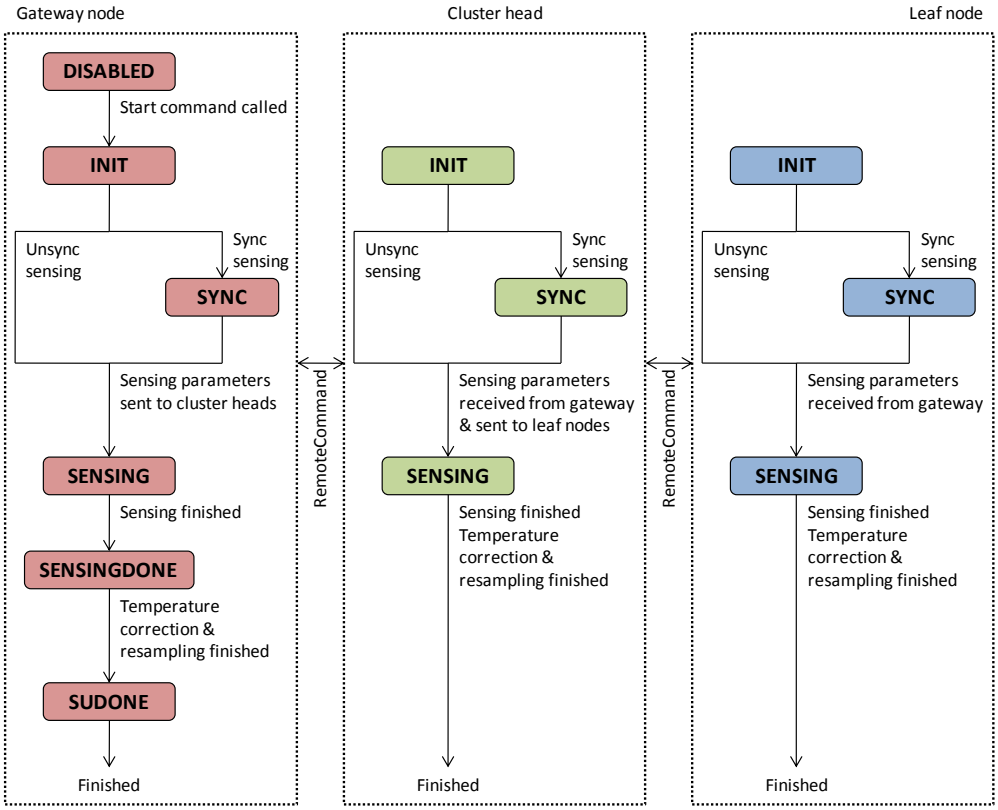


Figure 3.2 Flowchart of *SensingUnit*: decentralized implementation.

3.2 Decentralized independent processing

The decentralized independent processing approach utilizes a smart sensor's onboard computing capability to extract important information from the measured data. Each node independently processes sensor data without the need for sharing information with neighboring nodes in the network. Thus, the wireless communication requirement can be significantly lower than the centralized data collection and the coordinated processing approaches, while discarding spatial information. The ISHMP Services Toolsuite currently provides two example applications of independent processing, *IndependentProcessingPSD* and *CableTensionEstimation*.

3.2.1 Independent processing application: *IndependentProcessingPSD*

IndependentProcessingPSD is developed to estimate the power spectral density (PSD) function for the data measured on sensor nodes using *SensingUnit*. Because the PSD does not contain phase information, time synchronization is unnecessary. The PSD functions are subsequently either collected at the gateway node or retained for further analysis. The network-wide flow of *IndependentProcessingPSD* is shown in Figure 3.3. The implementation is simplified as *SensingUnit* takes care of data measurements. When *SensingUnit* finishes sensing, the gateway node delivers parameters for the PSD calculation. The leaf nodes estimate the PSD with the received parameters. The gateway node utilizes *RemoteCommand* to request and receive the PSD from each leaf node. If received from leaf nodes, the PSD in the gateway node is transferred to the base station.

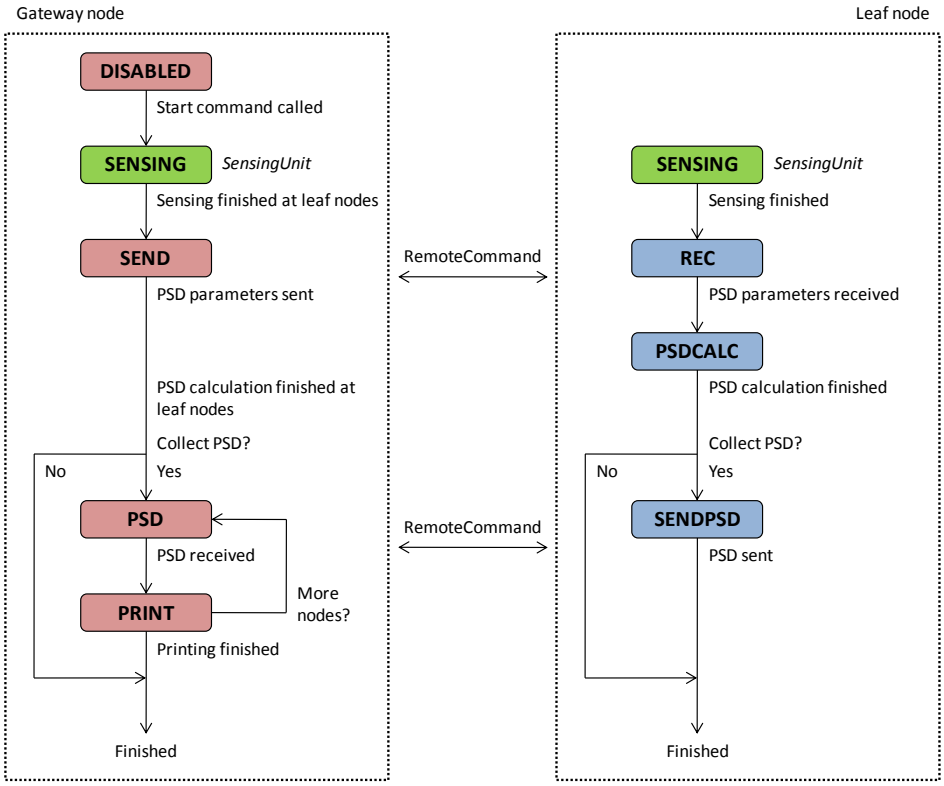


Figure 3.3 Flow of *IndependentProcessingPSD*.

Significant reduction in data communication can be achieved in the independent processing. For example, consider a sensor network of 30 nodes measuring sensor data of 21504 points. The centralized data collection transfers 30 times 21504 points (645120 points). Assuming 2048 FFT points, *IndependentProcessingPSD* collects 30 times 1023 (30690 points), reducing data communication to 4.76%.

3.2.2 Independent processing application: *CableTensionEstimation*

Cables are one of the most critical members in cable-stayed bridges; monitoring tension forces of the cables provides valuable information for assessing structural health. Several methods for estimating the cable tension are available including the direct measurement using the load cell, non-contact technique using the electromagnetic (EM) stress sensor (Wang *et al.* 2005), and vibration-based methods. Due to the convenience and cost effectiveness of sensor and sensor installation, the vibration-based methods have been recognized to be efficient in practice (Kim and Park 2007). Cho *et al.* (2010a) implemented a vibration-based method proposed by Zui *et al.* (1996) on the smart sensors and experimentally verified in laboratory testing using a string with both ends fixed. Because Zui's formula utilizes the first three natural frequencies, applying this formula can be difficult if cable-deck interaction is dominant in the lower natural frequency region and thus automated peak-picking may not be able to reliably distinguish the cable modes from the deck modes. As a more practical alternative, a closed form relationship between natural frequencies and the tension force is selected for cable tension estimation in this study (see Appendix C for details).

CableTensionEstimation is an application that calculates cable tensions based-on the vibration-based method described in Appendix C. Because the estimation method requires the natural frequencies of each cable, information sharing between leaf nodes attached on different cables is not necessary. Thus, *IndependentProcessingPSD* in conjunction with peak-picking is appropriate to implement *CableTensionEstimation*. *IndependentProcessingPSD* performs unsynchronized sensing and estimate the power spectrum that can provide natural frequencies of cables using an automated peak-picking method; cable tensions can be obtained using the natural frequencies. The process of *CableTensionEstimation* is shown in Figure 3.4.

A simple peak-picking procedure is implemented in *CableTensionEstimation*: a peak is searched in a small frequency range around an approximate natural frequency initially provided to *CableTensionEstimation*. Two assumptions are made to justify the peak-picking method: (1) the natural frequencies of the cable are well separated and (2) the changes over time are small compared to difference between two consecutive natural frequencies. These assumptions will be discussed in the later chapter.

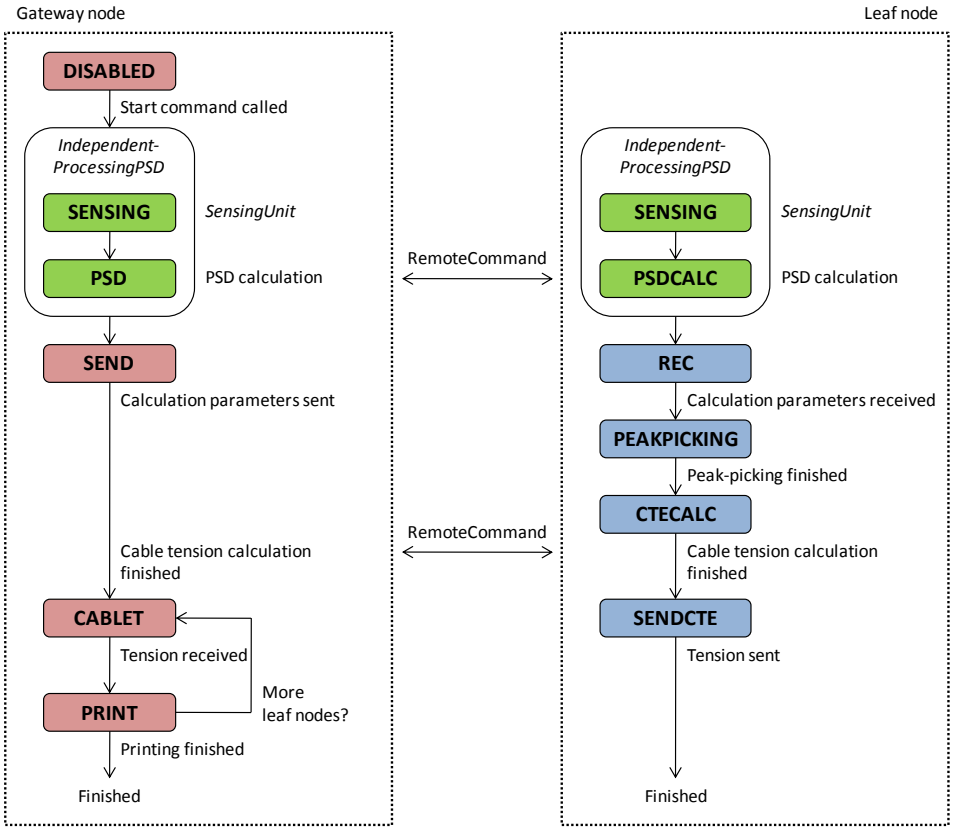


Figure 3.4 Flow of *CableTensionEstimation*.

3.3 Decentralized coordinated processing

As opposed to the independent processing approach presented in the previous section, coordinated processing allows the sensor nodes to communicate with each other and share information. As shown in Figure 2.2c, the sensor network is divided into local sensor communities where data communication and processing are taking place to extract meaningful information from raw sensor data. In contrast to the situation with independent processing strategies, coordinated processing allows estimation of spatial information that can be used to produce a global picture of a structural system (Sim *et al.* 2009). This section discusses design considerations of *DecentralizedDataAggregation* as an implementation of the coordinated processing approach.

DecentralizedDataAggregation is an application for data acquisition and processing implementing NExT and RDT in the hierarchical network shown in Figure 2.2c. *DecentralizedDataAggregation* outputs either correlation or RD function, depending on the user-specified input. The network is divided into local communities in which correlation or RD functions are calculated at each node and gathered by the cluster-heads as previously described. The correlation or RD functions can be either collected at the base station or retained at the cluster-heads for further analysis such as in-network system identification and damage detection. *DecentralizedDataAggregation* can be used as a foundation for other application development that requires the decentralized network such as decentralized modal analysis (Sim *et al.* 2010a; Sim *et al.* 2010b), decentralized

system identification (Sim *et al.* 2009), and decentralized damage detection (Jang *et al.* 2010b).

To appropriately realize the decentralized implementation of data processing based on NExT and RDT in the hierarchical network, the design of *DecentralizedDataAggregation* requires careful consideration of (1) network topology, (2) controlled network-wide flow, (3) fault tolerance, and (4) data processing methods.

3.3.1 Network topology

The cluster tree network topology adopted in *DecentralizedDataAggregation* employs three types of sensor nodes that can be categorized based on the role in the sensor network (see Figure 3.5): (1) gateway node, (2) cluster-head, and (3) leaf node. The gateway node is directly linked to the base station, controlling network-wide information and data flow. The gateway node disseminates information for sensing, data processing, and the network topology to the cluster-head and leaf nodes. The cluster-head coordinates with the leaf nodes to calculate correlation or RD functions in the local community using the measured data. Note that overlapping nodes are allowed so that phase information from two overlapping local communities can be related to each other.

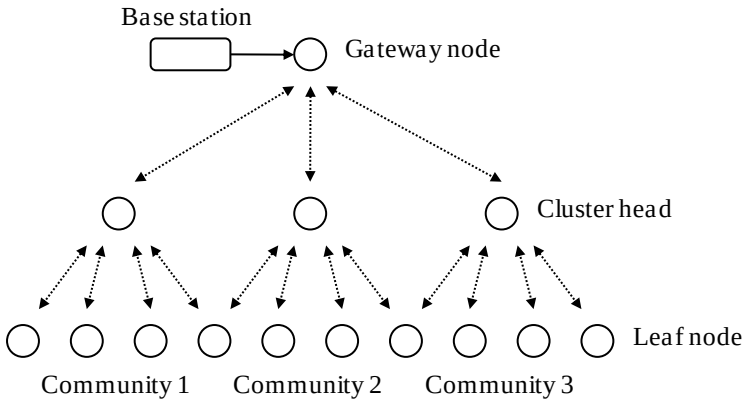


Figure 3.5 Network topology for *DecentralizedDataAggregation*.

3.3.2 Control flow

DecentralizedDataAggregation consists of four stages: (1) initialization, (2) synchronized sensing, (3) decentralized RD function estimation, and (4) data collection. Each stage is defined as follows.

(1) **Initialization**: Parameters for sensing (e.g., sampling rate, data length, and sensing channels) are distributed from the gateway node to all sensor nodes, including the cluster-head and leaf nodes. The network-wide time synchronization is then performed.

(2) **Synchronized sensing**: The gateway node disseminate the parameter specifying the sensing start time to the all sensor nodes in the network. Once the sensor nodes finish sensing, the measured data is subsequently resampled to produce the synchronized data.

(3) **Decentralized data processing**: Once the synchronized data is ready, *DecentralizedDataAggregation* decentrally process data in each local community. The cluster-head sends reference information (e.g., sensor data for NExT and trigger information for RDT) to the leaf nodes in the local community, and the correlation or RD function is calculated from the data at each node using the reference information. The processed data (e.g., correlation or RD function) is then sent to the cluster-head.

(4) **Data collection**: If the processed data is specified to be the output of *DecentralizedDataAggregation*, the gateway node collects the processed data from each local community, and subsequently transmits to the base station to save to a user-defined output file.

Completion of the process described herein results in the correlation or RD function estimated in the decentralized hierarchical network

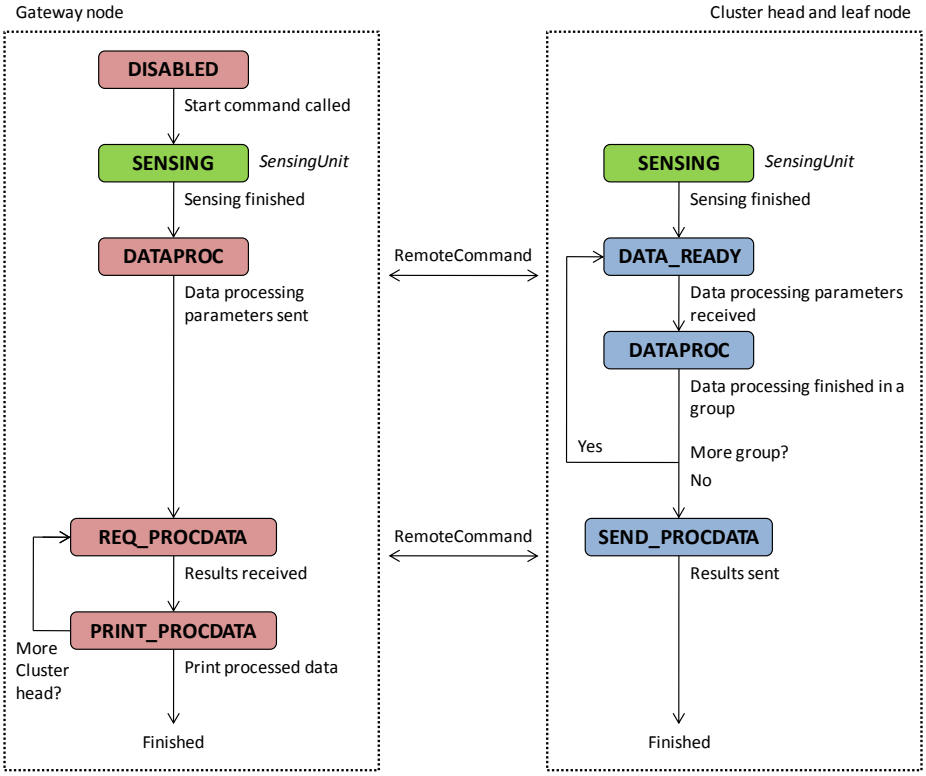


Figure 3.6 Flow of *DecentralizedDataAggregation.*

3.3.3 Fault tolerance

SHM applications for the wireless sensor network should be able to appropriately handle failures that unexpectedly take place during operation. Failures most often occur during wireless communication and sensing. In particular, when the distance between nodes is long or the battery is low, communication failure is more likely to happen (Linderman *et al.* 2010). For a SHM application to be employed in a wireless sensor network for monitoring of full-scale civil structures, this issue should be appropriately addressed so that the application proceeds even when unexpected failures occur.

In designing *DecentralizedDataAggregation*, fault-tolerance features are considered to improve reliability for full-scale implementations. The objective of the fault-tolerance is to ensure *DecentralizedDataAggregation* will continue with the working sensor nodes. Because the gateway node, which controls the flow of *DecentralizedDataAggregation*, should remain working throughout the operation, the gateway node removes failed nodes from the network. The cluster-head can remove its leaf nodes in case of failures, and its local sensor community is disregarded if the cluster-head fails. Leaf nodes can be removed from the network by the gateway node or cluster-heads if any failure occurs. This process is illustrated in Figure 3.7. In summary, failures are primarily handled in the node that resides in higher levels in the network hierarchy, reconfiguring the network to remove failed nodes in lower levels.

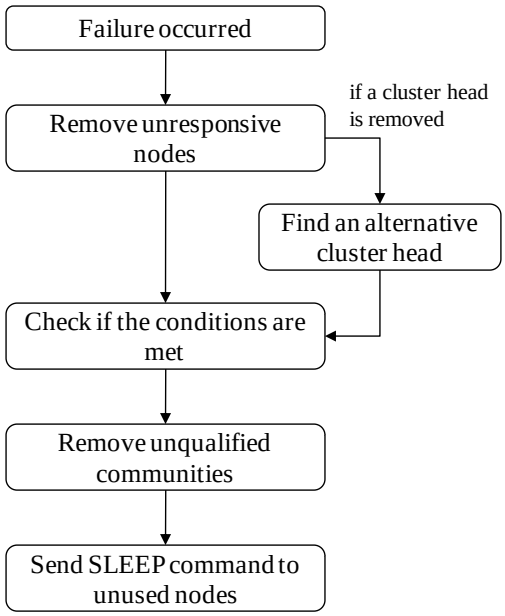


Figure 3.7 Flowchart for fault tolerance implementation.

3.3.4 Data processing methods

DecentralizedDataAggregation supports two data processing methods, Natural Excitation Technique (NExT) and Random Decrement Technique (RDT). Required to specify before running *DecentralizedDataAggregation*, the selected method decentrally processes sensor data to output correlation or RD function as well as the length of reference signals in the case of RDT. These two processing methods are implemented as two separate services that have the identical output data type and similar interfaces for *DecentralizedDataAggregation* so that NExT and RDT can be interchangeably used. The RDT-based data aggregation is more efficient in wireless communication than the NExT-based approach; the efficacy of RDT will be described in more detail in Chapter 5.

3.4 Summary

Implementations of the decentralized processing approaches on the WSSN consisting of Imote2 smart sensors were described in this chapter. *SensingUnit* is a basic software

component that performs network-wide synchronized or synchronized sensing. *SensingUnit* can be commonly used in developing SHM applications. Two applications of the independent processing scheme were described: (a) *IndenpendentProcessingPSD* uses *SensingUnit* to measure data, of which power spectrum is calculated and (b) *CableTensionEstimation* estimates tension forces of cables using power spectrum obtained from *IndependentProcessingPSD*. As an example for decentralized coordinated processing, *DecentralizedDataAggregation* collects correlation functions based on NExT and RDT. The next chapter develops decentralized modal analysis tailored to the WSSN based on *DecentralizedDataAggregation*.

DECENTRALIZED MODAL ANALYSIS

The WSSN ultimately requires decentralized local processing utilizing smart sensor's onboard computing capability to efficiently use limited resources such as available bandwidth and battery. The processed data in the decentralized hierarchical network shown in Figure 2.2c contains only local information of each local sensor community. Decentralized modal analysis provides a means to obtain a global picture of a structure by combining the local information. This chapter presents formulation and numerical validation of decentralized modal analysis.

4.1 Formulation

An automated, hierarchical decentralized approach for modal analysis using smart sensors is proposed to obtain the global modal properties using a decentralized network topology. It consists of two parts: (i) local feature extraction and (ii) determination of global modal properties. In the WSSN, local features are estimated independently in each local sensor community and subsequently collected at the base station, where the global modal properties are determined. This section describes this process.

4.1.1 Local feature extraction

Consider the structure and sensor network topology depicted in Figure 4.1. The structure is divided into overlapping subdomains, represented by Ω_i (i=1, ···,n). Data aggregation and processing are conducted independently within each subdomain. In this study, two cases regarding the input excitation are considered. In the first, the input excitation is assumed to be measurable, allowing the transfer function to be estimated. By taking the inverse FFT of the transfer function, the impulse response function can be obtained, and subsequently used as input to ERA. The second case assumes that the input excitation is unavailable, and NExT is employed. In this approach, the cross-correlation functions between the measured responses are used as the input to ERA. In both cases, only the identified local information is collected centrally for determination of the global modal properties. More information regarding implementation of these two approaches on smart sensor networks can be found in Nagayama and Spencer (2007).

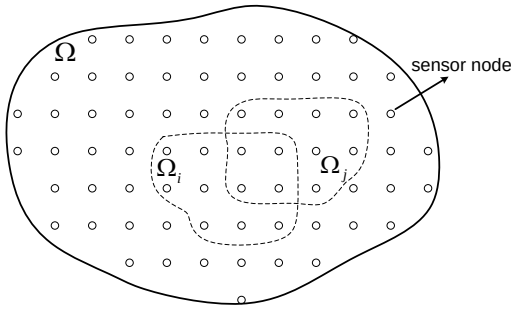


Figure 4.1 Structure and overlapping subdomains (*i, j* = 1~n).

4.1.2 Determination of global modal properties

Once the local information is collected centrally, the first task is to delineate the true modes from the noise modes. In this study, the true modes are selected based on the number of identified natural frequencies from the subdomains (Zimmerman *et al.* 2008). The true modes should be identified in the majority subdomains, while the noise modes will randomly appear in the subdomains. Thus, if a specific natural frequency is identified in a substantial number of the subdomains, it is considered as a true mode. If ERA fails to find the true mode in certain subdomains, the cross spectrum is alternatively used to estimate the local mode shapes. Once the true modes are determined, the corresponding mode shapes can be combined together; the remainder of this section describes this process.

Consider global mode shape ϕ_{Ω}^{m} for the m^{th} mode, along with the previously determined local mode shapes $\phi_{\Omega 1}^{m}$, $\phi_{\Omega 2}^{m}$, ⋯, $\phi_{\Omega 3}^{m}$ associated with respective subdomains. The local mode shapes $\phi_{\Omega i}^{m}$ and $\phi_{\Omega k}^{m}$ associated with two neighboring subdomains can be expressed as:

$$\phi_{\Omega_i}^{m} = \begin{pmatrix} 1 \\ \phi_{i,2} \\ \vdots \\ \phi_{i,p} \\ \phi_{i,1}^{o} \\ \vdots \\ \phi_{i,r}^{o} \end{pmatrix} \quad \text{and} \quad \phi_{\Omega_k}^{m} = \begin{pmatrix} 1 \\ \phi_{k,2} \\ \vdots \\ \phi_{k,q} \\ \phi_{k,1}^{o} \\ \vdots \\ \phi_{k,r}^{o} \end{pmatrix} \tag{4.1}$$

where the superscript *o* denotes the overlapping node in the i^{th} and k^{th} subdomains, r is the number of the overlapping nodes, and p and q are the number of non-overlapping nodes in the i^{th} and k^{th} subdomains, respectively. To allow assembly, the mode shapes in Equation (4.1) should be rescaled to have the same values at the overlapping nodes, i.e.,

$$R_i \begin{pmatrix} \phi_{i,1}^{o} \\ \vdots \\ \phi_{i,r}^{o} \end{pmatrix} = R_k \begin{pmatrix} \phi_{k,1}^{o} \\ \vdots \\ \phi_{k,r}^{o} \end{pmatrix} \tag{4.2}$$

where R_i and R_k are the normalization factors for the mode shapes $\phi_{\Omega i}^{m}$ and $\phi_{\Omega k}^{m}$, respectively. The global mode shape is the union of the local mode shapes as:

$$\phi_{\Omega}^{m} = \bigcup_{i=1}^{n} R_i \phi_{\Omega_i}^{m} \tag{4.3}$$

In the presence of noise, the solution to Equation (4.2) for any $r>1$ does not exist in general. Therefore, the normalization factor R_i (i=1, 2, ⋯, n) must be approximately determined, for example as a solution in the least-square sense. Because the subdomains are interconnected, Equation (4.2) can be expanded up to $n(n\text{-}1)/2$ equations for all pairs of the overlapping local mode shapes as follows.

$$
\begin{aligned}
\phi^m_{\Omega_1\Omega_2} &= R_2\phi^m_{\Omega_2\Omega_1} + \varepsilon_{12} \\
\phi^m_{\Omega_1\Omega_3} &= R_2\phi^m_{\Omega_3\Omega_1} + \varepsilon_{13} \\
&\vdots \\
\phi^m_{\Omega_1\Omega_n} &= R_2\phi^m_{\Omega_n\Omega_1} + \varepsilon_{1n} \\
R_2\phi^m_{\Omega_2\Omega_3} &= R_3\phi^m_{\Omega_2\Omega_3} + \varepsilon_{23} \\
&\vdots \\
R_2\phi^m_{\Omega_2\Omega_n} &= R_3\phi^m_{\Omega_n\Omega_2} + \varepsilon_{2n} \\
&\vdots \\
R_{n-1}\phi^m_{\Omega_{n-1}\Omega_n} &= R_n\phi^m_{\Omega_n\Omega_{n-1}} + \varepsilon_{(n-1)n}
\end{aligned}
\tag{4.4}
$$

where $\phi^m_{\Omega i\Omega k}$ is the m^{th} local mode shape in the i^{th} domain at the nodes that overlap the k^{th} domain, and ε_{ik} is the error between the mode shapes. Note that the normalization factor for the first subdomain R_1 is selected to be 1. In matrix form, Equation (4.4) becomes

$$
\mathbf{y} = \mathbf{X}\mathbf{R} + \boldsymbol{\varepsilon} \tag{4.5}
$$

where

$$
\mathbf{y} = \begin{pmatrix} \phi^m_{\Omega_1\Omega_2} \\ \phi^m_{\Omega_1\Omega_3} \\ \vdots \\ \phi^m_{\Omega_1\Omega_n} \\ 0 \\ \\ \vdots \\ \\ 0 \end{pmatrix},\quad
\mathbf{X} = \begin{pmatrix}
\phi^m_{\Omega_2\Omega_1} & 0 & \cdots & 0 \\
0 & \phi^m_{\Omega_3\Omega_1} & \cdots & 0 \\
\vdots & \vdots & \ddots & \vdots \\
0 & 0 & \cdots & \phi^m_{\Omega_n\Omega_1} \\
-\phi^m_{\Omega_2\Omega_3} & \phi^m_{\Omega_3\Omega_2} & \cdots & 0 \\
\vdots & \vdots & \ddots & \vdots \\
-\phi^m_{\Omega_2\Omega_n} & 0 & \cdots & \phi^m_{\Omega_n\Omega_2} \\
\vdots & \vdots & \vdots & \vdots \\
-\phi^m_{\Omega_{n-1}\Omega_n} & 0 & \cdots & \phi^m_{\Omega_n\Omega_{n-1}}
\end{pmatrix},\quad
\mathbf{R} = \begin{pmatrix} R_2 \\ R_3 \\ \vdots \\ R_4 \end{pmatrix},\quad
\boldsymbol{\varepsilon} = \begin{pmatrix} \varepsilon_{12} \\ \varepsilon_{13} \\ \vdots \\ \varepsilon_{1n} \\ \varepsilon_{23} \\ \vdots \\ \varepsilon_{2n} \\ \vdots \\ \varepsilon_{(n-1)n} \end{pmatrix}
\tag{4.6}
$$

The estimator $\hat{\mathbf{R}}$ that minimizes the square of errors $\boldsymbol{\varepsilon}^T\boldsymbol{\varepsilon}$ is given by (see Montgomery and Runger 1994)

$$
\hat{\mathbf{R}} = \left(\mathbf{X}^T\mathbf{X}\right)^{-1}\mathbf{X}^T\mathbf{y} \tag{4.7}
$$

Using the normalization factor $\mathbf{R}$, the local mode shapes are scaled and assembled to obtain the global mode shape. At the overlapping nodes, the local mode shapes are averaged to obtain the associated values of the global mode shape.

The accuracy of the combined global mode shapes can be evaluated by the error between the combined and reference global mode shapes, ϕ_{Ω}^{m} and $\phi_{\Omega,ref}^{m}$, respectively:

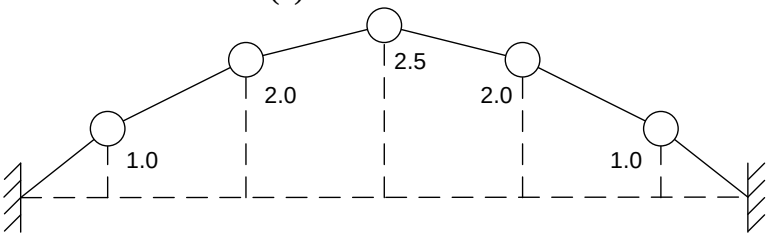
$$e_m = \phi_{\Omega}^{m} - \phi_{\Omega,ref}^{m} \tag{4.8}$$

The reference global mode shapes are estimated using all measured accelerations simultaneously. Note that the reference global mode shapes correspond to the centralized data acquisition and processing scheme commonly used in the wired sensor networks, while the proposed decentralized approach is for the WSSN. For the error e_m to be meaningful, the two global mode shapes should be appropriately normalized. Thus, $\phi_{\Omega,ref}^{m}$ is normalized so that the largest element in $\phi_{\Omega,ref}^{m}$ is equal to 1, and ϕ_{Ω}^{m} is then scaled to minimize the square error between ϕ_{Ω}^{m} and $\phi_{\Omega,ref}^{m}$:

$$\min\left(\phi_{\Omega}^{m} - \phi_{\Omega,ref}^{m}\right)^{T}\left(\phi_{\Omega}^{m} - \phi_{\Omega,ref}^{m}\right) \tag{4.9}$$

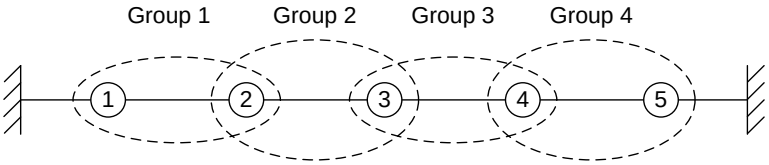
To better understand the proposed method employing the least squares approximation, consider the 5 DOF spring-mass model in Figure 4.2a and its first mode shape in Figure 4.2b. This model is divided into subgroups forming two different topologies as shown in Figure 4.3. Note that subgroups share one overlapping node in Topology 1 and two in Topology 2.

(a) 4 DOF model.

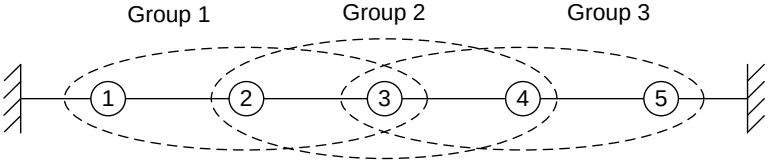


(b) Exact global mode shape (1st mode).

Figure 4.2 4 DOF model and mode shape.



(a) Topology 1: 1 overlapping node



(b) Topology 2: 2 overlapping nodes

Figure 4.3 Topologies with different numbers of overlapping nodes.

Table 4.1 summarizes the global and local mode shapes, normalized with respect to the first node in each mode shape and with 10% error. In Topology 2, the normalization factors calculated using Equations (4.7) are R_2=2.0694 and R_3=2.7388. The scaled local mode shapes are

$$\phi_{\Omega_1} = \begin{pmatrix} 1.0000 \\ 2.2000 \\ 2.7500 \end{pmatrix} \quad R_2\phi_{\Omega_2} = \begin{pmatrix} 2.0694 \\ 2.8454 \\ 2.2763 \end{pmatrix} \quad R_3\phi_{\Omega_3} = \begin{pmatrix} 2.7388 \\ 2.4102 \\ 1.2051 \end{pmatrix} \tag{4.10}$$

Taking the average values at the overlapping nodes, the global mode shape is assembled as

$$\phi_{\Omega} = \begin{pmatrix} 1.0000 & 2.1347 & 2.7781 & 2.3433 & 1.2051 \end{pmatrix}^T \tag{4.11}$$

where T denotes the matrix transpose. The same procedure can be applied to Topology 1, resulting in the global mode shape as follows.

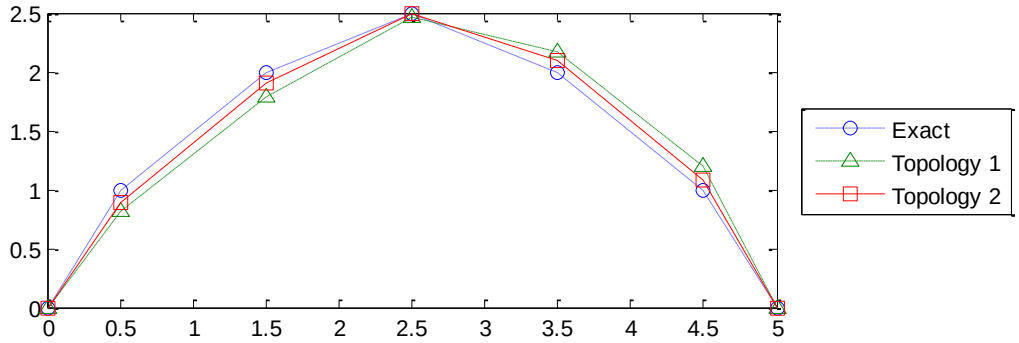
$$\phi_{\Omega} = \begin{pmatrix} 1.0000 & 2.2000 & 3.0250 & 2.6620 & 1.4641 \end{pmatrix}^T \tag{4.12}$$

Note that averaging is not required to assemble the scaled local mode shapes in Topology 1, because the subgroups share only one node.

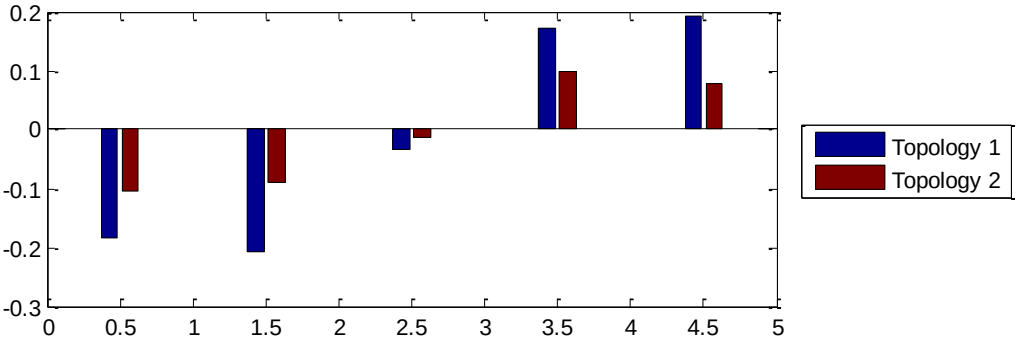
Figure 4.4 compares the exact global mode shape and the combined global mode shapes of topologies 1 and 2. After normalizing the combined and reference global mode shapes, which is the exact in this example, the error e_1 is calculated. From Figure 4.4, the combined global mode shape of Topology 2 is seen more accurate than that of Topology 1. In the subsequent sections, numerical examples are provided to investigate the efficacy of the proposed method in detail, mainly focused on the sensor topologies.

Table 4.1 Global and local mode shapes.

Mode shape			Mass				
			1	2	3	4	5
Exact global mode shape			1.0	2.0	2.5	2.0	1.0
Local mode shape w/ 10% error	Topology 1	Group 1	1.0	2.2	-	-	-
		Group 2	-	1.0	1.375	-	-
		Group 3	-	-	1.0	0.88	-
		Group 4	-	-	-	1.0	0.55
	Topology 2	Group 1	1.0	2.2	2.75	-	-
		Group 2	-	1.0	1.375	1.1	-
		Group 3	-	-	1.0	0.88	0.44



(a) Assembled mode shapes.



(b) Errors of assembled mode shapes.

Figure 4.4 Comparison of the global mode shapes.

Numerical examples of a plate and planar truss are provided to validate the proposed approach. In these examples, the effects of the sensor topologies on the accuracy of the combined global mode shapes are investigated.

4.2 Examples

4.2.1 Plate model

Consider the uniform plate shown in Figure 4.5. The top and left edges of the plate are considered as fixed, and the bottom and right edges are simply supported. The plate is assumed to be 0.7 m by 1 m and made of steel having an elastic modulus of 200 GPa, a Poisson ratio of 0.3, mass density of 7.83×10^3 kg/m^3, and thickness of 1 mm. The numerical model of the plate is implemented in MATLAB using the 12 DOF (3DOF at each node) rectangular Kirchhoff plate element known as the ACM element (Zienkiewicz and Cheung 1964). The shape functions are selected to be incomplete, 4th-order polynomials in the *x*- and *y*-directions resulting in 12 terms. The first seven out-of-plane mode shapes of the plate model are presented in Figure 4.6, and the transfer functions between the input excitation applied vertically at the nodes marked by an "X" and the accelerations at nodes N1 and N2 (see Figure 4.5) are shown in Figure 4.7.

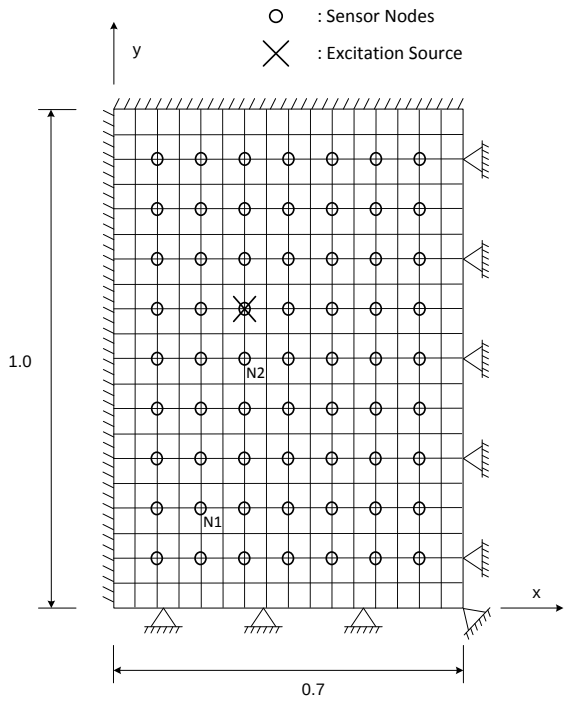


Figure 4.5 Plate model.

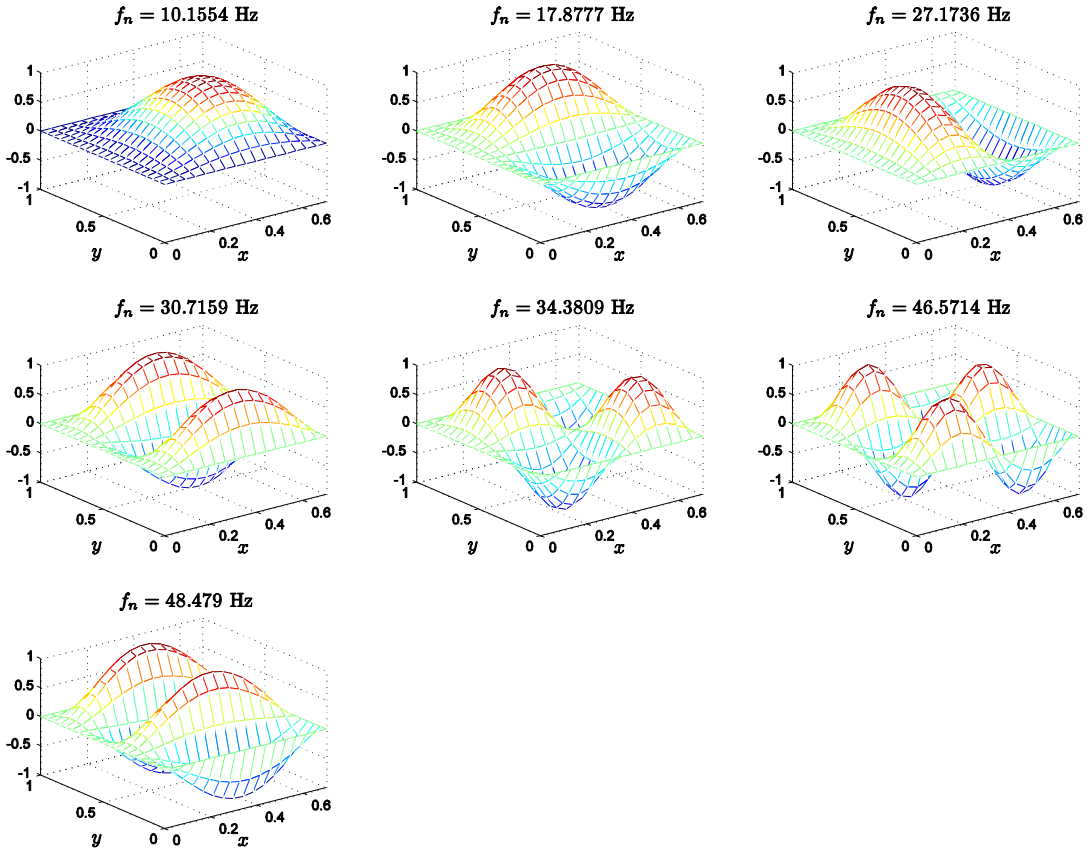
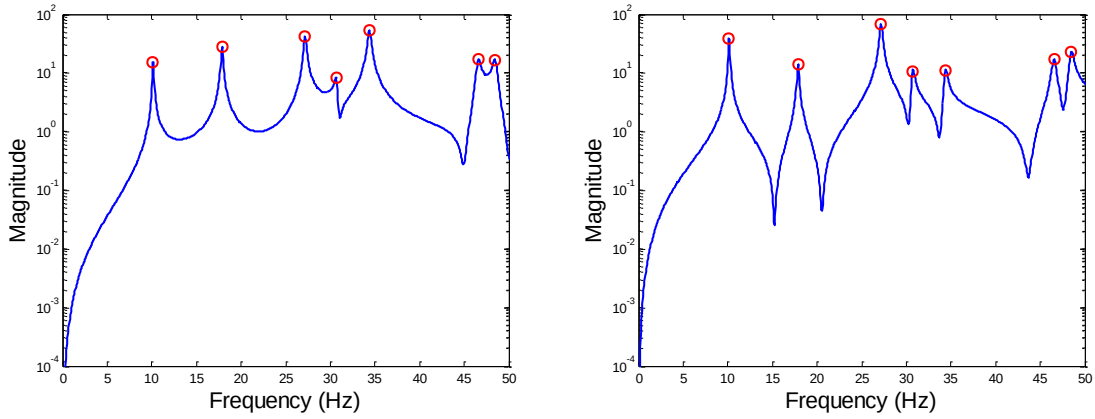


Figure 4.6 Global mode shapes from the Finite Element model.



(a) N1. (b) N2.

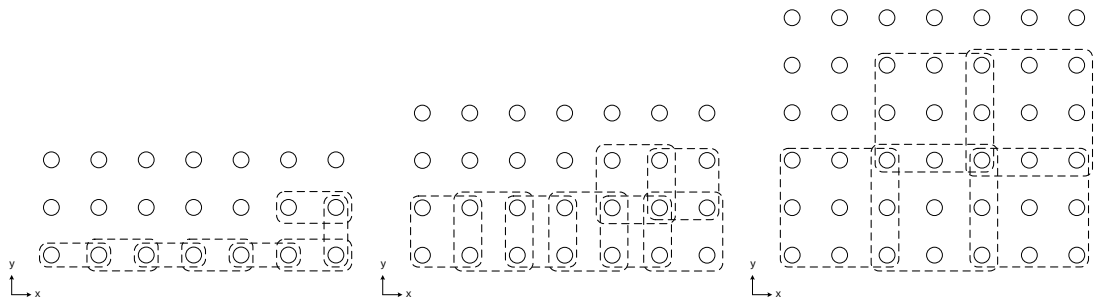
Figure 4.7 Transfer functions (Circles represent natural frequencies).

A total of 63 evenly spaced sensor nodes are selected as shown in Figure 4.5 to obtain vertical accelerations under the input excitation. As summarized in Table 4.2, four different simulation cases are considered: an impulse loading (cases 1 and 2) or a band-limited white noise up to 50 Hz (cases 3 and 4). 5% RMS noise is added to all measurements. ERA is used to identify the mode shapes and frequencies in cases 1, 2, and 3, whereas NExT/ERA is used in case 4. As the input excitation to large-scale civil engineering structures is not generally available, output-only modal analysis for random excitations is often employed; thus, case 4 may be considered as the most important for structural health monitoring.

Table 4.2 Simulation cases.

Excitation type	Input measured	Input not measured
Impulse	Case 1 (ERA)	Case 2 (ERA)
Band-limited white noise	Case 3 (ERA)	Case 4 (NExT/ERA)

To investigate the effect of the sensor topologies on the combined global mode shapes, the nodes are grouped in four different ways, as shown in Figure 4.8: (a) Topology 1: two nodes in each group with one overlapping node, (b) Topology 2: four nodes in each group with two overlapping nodes, and (c) Topology 3: nine nodes in each group with three overlapping nodes. Note that Topology 1 corresponds to the approach adopted by Zimmerman *et al.* (2008).



(a) Topology 1: 2 nodes in each group. (b) Topology 2: 4 nodes in each group. (c) Topology 3: 9 nodes in each group.

Figure 4.8 Schematic view of subdomains (dashed line).

From the data communication perspective, having fewer local groups is advantageous due to less amount of data being transmitted over the radio. In a local group, the reference sensor sends sensor data that is generally long and all other sensors transmits the condensed data such as the correlation function of the cross spectrum. Thus, the amount of transmitted data can be drastically reduced by adopting a topology with less number of local groups. Assuming the time history record of length N and n_d times averaging without any overlap between spectral windows, the total transmitted data (Nagayama and Spencer 2007) is at most

$$N_{data} = n_g \left(N \cdot n_d + \frac{N}{2} \cdot (n_s - 1) \right) \tag{4.13}$$

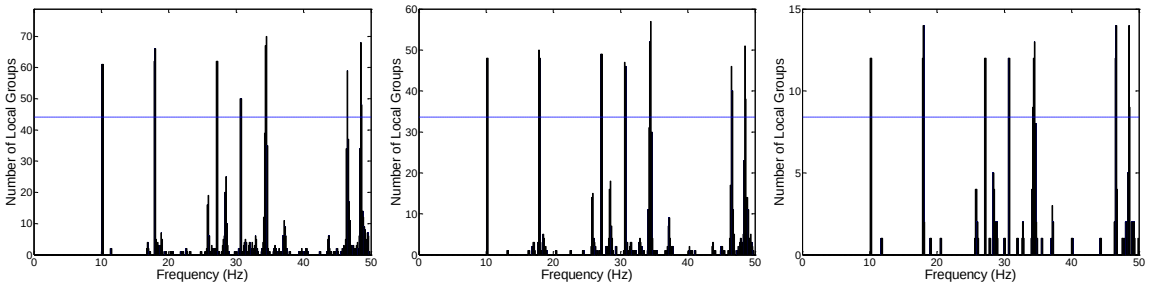
where n_g and n_s are the numbers of local groups and sensor nodes, respectively. From Equation (4.13) with $N = 1024$, N_{data} are about 1.30×10^6, 1.06×10^6, and 0.29×10^6 for topology 1, 2, and 3, respectively. Thus, Topology 3 has the least data communication requirement.

As previously described, the number of groups in which a natural frequency is locally identified is utilized to delineate between true modes and noise modes. The natural frequencies estimated in each group are collected, and if the number of collected natural frequencies in a specific frequency range is greater than a predetermined threshold value, it is assumed to be a true mode. The threshold is selected to be 70% of the number of local groups. The frequency range in which identified frequencies are counted is $(f_c - \Delta f \quad f_c + \Delta f)$ where f_c is the central frequency of the range and $2\Delta f$ is the width. In this study, f_c and Δf are selected as follows.

$$\begin{aligned} f_c &= k \cdot \frac{f_s}{N_{FFT}} \quad k = 1, 2, \cdots \\ \Delta f &= 2 \cdot \frac{f_s}{N_{FFT}} \end{aligned} \tag{4.14}$$

where f_s is the sampling rate, N_{FFT} is the number of FFT points, and k is any positive integer such that f_c is less than the bandwidth of the input excitation. Note that the adjacent ranges are made to overlap with each other to prevent the case that the identified frequencies are evenly distributed over two adjacent non-overlapping ranges.

Figure 4.9 shows the typical number of groups in which a natural frequency is found in case 4 (random excitation, input not measured). The identified frequencies in the local groups are concentrated at several frequencies such as about 10 Hz, 18 Hz, 27 Hz, 31 Hz, 34 Hz, 47 Hz, and 49 Hz that are finally considered as corresponding to true modes. In some frequency ranges, the number of local groups is greater than the total number of local groups due to noise modes that are closely located to the true modes. In this case, the frequency that is closest to f_c is considered as the natural frequency of the local group.

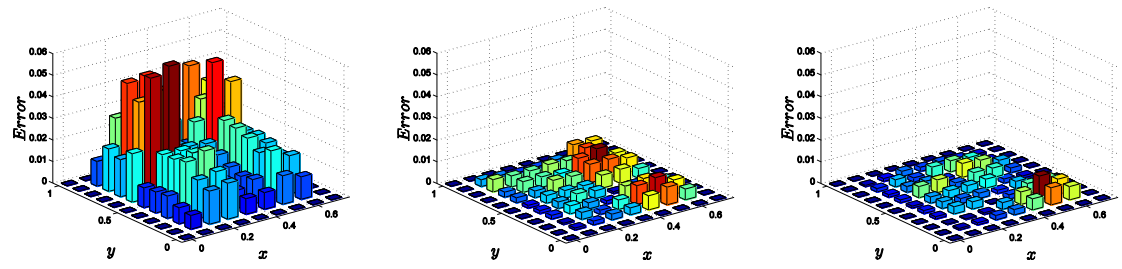


(a) Topology 1 (b) Topology 2 (c) Topology 3

Figure 4.9 Number of identified natural frequencies (case 4: random excitation and input not measured).

The global mode shapes can be assembled with the local mode shapes using the proposed method previously described. If a true mode is not identified in a certain local group, cross spectra of the accelerations in the group are alternatively utilized to estimate the local mode shapes. The cross spectrum values at the spectral line nearest to the natural frequency of the majority groups are assumed as the mode shape. Note that mode shapes obtained only from the cross spectra may not be as accurate as those from the ERA because the cross spectra have values only at the spectral lines.

The error e_k defined in Equation (4.8) is calculated to assess the accuracy of the combined global mode shape. Typical plots of the error e_2 for each topology in case 4 are shown in Figure 4.10. Improvement in accuracy is clearly shown for Topologies 2 and 3, where a larger size of local groups and more overlapping nodes are employed.



(a) Topology 1. (b) Topology 2. (c) Topology 3.

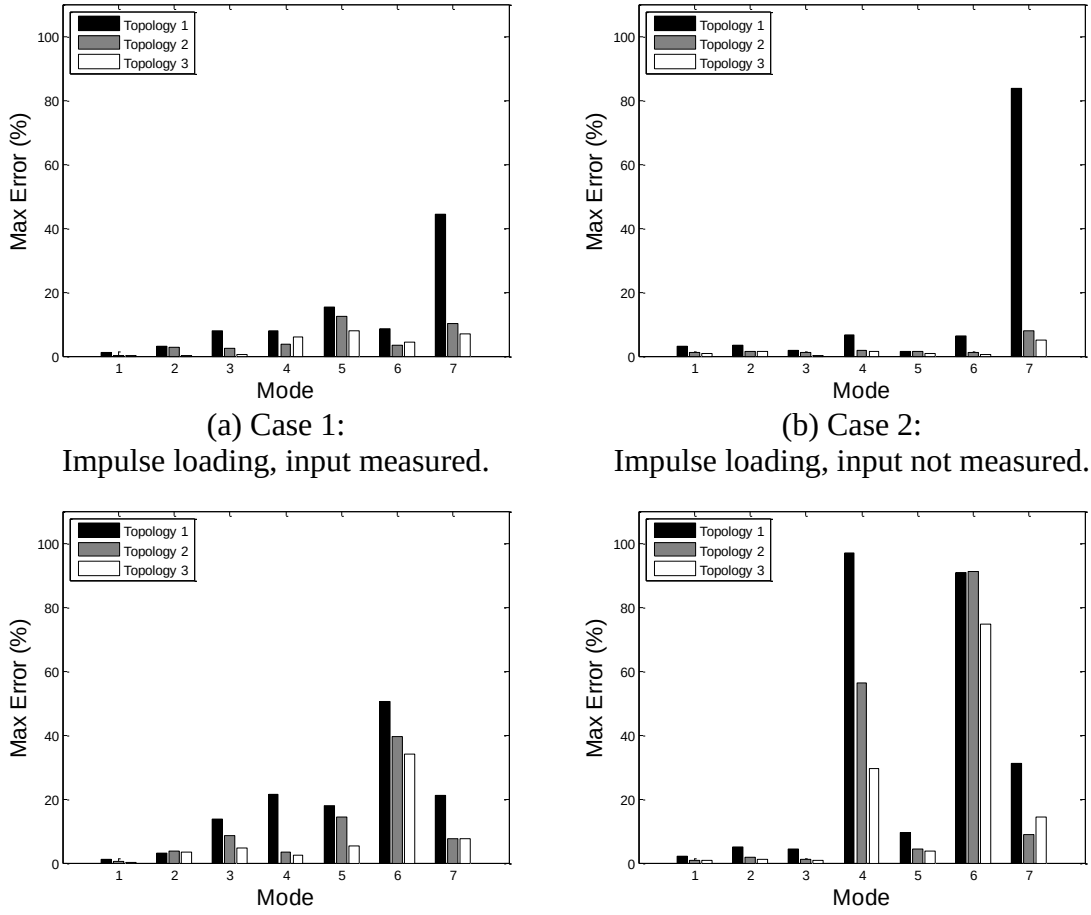
Figure 4.10 Absolute value of the errors between the combined and reference global mode shapes (case 4, 2nd mode).

Two error measures defined previously in Equations (4.15) and (4.16) are considered to quantitatively evaluate the accuracy of the combined global mode shapes.

$$E_{max,k} = \max\left(\left|e_{k,1}\right|, \left|e_{k,2}\right|, \cdots, \left|e_{k,n}\right|\right) \tag{4.15}$$

$$E_{avg,k} = \text{mean}\left(\left|e_{k,1}\right|, \left|e_{k,2}\right|, \cdots, \left|e_{k,n}\right|\right) \tag{4.16}$$

Repeating the simulation 100 times for each case, the averages of each error measure are calculated for the first seven modes as shown in Figure 4.11 and Figure 4.12. Zoomed figures from 0% to 15% are also provided in Figure 4.12 to clearly show $E_{avg,k}$. In Figure 4.11 and Figure 4.12, topology 3 consistently has the smallest errors in most cases. It can be concluded that sufficiently large local groups and multiple overlapping nodes as in Topology 3 contribute to reliable and accurate estimation of global modal properties.



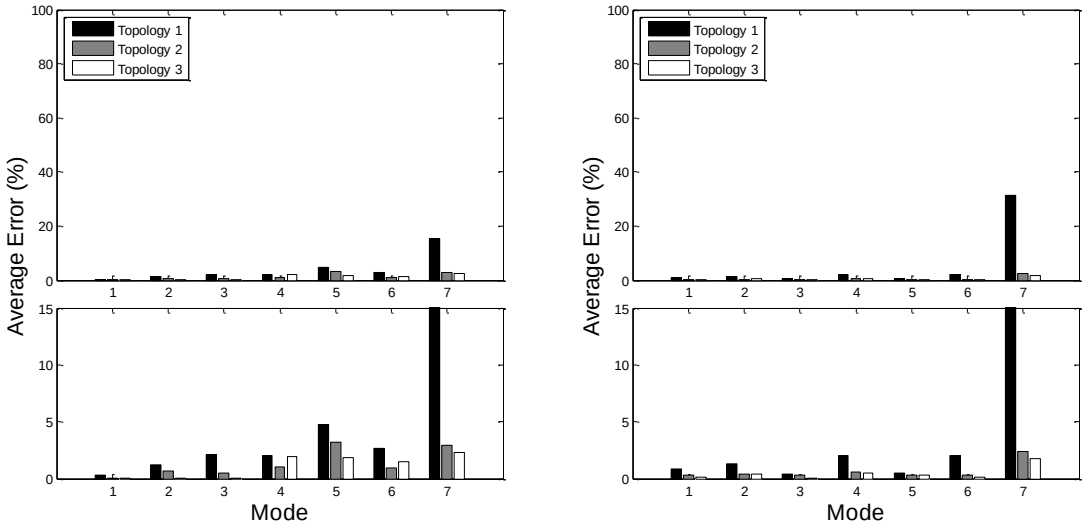
(a) Case 1: Impulse loading, input measured.

(b) Case 2: Impulse loading, input not measured.

(c) Case 3: Random excitation, input measured.

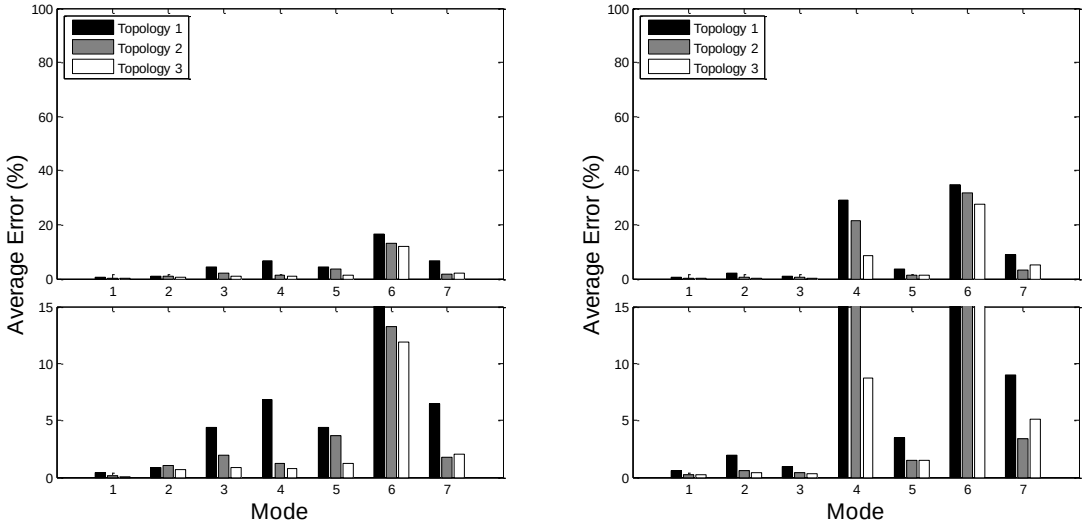
(d) Case 4: Random excitation, input not measured.

Figure 4.11 Maximum error.



(a) Case 1:
Impulse loading, input measured.

(b) Case 2:
Impulse loading, input not measured.



(c) Case 3:
Random excitation, input measured.

(d) Case 4:
Random excitation, input not measured.

Figure 4.12 Average error.

4.2.2 Truss model

Consider the three-dimensional truss model shown in Figure 4.13. This simply supported truss consists of 53 elements that have the identical sectional and material properties shown in Table 4.3. The input excitation, either impulse loading or random excitation, is applied vertically as shown in Figure 4.13. Transfer functions between the excitation and the accelerations at nodes N1 and N2 marked by circles in Figure 4.13 are shown in Figure 4.14. As in the plate example, four cases in Table 4.2 are considered. For cases 3 and 4, a band-limited white noise of which bandwidth is from 0 to 120 Hz that

encompasses the first four modes of the truss is applied. Vertical accelerations at all lower nodes are obtained in each case. 5% RMS noise is added to all measurements.

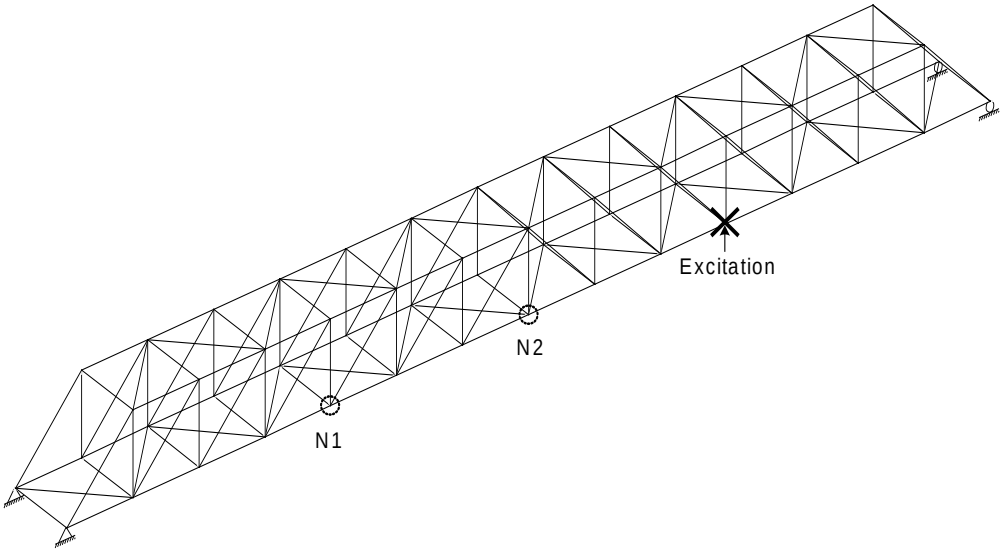
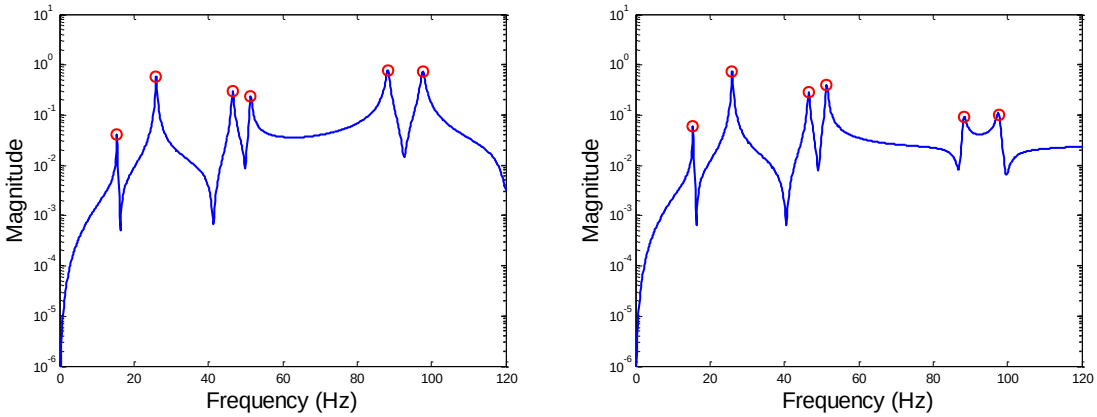


Figure 4.13 Truss model.

Table 4.3 Sectional and material properties.

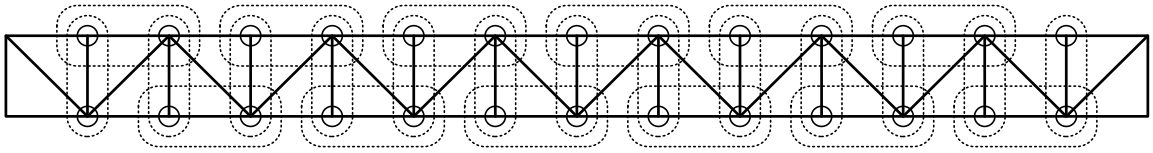
Cross sectional area (m^2)	Moment of inertia (m^4)			Elastic modulus (Pa)	Shear modulus (Pa)	Mass density (kg/m^3)
	I_x	I_y	I_z			
3.0×10^{-4}	5.0×10^{-9}	2.5×10^{-9}	2.5×10^{-9}	1.999×10^{11}	7.692×10^{11}	7.827×10^{3}



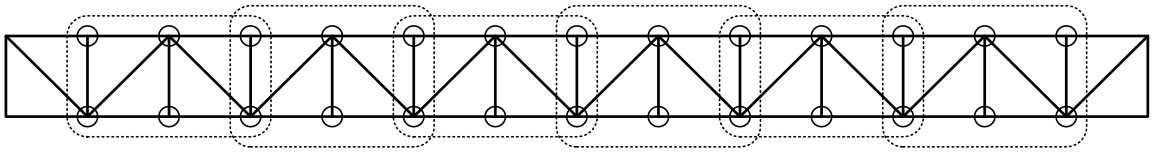
(a) Node N1. (b) Node N2.

Figure 4.14 Transfer functions (Circles represent natural frequencies).

Two types of sensor topologies in Figure 4.15 are considered: (a) for Topology 1 each local group consist of two nodes, one of which is the overlapping node and (b) for Topology 2 each local group has four nodes, two of which are the overlapping nodes.



(a) Topology 1: 1 overlapping node.



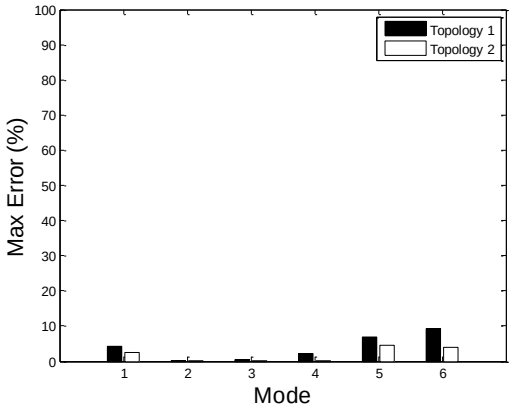
(b) Topology 2: 2 overlapping nodes.

Figure 4.15 Plan view of the truss model for sensor topologies.

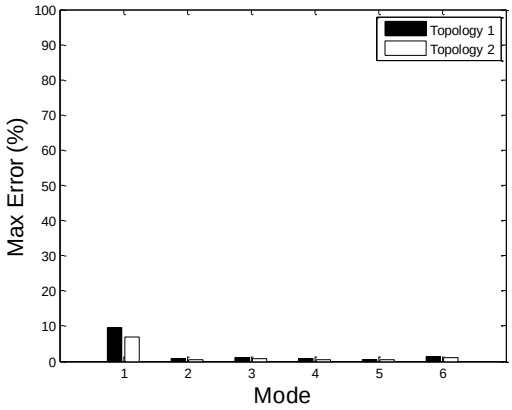
The proposed method is applied to determine the global modal properties of the truss. Modal analysis is conducted in each group using ERA (cases 1, 2, and 3) or NExT/ERA (case 4) to estimate local modal properties, and the true modes are obtained based on the locally identified natural frequencies. However, the 1st mode that does not have significant energy as can be seen in Figure 4.14 is not determined as a true mode in case 4. This mode can be manually identified if correlation function is collected to the base station.

As in the plate example, the maximum and average errors between the reference and combined global mode shapes are selected to evaluate the accuracy. The simulation is repeated 100 times, and the averages of the errors are obtained for the first 7 modes as shown in Figure 4.16 and Figure 4.17. In these graphs, topology 2 (larger local groups and multiple overlapping nodes) exhibits consistently smaller error than that of topology 1. Particularly in case 4, which is considered as the most important for the civil structures due to the difficulty in measuring the input excitation, the difference between the errors for topologies 1 and 2 is clearly seen in Figure 4.16d and Figure 4.17d.

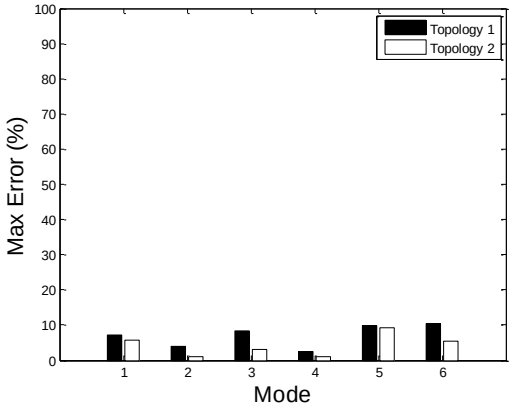
To visualize the difference, typical combined global mode shapes of the 3rd, 6th, and 7th modes for topologies 1 and 2 are compared with the reference global mode shapes in Figure 4.18, Figure 4.19, and Figure 4.20. Here, the mode shape at upper nodes is assumed having the same values with the corresponding lower nodes to fully draw mode shapes of the truss. The discrepancy between the reference and the combined global mode shape of topology 1 is clear; topology 2 shows better agreement with the reference.



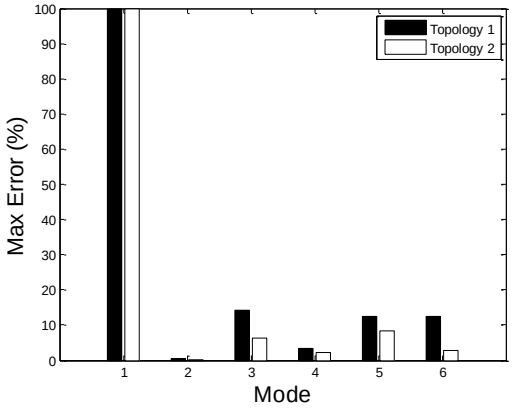
(a) Case 1:
Impulse loading, input measured.



(b) Case 2:
Impulse loading, input not measured.

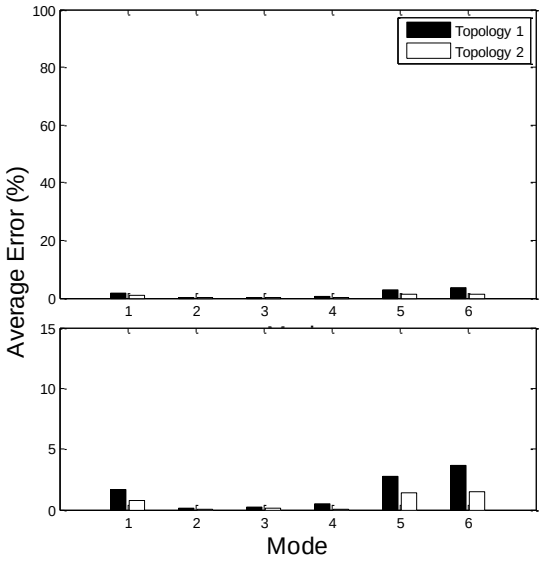


(c) Case 3:
Random excitation, input measured.

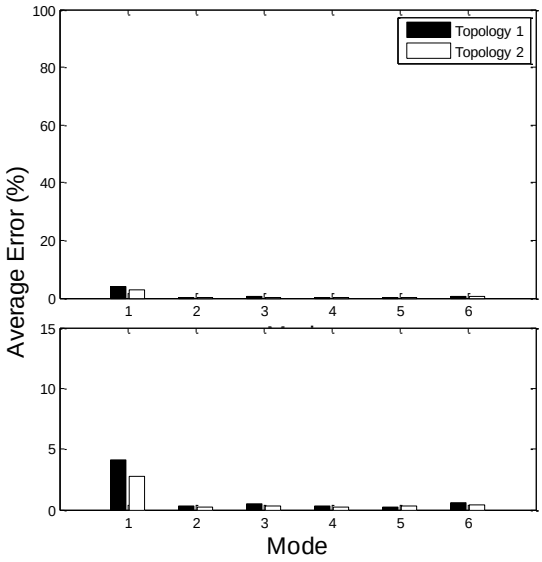


(d) Case 4:
Random excitation, input not measured.

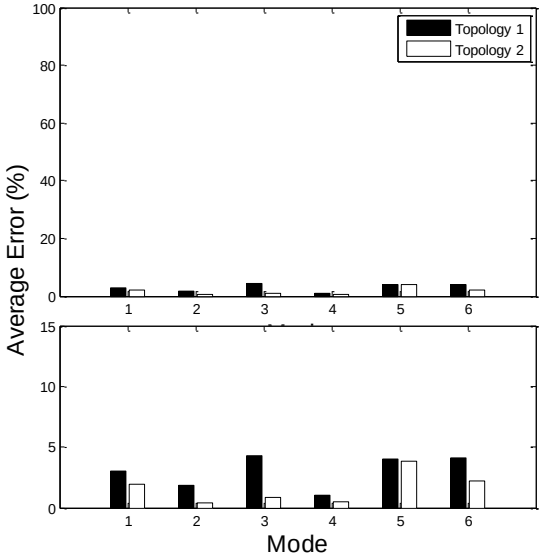
Figure 4.16 Maximum error.



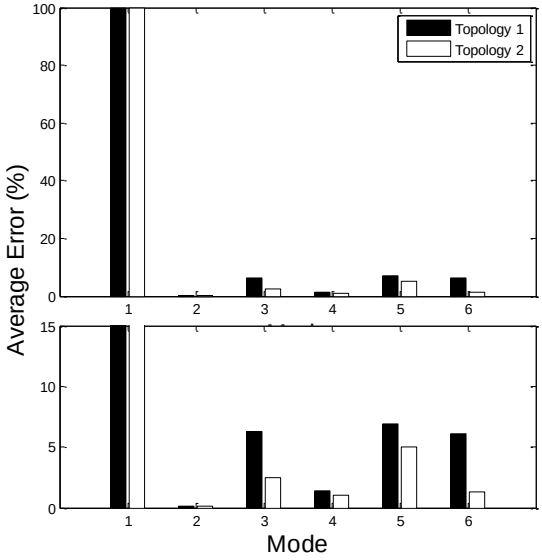
(a) Case 1:
Impulse loading, input measured.



(b) Case 2:
Impulse loading, input not measured.

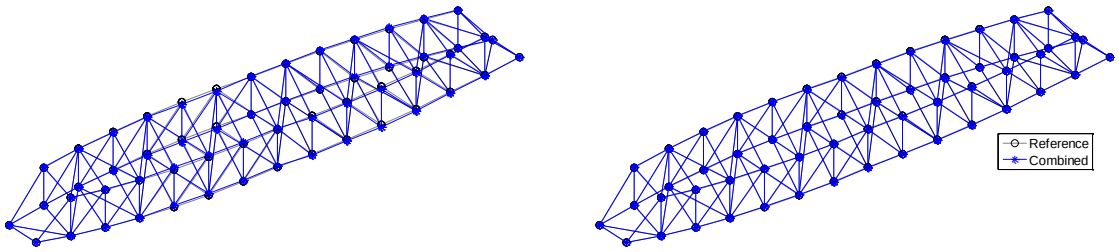


(c) Case 3:
Random excitation, input measured.



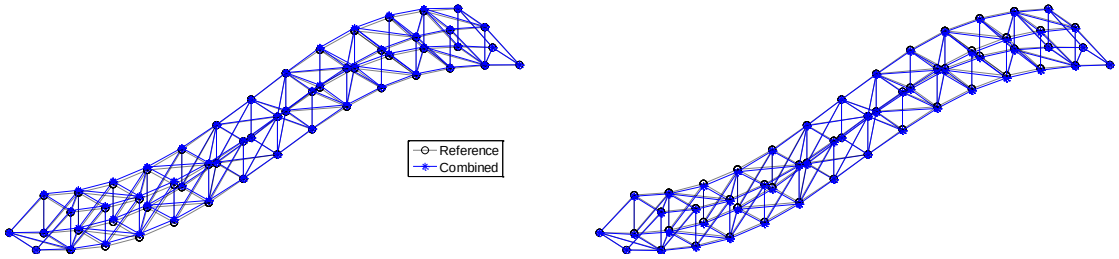
(d) Case 4:
Random excitation, input not measured.

Figure 4.17 Average error.



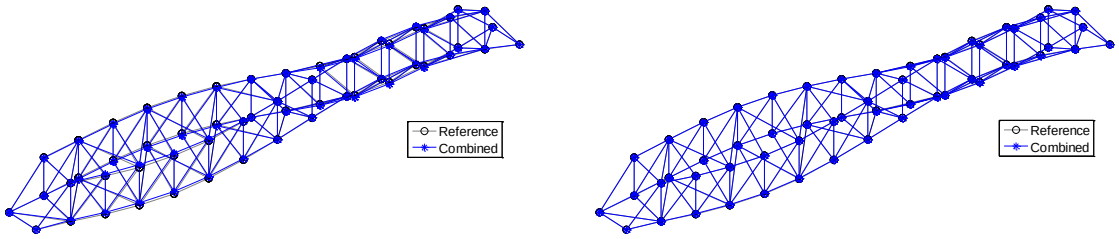
(a) Topology 1. (b) Topology 2.

Figure 4.18 Comparison of mode shapes: 3rd mode in case 4.



(a) Topology 1. (b) Topology 2.

Figure 4.19 Comparison of mode shapes: 6th mode in case 4.



(a) Topology 1. (b) Topology 2.

Figure 4.20 Comparison of mode shapes: 7th mode in case 4.

4.3 Summary

An automated, hierarchical, decentralized approach for modal analysis using smart sensors has been presented. Decentralized modal analysis consists of two main parts: (1) local feature extraction and (2) determination of global modal properties. Modal analysis is conducted independently in each sensor group to estimate the local modal information using NExT/ERA. Frequency for which the local groups identify the natural frequencies is utilized to delineate between the true and noise modes. Local mode shapes are then assembled using a least squares approximation to estimate the global mode shape.

The decentralized modal analysis approach was numerically validated. From the plate and truss examples, sensor topologies were investigated in terms of the size of local groups and the number of overlapping nodes. Larger groups with overlapping nodes were found to reduce errors in the assembled global mode shape. The numerical results showed global modal properties can be reliably and accurately estimated if sensor

topologies are appropriately selected to have sufficiently large local groups and multiple overlapping nodes.

In the next chapter, the Random Decrement Technique is considered as an alternative of NExT for more efficient data communication in decentralized data processing in the WSSN.

DECENTRALIZED RANDOM DECREMENT TECHNIQUE

The Random Decrement Technique (RDT) is in nature well-suited to the decentralized in-network processing for output-only system identification. While the NExT-based decentralized data aggregation proposed by Nagayama and Spencer (2007) has been recognized as an efficient way to extract modal information, RDT has potential to further reduce wireless communication in WSSN. This chapter develops the Random Decrement Technique for use in the decentralized computing environment of WSSNs.

5.1 Random decrement technique

Cole (1968) initially proposed RDT to estimate the dynamic properties of space structures excited by immeasurable ambient excitation. The basic assumption is that the dynamic response of a structure under ambient excitation is ergodic. From the structural response, n time history segments in the interval $[t_i, t_i+\tau]$ (i=1,⋯,n) are selected such that the displacement at t_i is equal to a specific trigger level. The response of a system at $t_i+\tau$ is comprised of three components:

1. Deterministic response due to the initial displacement at time t_i
2. Deterministic response due to the initial velocity at time t_i
3. Random response due to the random excitation between t_i and $t_i+\tau$

If the average is taken over a sufficiently large number of the segments, the third part of the response due to the random excitation will tend toward zero. Furthermore, the velocity at time t_i is uncorrelated with the displacement and has zero mean. Thus, the part of the response due to the initial velocity also tends to zero. The resulting RD function is the free vibration caused by a nonzero initial displacement.

Vandiver *et al.* (1982) provided a mathematical foundation for the random decrement function, showing that the RD function is proportional to the autocorrelation function for a linear, time-invariant system excited by a zero-mean, stationary, Gaussian random process. Later, Brincker *et al.* (1992, 1995) introduced a general triggering function and showed that RDT estimates a weighted sum of the auto- and cross-correlation functions and their time derivatives. For completeness, this section briefly describes the fundamental concept of RDT.

Consider stationary, Gaussian random processes $X_1(t)$ and $X_2(t)$. The auto- and cross-random decrement (RD) functions of $X_1(t)$ and $X_2(t)$ can be defined as the expected value of $X_1(t)$ and $X_2(t)$ given a trigger condition $C_{X_1(t_i)}$ (Brincker *et al.* 1992, 1995):

$$D_{X_1X_1}(\tau) = E\left[X_1(t_i+\tau) \middle| C_{X_1(t_i)} \right] \tag{5.1}$$

$$D_{X_2X_1}(\tau) = E\left[X_2(t_i+\tau) \middle| C_{X_1(t_i)} \right] \tag{5.2}$$

$$C_{X_1(t_i)} = \left[a_1 \le X_1(t_i) < a_2, b_1 \le \dot{X}_1(t_i) < b_2 \right] \tag{5.3}$$

where $E[\cdot]$ denotes expectation. Equations (5.1) and (5.2) also can be written as:

$$D_{X_1X_1}(\tau) = \frac{R_{X_1X_1}(\tau)}{\sigma^2_{X_1}}\bar{a} - \frac{\dot{R}_{X_1X_1}(\tau)}{\sigma^2_{\dot{X}_1}}\bar{b} \tag{5.4}$$

$$D_{X_2X_1}(\tau) = \frac{R_{X_2X_1}(\tau)}{\sigma^2_{X_1}}\bar{a} - \frac{\dot{R}_{X_2X_1}(\tau)}{\sigma^2_{\dot{X}_1}}\bar{b} \tag{5.5}$$

where $R_{X_1X_1}(\tau) = E[X_1(t)X_1(t+\tau)]$ and $R_{X_2X_1}(\tau) = E[X_2(t)X_1(t+\tau)]$ are the auto- and cross-correlation functions, respectively, and $\bar{a}$ and $\bar{b}$ can be defined in terms of the probability density functions $p_{X_1}(x)$ and $p_{\dot{X}_1}(\dot{x})$, respectively, as:

$$\bar{a} = \frac{\int_{a_1}^{a_2} x p_{X_1}(x)dx}{\int_{a_1}^{a_2} p_{X_1}(x)dx} \quad \text{and} \quad \bar{b} = \frac{\int_{b_1}^{b_2} \dot{x} p_{\dot{X}_1}(\dot{x})d\dot{x}}{\int_{b_1}^{b_2} p_{\dot{X}_1}(\dot{x})d\dot{x}} \tag{5.6}$$

In this study, the positive-point trigger condition (Asmussen, 1997) is considered:

$$C_{X_1(t_i)} = \left[\alpha_1\sigma_{X_1} \le X_1(t_i) < \alpha_2\sigma_{X_1}, -\infty \le \dot{X}_1(t_i) < \infty \right] \tag{5.7}$$

where $0 \le \alpha_1 < \alpha_2 \le \infty$. Equations (5.4) and (5.5) are then:

$$D_{X_1X_1}(\tau) = \frac{R_{X_1X_1}(\tau)}{\sigma^2_{X_1}}\bar{a} \text{ and } D_{X_2X_1}(\tau) = \frac{R_{X_2X_1}(\tau)}{\sigma^2_{X_1}}\bar{a}\,. \tag{5.8}$$

Now, consider the equation of motion for a multi-degree-of-freedom system:

$$\mathbf{M}\ddot{\mathbf{X}}(t) + \mathbf{C}\dot{\mathbf{X}}(t) + \mathbf{K}\mathbf{X}(t) = \mathbf{F}(t) \tag{5.9}$$

where **M**, **C**, and **K** are the mass, damping, and stiffness matrices, respectively. The externally applied force is assumed to be a zero-mean, stationary, Gaussian random process, and the mass, damping, and stiffness are assumed to be deterministic. James *et al.* (1993) showed that the correlation function between responses and a reference response X_k is the homogeneous solution of the equation of the motion:

$$\mathbf{M}\ddot{\mathbf{R}}_{\mathbf{X}X_k}(\tau) + \mathbf{C}\dot{\mathbf{R}}_{\mathbf{X}X_k}(\tau) + \mathbf{K}\mathbf{R}_{\mathbf{X}X_k}(\tau) = \mathbf{0} \tag{5.10}$$

Substituting Equation (5.8) into Equation (5.10), and multiplying by $\sigma^2_{X_1}/\bar{a}$ yields

$$\mathbf{M}\ddot{\mathbf{D}}_{\mathbf{X}X_k}(\tau) + \mathbf{C}\dot{\mathbf{D}}_{\mathbf{X}X_k}(\tau) + \mathbf{K}\mathbf{D}_{\mathbf{X}X_k}(\tau) = \mathbf{0} \tag{5.11}$$

where $\mathbf{D}_{\mathbf{X}X_k}(\tau)$ is the vector of RD functions, with the scalar response process X_k being referenced for the trigger condition. Thus, the RD functions $\mathbf{D}_{\mathbf{X}X_k}(\tau)$ are seen to satisfy the homogeneous equation of motion.

The RD functions can be estimated from data as follows:

$$\hat{D}_{X_j X_k}(\tau) = \frac{1}{N}\sum_{i=1}^{N} x_j(t_i + \tau) \Big| C_{x_k(t_i)} \tag{5.12}$$

where $\hat{D}_{X_j X_k}(\tau)$ is the RD function obtained from $x_j(t)$ with respect to the reference $x_k(t)$, N is the total number of trigger events, $C_{x_k(t_i)}$ is the specified positive point trigger condition, and t_i is the i^{th} time obtained from the trigger event $C_{x_k(t_i)}$. When $j = k$, $\hat{D}_{jj}(\tau)$ is an auto-RD function.

Once the RD functions are obtained, modal properties of the structure can be estimated using a wide range of system identification methods. For example, the RD function can be directly used in the time domain methods such as ERA and SSI. Also, the unscaled frequency response function obtained by taking the Fourier transform of the RD function can be used as an input to frequency domain methods such as Peak Peaking and FDD. Figure 5.1 illustrates the use of the RD functions for various identification methods.

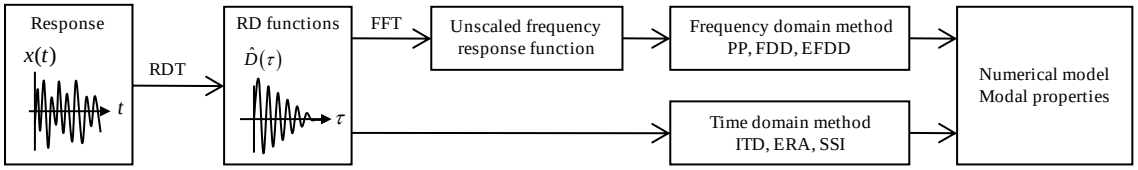


Figure 5.1 Application of RDT for system identification.

5.2 RDT-based decentralized data aggregation

RDT can significantly enhance the efficiency of data aggregation in the distributed computing environment in WSSNs. As previously described, central data collection and processing in WSSNs can cause severe data congestion due to limited communication bandwidth; decentralized in-network processing for data condensation can mitigate this problem.

To better understand the efficiency of the decentralized processing, consider the centralized implementation of community-wide data processing shown in Figure 5.2. Assuming the community has n_s nodes, each sensor node measures data and transmits to Node 1 that provides reference information for correlation function estimation. For time history records of length N and n_d averages, the amount of transmitted data is $N \times n_d \times (n_s - 1)$.

Nagayama *et al.* (2007) proposed a decentralized NExT implementation, taking advantage of each node's computing capability to reduce data communication (see Figure 5.3). Node 1 sends a measured time history record as a reference signal to each node. Correlation functions are calculated in all nodes in the community and subsequently collected at Node 1. The amount of transmitted data is at most $N \times n_d + N/2 \times (n_s - 1)$. As the numbers of nodes or averages increase, the efficiency of the decentralized NExT implementation becomes clearer.

The decentralized RDT implementation shown in Figure 5.4 can further reduce data communication. In this approach, Node 1 sends the trigger information found in

Equation (5.7) to all nodes in the community. Once the reference is received, each node calculates the RD functions that are subsequently collected at Node 1. Note that (1) the trigger information is in general much shorter than the time history record used as the reference for NExT, and (2) transmission of the reference takes a significant portion of the total communication in the decentralized NExT implementation, particularly when long records are used. Thus, the RDT-based decentralized data aggregation can considerably reduce data communication requirement.

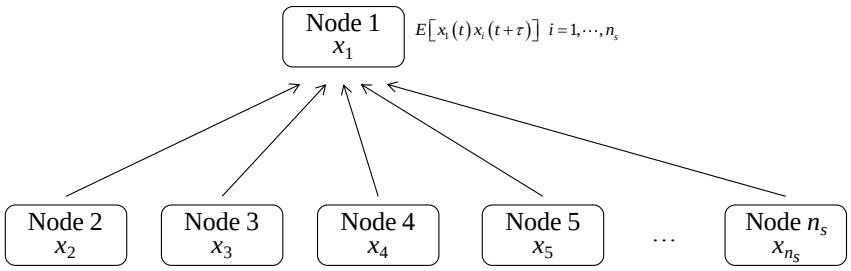


Figure 5.2 Centralized NExT implementation (Nagayama *et al.* 2007).

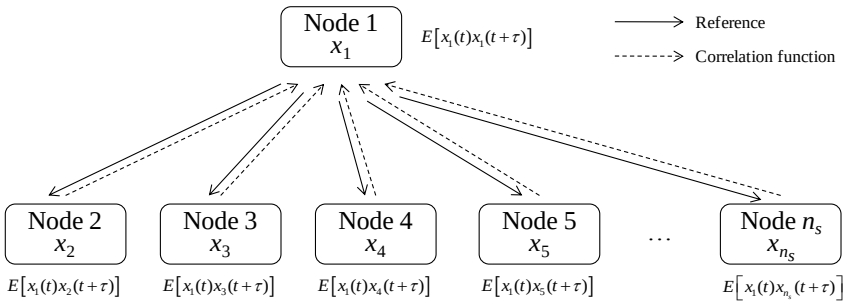


Figure 5.3 Decentralized NExT implementation (Nagayama *et al.* 2007).

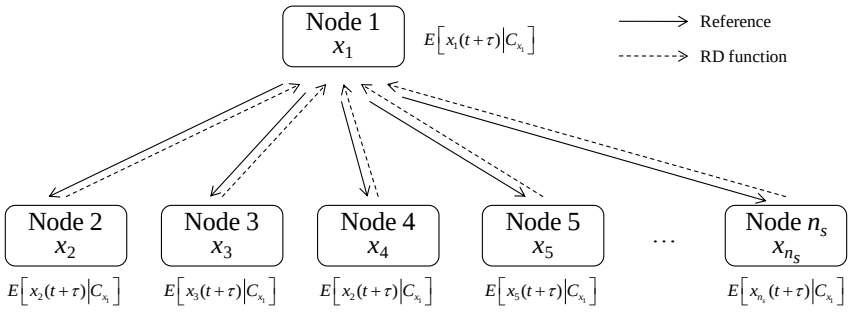


Figure 5.4 Decentralized RDT implementation.

Data communication required by the decentralized RDT implementation is closely related to the number of triggering points. For the positive-point trigger condition found in Equation (5.7), the expected number of triggering points is (Asmussen 1997):

$$E\left[n\left(a_1,a_2\right)\right]=\left(N_X-N_\tau\right)\cdot\int_{a_1}^{a_2}p_X\left(x\right)dx \tag{5.13}$$

where $n(a_1, a_2)$ is the number of triggering points between a_1 and a_2, Δt is the sampling rate, $p_X(x)$ the probability density function of $X(t)$, and N_X and N_τ are the number points in $X(t)$ and the RD function, respectively. Thus, the total number of points to be wirelessly transferred in the decentralized RDT implementation is:

$$\left(N_X - N_\tau\right)\int_{a_1}^{a_2} p_X\left(x\right)dx + N_\tau\left(n_s - 1\right) \tag{5.14}$$

To better understand the efficiency of the decentralized RDT implementation from the data communication perspective, consider data communication requirements in each of three implementations (i) centralized data collection, (ii) decentralized NExT, and (iii) decentralized RDT. Assuming $N = 1024$, $n_d = 20$, $n_s = 6$, $N_X = 20{,}480$, $N_\tau = 512$, $a_1 = \sigma_X$, $a_2 = 1.5\times\sigma_X$, and $X(t)$ follows the Gaussian distribution, the numbers of transferred data points are:

- Centralized implementation: $N \cdot n_d \cdot \left(n_s - 1\right) = 102{,}400$
- Decentralized NExT: $N \cdot n_d + N/2 \cdot \left(n_s - 1\right) = 23{,}040$
- Decentralized RDT: $\left(N_X - N_\tau\right)\int_{a_1}^{a_2} p_X\left(x\right)dx + N_\tau\left(n_s - 1\right) = 4{,}394$

In this particular example, the decentralized RDT implementation reduces data communication to 4.3 % of the centralized implementation and 19.1 % of the decentralized NExT implementation.

RDT exhibits less data communication requirements than NExT because RDT does not require the entire time history for a reference, but rather needs only the trigger information. The next section presents a numerical example to validate the feasibility of RDT in the distributed computing environment for WSSNs.

5.3 Numerical validation

The efficacy of the proposed RDT-based decentralized data aggregation is numerically investigated. The performance in the distributed computing environment is checked in two criteria: (1) estimation accuracy and (2) data communication requirements. To assess the accuracy, global modal properties are estimated from local information (Sim *et al.* 2010) obtained by the RDT-based decentralized data aggregation. The NExT-based decentralized data aggregation is employed as a reference.

Consider the simply supported truss model in Figure 5.5. The truss consists of 53 elements, for which the area moments of inertia are $I_x = 5.0\times10^{-9}\,\text{m}^4$ and $I_y = I_z = 2.5\times10^{-9}\,\text{m}^4$, the elastic modulus is 1.999×10^{11} Pa, the shear modulus is 7.692×10^{11} Pa, and the mass density is $7.827\times10^{3}\,\text{kg/m}^3$. A band limited white noise on the interval 0 Hz to 120 Hz is applied vertically as shown in Figure 5.5. Vertical accelerations at all lower nodes are obtained, and 5% RMS noise is added to all measurements. For subsequent reference, the transfer functions between the input excitation and the measured accelerations at points N1 and N2 (see Figure 5.5) are obtained from the analytical model as shown in Figure 5.6. Because the 5th mode is not excited by the given input, the 5th peak is not present in the transfer functions.

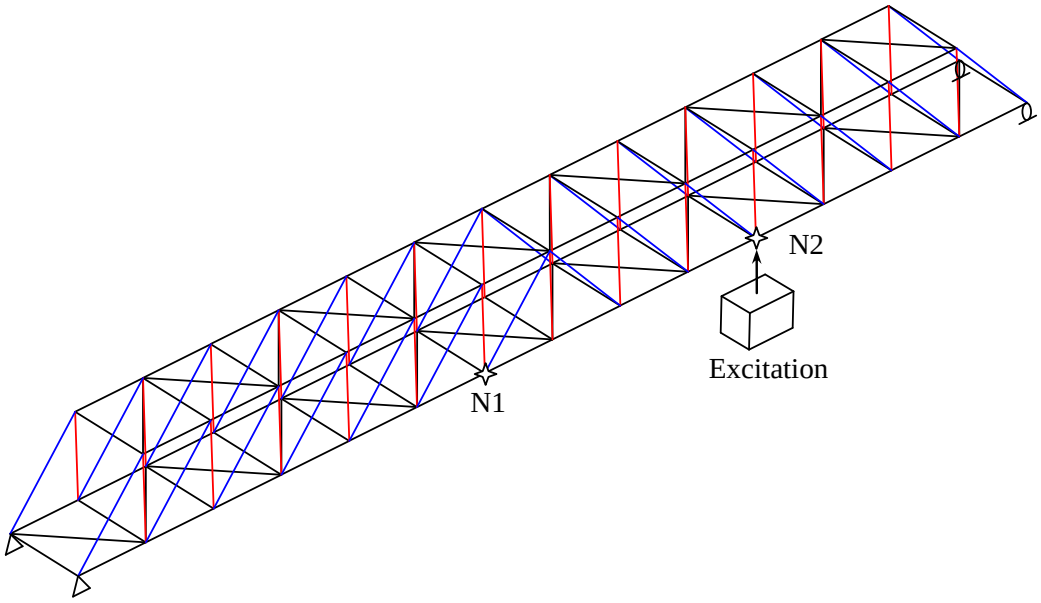


Figure 5.5 Truss model.

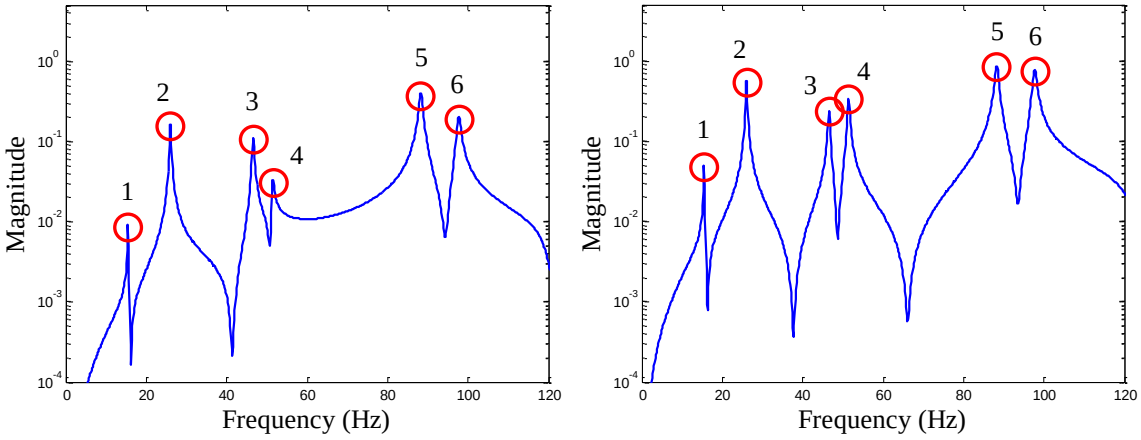


Figure 5.6 Transfer functions from the input excitation to accelerations at nodes N1 and N2.

The selected sensor topology shown in Figure 5.7 (Sim *et al.* 2010) consists of six groups with 26 nodes in total. Each local group has six nodes, two of which are the overlapping nodes. The second node of each group (see Figure 5.8) is selected as the cluster-head that provides the reference information (i.e., time history records for NExT and trigger information for RDT).

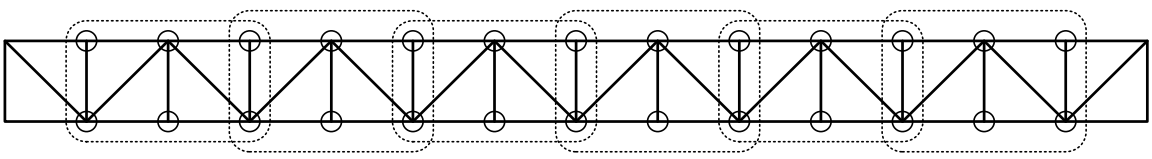


Figure 5.7 Sensor topology (top view).

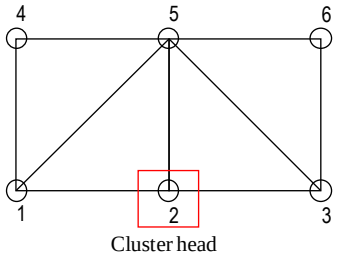


Figure 5.8 Topology of a local group (top view).

Once acceleration responses are obtained, RD and correlation functions are decentrally estimated in each local group. Subsequently, ERA is employed to estimate local modal properties, from which global modal properties are obtained (Sim *et al.* 2010). Simulation is repeated 100 times to accommodate the randomness.

The interval (a_1, a_2) for the positive-point trigger crossing is optimally chosen for RDT to accurately estimate modal properties while maintaining a low number of trigger crossings by maximizing the following function with respect to (a_1, a_2):

$$\frac{MAC}{E \cdot S} \tag{5.15}$$

where

$$MAC = \sum_i \frac{\left| \phi_{i,\text{est}}{}^T \phi_{i,\text{ext}} \right|^2}{\left(\phi_{i,\text{est}}{}^T \phi_{i,\text{ext}} \right)\left(\phi_{i,\text{est}}{}^T \phi_{i,\text{ext}} \right)} \tag{5.16}$$

$$E = \sum_i \frac{\left| f_{i,\text{est}} - f_{i,\text{ext}} \right|}{f_{i,\text{ext}}} \tag{5.17}$$

S is the total number of trigger crossings, $\phi_{i,est}$ and $\phi_{i,ext}$ are the i^{th} estimated and exact global mode shapes, respectively, and $f_{i,est}$ and $f_{i,ext}$ are the i^{th} estimated and exact natural frequencies, respectively. Note the modal assurance criterion (Allemang and Brown 1982) is utilized in Equation (5.16). Equation (5.15) is evaluated in the intervals with lower and upper bounds from 0 to 3σ as shown Figure 5.9: the optimal interval (a_1, a_2) that maximizes $MAC/(E \cdot S)$ is (1.75σ, 2.50σ). While determining the optimal interval in full-scale applications may not be possible due to the lack of the exact mode shapes, the trend shown in Figure 5.9 can be utilized to obtain appropriate intervals for accurate modal identification. In the subsequent analysis, the interval (1.75σ, 2.50σ) is utilized for the positive-point trigger crossing in the RDT-based decentralized data aggregation.

As shown in Figure 5.10, the global mode shapes obtained by RDT- and NExT-based decentralized data aggregation approaches match well with the exact mode shapes from the finite element model. Note that the 1st mode is not identified in both RDT- and NExT-based decentralized data aggregation approaches due to the low energy by the given responses (see Figure 5.6). The 1st mode is dominated by a transverse motion; it is neither well excited nor well observed, because the input excitation is vertically applied and only vertical accelerations are measured. MAC between the exact and estimated mode shapes shown in Table 5.1 is consistent with the observation made in Figure 5.10. As such, the decentralized implementation of RDT accurately estimates modal properties.

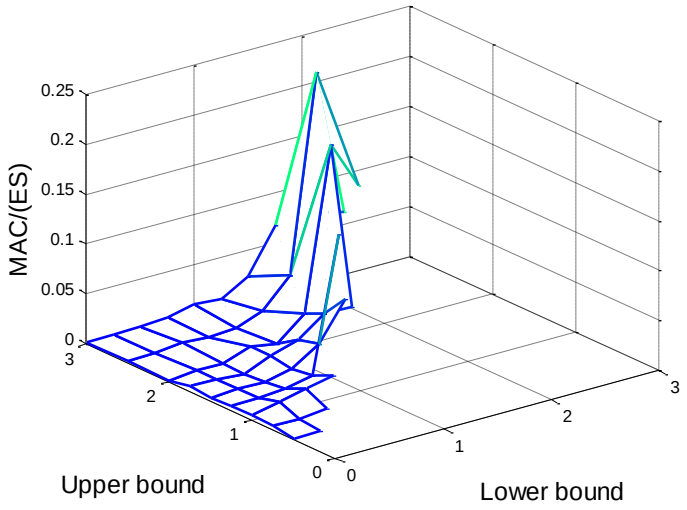
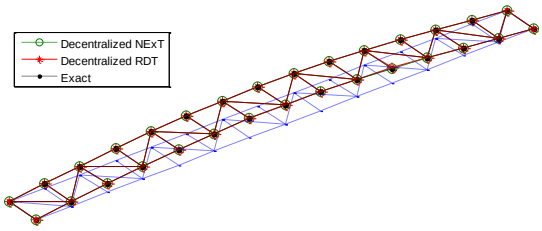
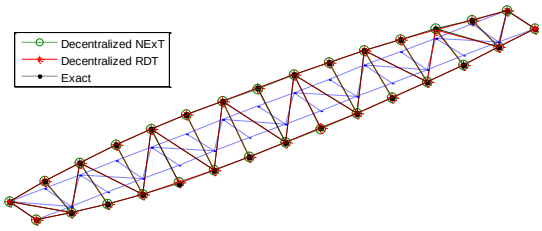


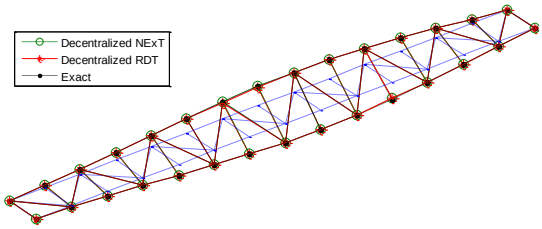
Figure 5.9 *MAC*/(*E*·*S*).



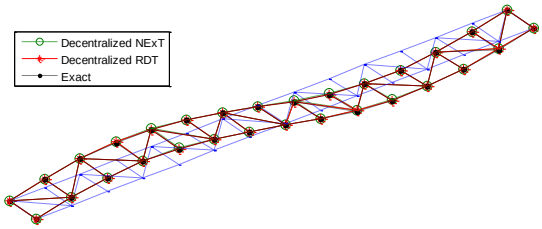
(a) Mode 2 (26.02 Hz).



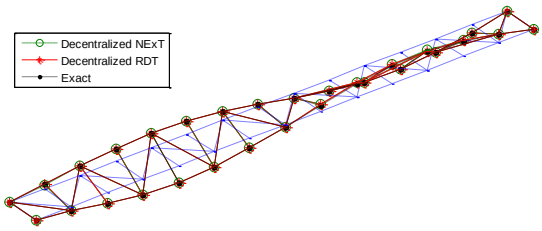
(b) Mode 3 (46.58 Hz).



(c) Mode 4 (51.39 Hz).



(d) Mode 6 (88.26 Hz).



(e) Mode 7 (97.65 Hz).

Figure 5.10 Global mode shapes (Note that Modes 1 and 5 are not represented well in the response, as shown in Figure 5.6).

Table 5.1 MAC between the exact and estimated mode shapes.

Mode	NExT-based decentralized data aggregation	RDT-based decentralized data aggregation
2	0.9999	0.9999
3	0.9998	0.9998
4	0.9995	0.9997
6	0.9144	0.9649
7	0.9251	0.9823

Now, the efficiency of RDT is investigated in terms of data communication. For this purpose, the identical acceleration records are used in the numerical simulation for both NExT/ERA and RDT/ERA. The length of an acceleration signal is 21,504 points, determined as:

- Number of averages $N_{\text{average}} = 20$
- Number of FFT points $N_{\text{FFT}} = 2048$
- Overlap between adjacent windows: $R_{\text{overlap}} = 0.5$

$$\text{Data points} = N_{\text{FFT}} \times \left(\left(1 - R_{\text{overlap}} \right) \times N_{\text{average}} + R_{\text{overlap}} \right) = 21,504 \tag{5.18}$$

In NExT-based decentralized data aggregation, the reference acceleration of 21,504 points in each group is broadcast to the leaf nodes by the cluster-head to compute the correlation functions.

When applying RDT, the trigger crossing information in the cluster-head is required as the reference in the leaf nodes. The number of trigger crossings in each of the cluster-heads for the six local groups are indicated in Table 5.2, along with the amount data required for NExT. The total transmitted data over the radio link in the case of RDT is only 22% of NExT. Thus, the efficiency in the data communication can be significantly improved.

Table 5.2 Transferred data for NExT and RDT.

Local group	NExT	RDT		RDT/NExT (%)
	Total[1]	Trigger crossing[2]	Total[3]	
1	26,629	729.35	5,854.35	21.98
2	26,629	727.82	5,852.82	21.98
3	26,629	732.35	5,857.35	22.00
4	26,629	730.80	5,855.80	21.99
5	26,629	728.06	5,853.06	21.98
6	26,629	729.75	5,854.75	21.99
Total	**159,774**	7,378.13	**35,128.13**	**21.99**

[1] The total is the sum of reference data (21,504 points) and correlation functions (1025 from 5 leaf nodes).

[2] The number of trigger crossings is a mean value obtained from 100 simulations.

[3] The total is the sum of trigger crossings and RD functions (1025 from 5 leaf nodes).

5.4 Experimental validation

5.4.1 Experimental setup

The performance of RDT-based decentralized data aggregation in the WSSN is experimentally investigated using the truss structure shown in Figure 5.11. As in the previous numerical example, the estimation accuracy for global modal properties and data communication requirements are assessed. *DecentralizedDataAggregation* installed on the Imote2 sensors is employed to decentrally calculate RD functions as well as correlation functions that are used as the reference.

The experimental structure considered is a simply supported truss that consists of steel hollow circular tubes with an inner diameter of 0.428 inches and an outer diameter of 0.612 inches (see Figure 5.11). A shaker is used to vertically excite the truss with a band-limited white noise on the interval 0–100 Hz.

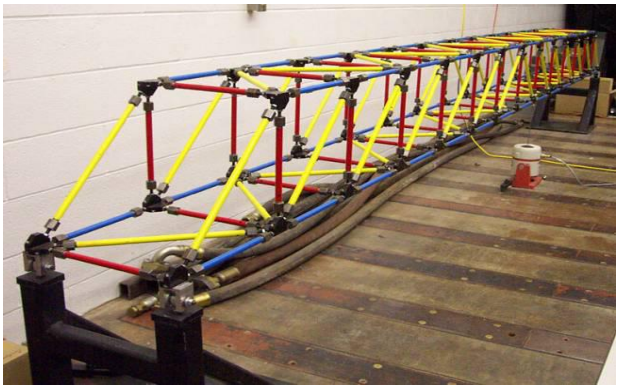


Figure 5.11 Truss structure.

A total of 14 Imote2 sensors with SHM-A acceleration sensor boards are installed on the bottom chord of the truss as shown in Figure 5.12 and Figure 5.13. The SHM-A, developed at the University of Illinois at Urbana-Champaign by Rice and Spencer (2009), is a general purpose, multimetric sensor board for the Imote2 sensor platform that can measure up to 3-axis of accelerations, light, temperature, and humidity. The sensor network is divided into four local sensor groups that consist of four or six sensor nodes. Sensor nodes S2, S6, S10, and S14 in Figure 5.13 serve as cluster-heads in each local sensor groups.

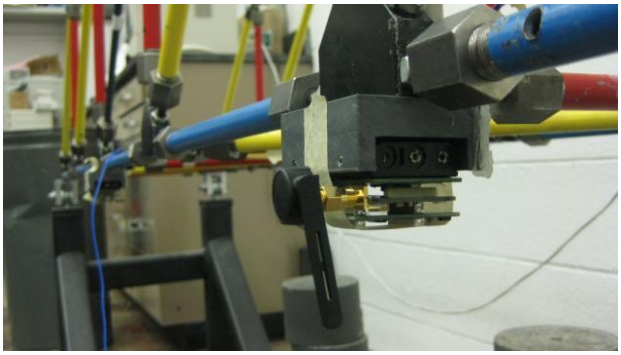


Figure 5.12 Imote2 sensor node attached to the bottom chord of the truss.

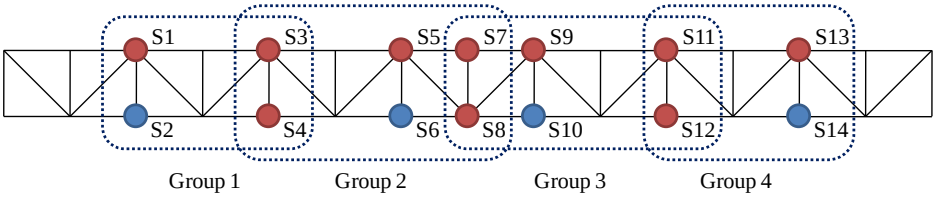


Figure 5.13 Sensor topology (plan view) (S2, S6, S10, and S14 are cluster-heads).

5.4.2 Data acquisition and in-network processing

Vertical accelerations are measured at each sensor node with a sampling rate of 280 Hz, with a 70 Hz cutoff frequency. The measured acceleration time histories are 21,504 points in length for both correlation and RD function estimation using *DecentralizedDataAggregation*. With respect to calculation of the correlation functions, a signal with 21,504 points allows 20 averages if 2,048 points of FFT and 50% overlap between windows are specified.

DecentralizedDataAggregation is consecutively employed to estimate the correlation and RD functions, as shown in Figure 5.14 and Figure 5.15, respectively. The positive-point trigger crossing with an interval of (σ, 2.5σ) was found to produce the best results. Local modal properties are estimated at the base station for both cases by ERA using the correlation and RD functions that are calculated on the Imote2 sensors, and subsequently global modal properties are obtained. In addition, raw acceleration time history data from all sensor nodes are centrally collected to provide a reference in comparison of NExT and RDT in the distributed computing environment. Using the centrally collected accelerations, NExT/ERA and RDT/ERA are employed to estimate reference modal properties.

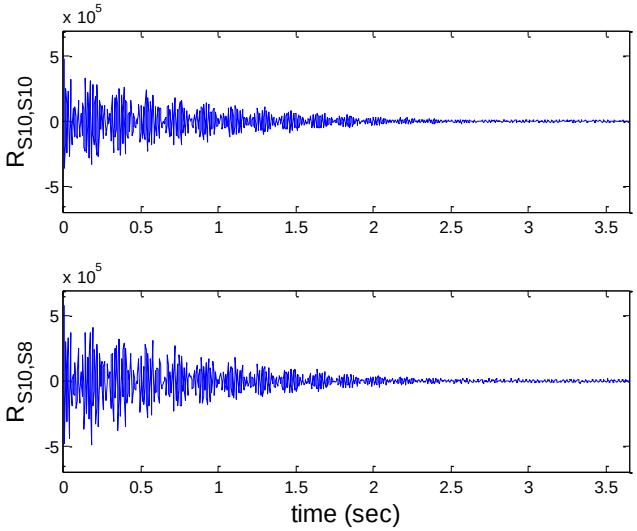


Figure 5.14 Auto-correlation of S10 (top) and cross-correlation between S10 and S8 (bottom).

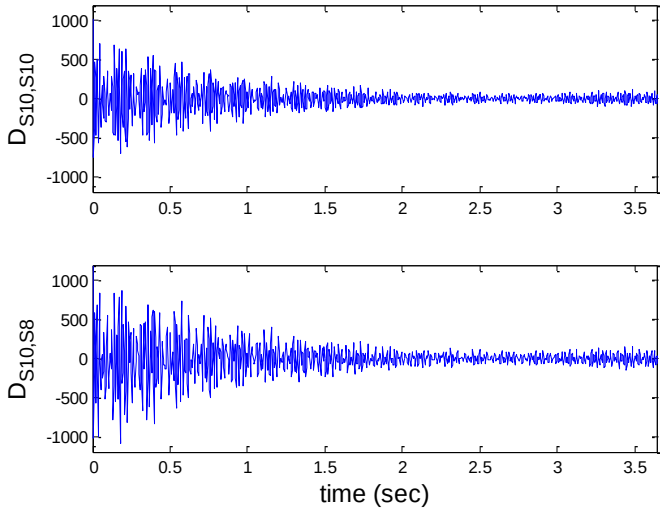


Figure 5.15 Auto-RD function of S10 (top) and cross-RD function between S10 and S8 (bottom).

5.4.3 Data processing at base station

Table 5.3 summarizes the identified natural frequencies of the truss for system identification method:

- Reference 1: Centralized processing using NExT/ERA
- Reference 2: Centralized processing using RDT/ERA
- Case 1: Decentralized processing using NExT/ERA
- Case 2: Decentralized processing using RDT/ERA

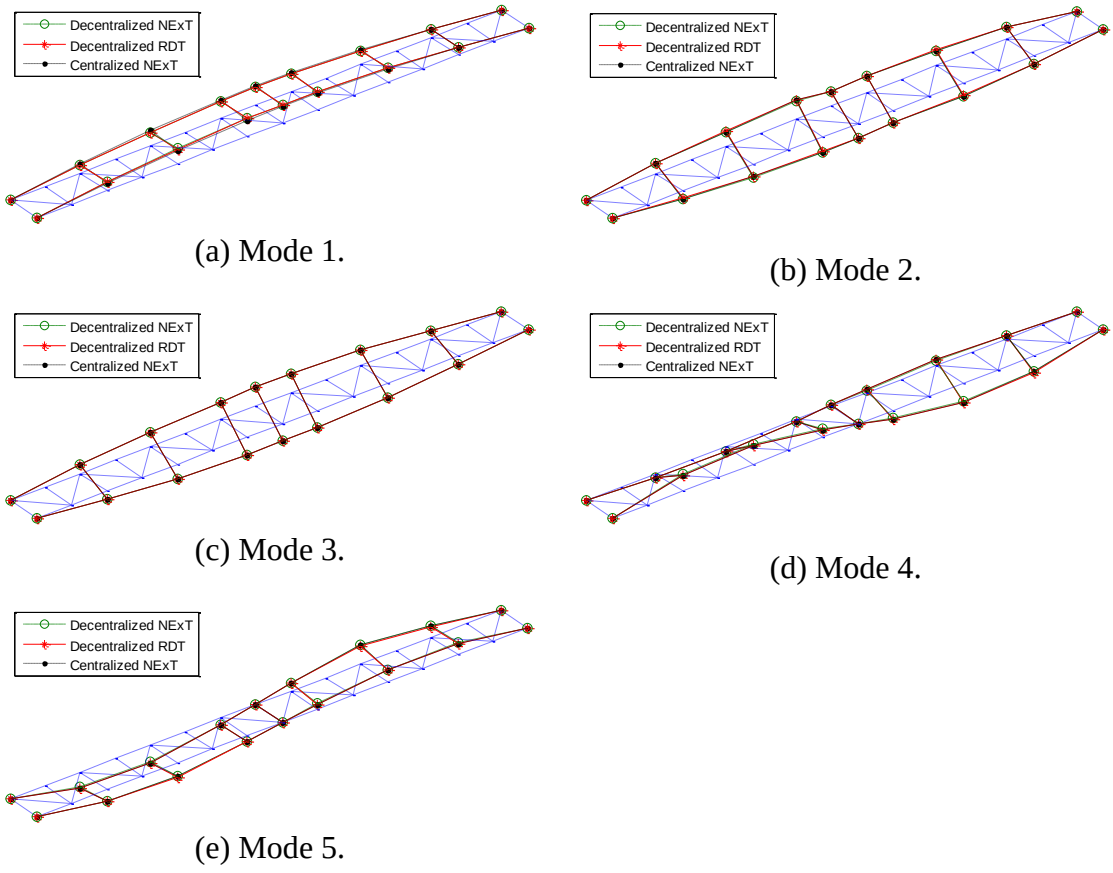
Compared to the cases of the centralized processing, both NExT/ERA and RDT/ERA based on the decentralized processing estimate natural frequencies with a reasonable accuracy (see Table 5.3).

Table 5.3 Identified natural frequencies.

Mode	Natural frequencies (Hz)			
	Centralized processing (reference)		Decentralized processing	
	Reference 1	Reference 2	Case 1	Case 2
	NExT/ERA	RDT/ERA	NExT/ERA	RDT/ERA
1	20.77	20.78	20.81	20.77
2	33.26	33.28	33.32	33.30
3	41.61	41.58	41.55	41.60
4	64.16	64.24	64.08	64.16
5	69.38	69.40	69.37	69.37

The global mode shapes for each case are compared as shown in Figure 5.16, which shows good agreements with the reference. Note that only Reference 1 is shown in Figure 5.16, because global mode shapes from Reference 1 and 2 are visibly

indistinguishable. The MAC values between the global mode shapes from the decentralized processing (Cases 1 and 2) and centralized processing (References 1 and 2) are calculated as shown in Table 5.4, all MAC values are close to 1, which is consistent with Figure 5.16. Both NExT/ERA and RDT/ERA in decentralized processing environment estimate global mode shapes accurately.



(a) Mode 1.

(b) Mode 2.

(c) Mode 3.

(d) Mode 4.

(e) Mode 5.

Figure 5.16 Global mode shapes.

Table 5.4 MAC with respect to the reference global mode shapes.

Mode	Reference 1 Centralized NExT/ERA		Reference 2 Centralized RDT/ERA	
	Case 1 NExT/ERA decentralized	Case 2 RDT/ERA decentralized	Case 1 NExT/ERA decentralized	Case 2 RDT/ERA decentralized
1	0.9997	1.0000	0.9997	1.0000
2	0.9980	0.9992	0.9988	0.9981
3	0.9996	1.0000	0.9996	1.0000
4	0.9943	0.9990	0.9854	0.9939
5	0.9975	0.9988	0.9966	0.9982

Wireless data communication required in correlation and RD function estimation is investigated to identify the efficiency of RDT-based decentralized data as summarized in Table 5.5: RDT reduces wireless data communication to 28.47% of NExT. As previously described, NExT requires an entire time history of the reference signal transferred for each local sensor group, which results in more data communication than RDT.

Table 5.5 Transferred data for NExT and RDT.

Local group	Transferred data (points)						RDT/NExT (%)
	Case 1 - NExT			Case 2 - RDT			
	Ref.[1]	CF[2]	Total	Ref.[3]	RDF[4]	Total	
1	21,504	3,075	24,579	3,211	3,075	6,286	25.57
2	21,504	5,125	26,629	3,168	5,125	8,293	31.14
3	21,504	5,125	26,629	3,171	5,125	8,296	31.15
4	21,504	3,075	24,579	3,210	3,075	6,285	25.57
Total	86,016	16,400	**102,416**	12,760	16,400	**29,160**	**28.47**

[1] Ref. is the length of the reference data in NExT.
[2] CF is the total length of correlation functions (number of leaf nodes × length of a correlation function).
[3] Ref. is the length of the reference data in RDT. This number varies depending on the reference data.
[4] RDF is the total length of RD functions (number of leaf nodes × length of a RD function).

5.5 Summary

The experiment using the Imote2 sensor nodes with *DecentralizedDataAggregation* embedded was conducted using the truss structure shown in Figure 5.11. In the experiment, the RDT-based decentralized data aggregation was demonstrated to be efficient in terms of wireless communication with accurate estimation results. In addition, performance of decentralized modal analysis and software reliability of *DecentralizedDataAggregation* were successfully verified in the laboratory environment.

DAMAGE DETECTION USING MULTIMETRIC SENSING

Structural damage detection can be enhanced by using a heterogeneous mix of measurands containing both local and global information. Herein, a damage detection method that extends the Stochastic DLV method (Bernal 2006) to use acceleration and strain in combination is described. This section first describes the DLV method briefly, provides a mathematical formulation for the extension to use acceleration and strain in the DLV method, and presents a numerical example to validate the multimetric approach.

6.1 Damage locating vector method

For completeness, the DLV method is briefly reviewed in this section. The flexibility matrices of a linear structure before and after damage are determined from the measured data and are denoted as F_u and F_d, respectively. Assume the load vectors L that produce identical displacements at the sensor locations before and after damage exist and can be written as

$$(F_u - F_d) \cdot L = F_\Delta \cdot L = 0 \tag{6.1}$$

These vectors, L, are termed the damage locating vectors (DLV). The DLVs constitute a set of loads that induce no stress in the damaged elements. Excluding the trivial case where F_Δ=0, L is in the null space of F_Δ and can be determined using the singular value decomposition.

Once determined, each of the DLVs can be applied to a numerical model of the undamaged structure, and the stress in each element calculated and normalized as follows:

$$\bar{\sigma}_j = \frac{\sigma_j}{\max_k(\sigma_k)} \quad \text{and} \quad \sigma_j = \sum_{i=1}^{m} \text{abs}\left(\frac{\sigma_{ij}}{\max_k(\sigma_{ik})} \right) \tag{6.2}$$

where σ_j is the normalized accumulated stress, σ_{ij} is the stress in the jth element induced by the ith DLV, and m is the number of DLVs. If the normalized stress in an element is zero, then this element is a damage candidate. However in practice, the cumulative stress may not be exactly zero due to intrinsic uncertainties such as measurement noise or model error. Thus, a threshold value needs to be selected to determine the damaged elements. If the cumulative stress of an element is less than the selected threshold, then the element is considered as a damage candidate. This nonzero error stress at the damaged element caused by uncertainties is required to be small for reliable damage detection.

Bernal (2006) extended the DLV method to accommodate the output-only case (*i.e.*, the input excitation is not measured), resulting in the stochastic DLV method. In this approach, an alternative matrix that spans the same null space as F_Δ in Equation (6.1) is

used to determine the DLVs. The stochastic DLV method has been extended, implemented, and experimentally verified on a network of smart sensors (Nagayama and Spencer 2007). In the next section, the stochastic DLV method is further extended to accommodate multimetric sensed data.

6.2 Derivation

The DLV method and the flexibility matrix estimation are based on global measurements such as displacement, velocity, and acceleration. Only accelerations have been used in the experimental verification of the DLV method (Gao *et al.* 2004; Nagayama and Spencer 2007). Herein, the mathematical formulation for the approach to combine the strain with the displacement, velocity, and acceleration in the stochastic DLV method is provided.

6.2.1 Strain flexibility matrix

Consider the flexibility matrix F_d that relates the external load to the displacement of the structure such that

$$\{u\} = F_d \{L\} \tag{6.3}$$

where $\{u\}$ and $\{L\}$ are the displacement and load vectors, respectively. The strain flexibility F^* is defined herein as

$$\{\varepsilon\} = F^* \{L\} \tag{6.4}$$

where $\{\varepsilon\}$ is the strain vector. Assume that a linear transformation T between the strain and the displacement exists such that

$$\{\varepsilon\} = T \{u\} \tag{6.5}$$

The strain flexibility matrix is then written as:

$$F^* = TF_d \tag{6.6}$$

In terms of the modal parameters, the flexibility matrix and the strain flexibility matrix can be expressed as:

$$F_d = \Phi_d D \Phi_d^{\mathrm{T}} \tag{6.7}$$

$$F^* = TF_d = T\Phi_d D \Phi_d^{\mathrm{T}} = \Phi_s D \Phi_d^{\mathrm{T}} \tag{6.8}$$

where Φ_d is the displacement mode shape matrix, Φ_s is the strain mode shape matrix, and

$$D = diag\left(\left[\left(d_1/\omega_1\right)^2 \quad \left(d_2/\omega_2\right)^2 \quad \cdots \quad \left(d_j/\omega_j\right)^2 \quad \cdots\right]\right) \tag{6.9}$$

where d_j and ω_j are the mass normalization constant and the natural frequency of the jth mode, respectively. Equation (6.8) shows that the strain flexibility matrix is not symmetric in the definition given in Equation (6.4), while the stochastic DLV method requires the symmetry of the flexibility matrix (Bernal 2006).

This difficulty can be overcome by introducing the pseudo element force vector $\{L_e\}$ such that

$$\{L\} = T^{\mathrm{T}} \{L_e\} \tag{6.10}$$

Note that for statically determinate structures, the pseudo element force $\{L_e\}$ is equal to the element force. Then the strain flexibility relating the strain and the pseudo element force can be defined as:

$$\{\varepsilon\} = F_s \{L_e\} \tag{6.11}$$

Substituting Equations (6.3), (6.5), and (6.10) into Equation (6.11),

$$F_s \{L_e\} = TF_d T^{\mathrm{T}} \{L_e\} \tag{6.12}$$

For an arbitrary force $\{L_e\}$, Equation (6.12) is satisfied if and only if

$$F_s = TF_d T^{\mathrm{T}} \tag{6.13}$$

or in terms of the modal parameters

$$F_s = T\left(\Phi_d D {\Phi_d}^{\mathrm{T}}\right) T^{\mathrm{T}} = \Phi_s D {\Phi_s}^{\mathrm{T}} \tag{6.14}$$

Thus, the symmetry of the strain flexibility defined in Equation (6.11) is ensured.

The flexibility matrix including both displacement and strain can be shown to be symmetric as follows. Consider a system under two load vectors, L_1 in the coordinate where displacements are defined, and L_2 which is a pseudo element force. The displacement and strain are thus

$$u = FL_1 + FT^{\mathrm{T}} L_2 \tag{6.15}$$

$$\varepsilon = Td = TFL_1 + TFT^{\mathrm{T}} L_2 \tag{6.16}$$

which leads to the combined flexibility matrix such that

$$\begin{bmatrix} u \\ \varepsilon \end{bmatrix} = \begin{bmatrix} F & FT^{\mathrm{T}} \\ TF & TFT^{\mathrm{T}} \end{bmatrix} \begin{bmatrix} L_1 \\ L_2 \end{bmatrix} \tag{6.17}$$

The symmetry of the combined flexibility matrix enables the use of multimetric data in the stochastic DLV method.

6.2.2 Multimetric data in the stochastic DLV method

The formulation herein follows the derivation of the stochastic DLV method (Bernal 2006), but extends the approach to accommodate the combined use of strain and acceleration. Consider a system represented in the following state space form

$$\begin{aligned} \dot{x} &= Ax + Bu \\ y &= C_d x + D_d u \end{aligned} \tag{6.18}$$

where y is the displacement vector, $C_d \in R_{m \times N}$, $D_d \in R_{m \times n}$, N is the order of the system, m and n are the numbers of displacements and inputs, respectively. Because the input force

is not directly transmitted to the displacement, D_d=0. Taking the first and second derivatives of the output in Equation (6.18) yields

$$\dot{y} = C_d A x + C_d B u \tag{6.19}$$

$$\ddot{y} = C_d A^2 x + C_d A B u + C_d B \dot{u} \tag{6.20}$$

The fact that the input force is not directly transmitted to the velocity leads to

$$C_d B = 0 \tag{6.21}$$

In addition, the output and direct transmission matrices for the acceleration in Equation (6.21) can be written as:

$$C_a = C_d A^2 \tag{6.22}$$

$$D_a = C_d A B \tag{6.23}$$

Substituting C_d in Equation (6.22) into Equations (6.21) and (6.23),

$$C_a A^{-2} B = 0 \tag{6.24}$$

$$C_a A^{-1} B = D_a \tag{6.25}$$

For velocity and displacement, similar equations can be derived as (Bernal 2006)

$$C_v A^{-1} B = C_d B = 0 \tag{6.26}$$

$$C_v B = C_d A B = D_a \tag{6.27}$$

where C_v and C_d are the output matrices for velocity and displacement. The output matrix $C_s \in R_{l \times N}$ for the strain where l is the number of strains is introduced by multiplying Equations (6.21) and (6.23) by the transform matrix T in Equation (6.5) to obtain

$$T C_d B = C_s B = 0 \tag{6.28}$$

$$T D_a = T C_d A B = C_s A B \tag{6.29}$$

Equations (6.24) ~ (6.29) can be combined in a simple form as

$$H_{pm} B = L_h D_a \tag{6.30}$$

where $H_{pm} \in R_{2(m+l) \times N}$ and $L_h \in R_{2(m+l) \times m}$ are given by

$$H_{pm} = \begin{bmatrix} C_c A^{1-p} \\ C_s A \\ C_c A^{-p} \\ C_s \end{bmatrix} \text{ and } L_h = \begin{bmatrix} I_{m \times m} \\ T_{l \times m} \\ 0_{m \times m} \\ 0_{l \times m} \end{bmatrix} \tag{6.31}$$

where p=0, 1, and 2 for displacement, velocity and acceleration, respectively, and C_c is the corresponding output matrix. The input matrix B in Equation (6.30) is then written as:

$$B = H^{+}_{pm} L_h D_a \tag{6.32}$$

where H^{+}_{pm} is the pseudo inverse (Penrose 1955) of H_{pm}.

The flexibility matrix can be determined (Bernal and Gunes 2004) as

$$F_p = -C_p A_p^{-(p+1)} B_p \tag{6.33}$$

The combined flexibility matrix is:

$$F = -\begin{bmatrix} C_c A^{-(p+1)} \\ C_s A^{-1} \end{bmatrix} \begin{bmatrix} B & BT^T \end{bmatrix} = Q_m D_m \tag{6.34}$$

where:

$$Q_m = -\begin{bmatrix} C_a A^{-(p+1)} \\ C_s A^{-1} \end{bmatrix} H^{+}_{pm} L_h \tag{6.35}$$

$$D_m = D_a \begin{bmatrix} I & T^T \end{bmatrix} \tag{6.36}$$

Subtracting the flexibility matrices from the damaged and undamaged states,

$$F_d - F_u = Q_{m,d} D_{m,d} - Q_{m,u} D_{m,u} \tag{6.37}$$

where the subscripts d and u denote damaged and undamaged states. Introducing $\Delta D_m = D_{m,d} - D_{m,u}$, $\Delta Q_m = Q_{m,d} - Q_{m,u}$, and $\Delta F = F_d - F_u$, Equation (6.37) becomes

$$\Delta F = \Delta Q_m D_{m,u} - Q_{m,d} \Delta D_m \tag{6.38}$$

Performing the singular value decomposition, ΔD_m can be written as

$$\Delta D_m = \begin{bmatrix} U_1 & U_2 \end{bmatrix} \begin{bmatrix} s_1 & 0 \\ 0 & s_2 \end{bmatrix} \begin{bmatrix} Z_1^T \\ Z_2^T \end{bmatrix} \tag{6.39}$$

Assuming that the singular value s_2 is negligible, post-multiplying Z_2 to Equation (6.38) yields:

$$\Delta F \cdot Z_2 \simeq \Delta Q_m D_{m,u} Z_2 \tag{6.40}$$

Taking the transpose of Equation (6.40) leads to the fundamental expression for the stochastic DLV method:

$$\left(Z_2^T\right) \Delta F \simeq \left(Z_2^T D_{m,u}\right) \Delta Q_m^T \tag{6.41}$$

Because most vectors in the null space of $\Delta Q_m{}^T$ will be in the near null space of the change in the flexibility matrix (Bernal 2006), Q_m can be employed as a substitute for the flexibility matrix. The singular value decomposition of $\Delta Q_m{}^T$ leads to the DLVs.

6.2.3 DLV from multimetric data

The damage locating vectors obtained using the singular value decomposition consist of two parts corresponding to displacement and strain. The combined damage locating vector $\{L_c\}$ is written as:

$$\{L_c\} = \begin{bmatrix} \{L_1\} \\ \{L_2\} \end{bmatrix} \tag{6.42}$$

where $\{L_1\}$ and $\{L_2\}$ are the load vectors for displacement and strain, respectively. To apply $\{L_2\}$ to the structural model, $\{L_2\}$ needs be converted as $\{L_n\}=T^T\{L_2\}$. Note that the load vectors $\{L_1\}$ and $\{L_n\}$ are not necessarily applied at co-located or separately located nodes; they can share none, some, or all of the nodes. Applying the load vectors, the normalized accumulated stress can be obtained using Equation (6.2).

If only strain measurements are used in damage detection, the stochastic DLV method requires the strain in the damaged element to be measured. For statically determinate structures, the pseudo element force $\{L_2\}$ is equal to the element force, resulting in zero stress at the unmeasured elements regardless of the damage status of the elements. For redundant structures, although the stress induced by $\{L_2\}$ is redistributed over the adjacent elements, the redistributed stress is generally not significant when compared to the stress induced by $\{L_2\}$. Due to the small stress, the damage localization in an unmeasured element is difficult in practice. This limitation on the use of strain measurement for damage detection can be resolved by using acceleration along with strain, as will be demonstrated in the later section.

6.3 Numerical validation

6.3.1 Overview of simulation

Numerical simulation is conducted to verify the efficacy of the proposed extension of the stochastic DLV method using multimetric data. Consider the 53 degree-of-freedom (DOF) planar truss model shown in Figure 6.1. Damage is simulated by 10% and 40% stiffness reductions in element 16. Independent band-limited white noise forces are applied as the input excitation in both horizontal and vertical directions at all nodes. Vertical and horizontal accelerations are measured at nodes 5~7 and 19~21 and the strain is measured at elements 16~25. Gaussian measurement noise with a bandwidth up to the Nyquist frequency and an amplitude of 5% RMS the corresponding measured signal is added to the acceleration and strain. Because these nodes and elements are in the fourth and fifth bays of the truss model, only the elements in these bays (elements 16~25) are considered in damage detection (Gao *et al.* 2005). Note that the structural model is internally statically indeterminate because of these bays.

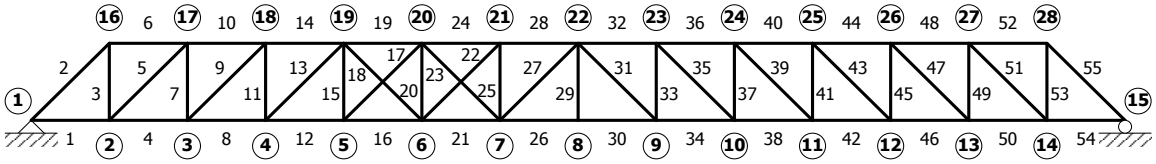


Figure 6.1 53 DOF planar truss model.

Simulation cases in Table 6.1 are considered for both the 10% and 40% damage: (a) Case 1: strain-only, (b) Cases 2~5: acceleration-only, (c) Cases 6~10: multimetric data when strain in the damaged element is measured, and (d) Cases 11~15: Multimetric data without measuring strain in the damage element. Each simulation case is repeated 500

times independently to account for the statistical nature of the problem while investigating the accuracy and robustness of the proposed method.

The mean and standard deviation of the normalized accumulated stress for cases 1 (strain-only), 3 (acceleration-only), and 10 (multimetric) are shown in Figure 6.2. As can be seen in these figures, the normalized accumulated stress in element 16 is considerably smaller than other elements of interest. Therefore, the stochastic DLV method identifies element 16 as a damage candidate in these cases. The normalized accumulated stress at element 16 for Case 10 (*i.e.*, multimetric data) has a smaller mean value and less variation than that for Case 2 (*i.e.*, acceleration-only). Although Case 1 (*i.e.*, strain-only) has the smallest stress at element 16, the strain in the damaged element should be measured, as previously illustrated. As such, multimetric data is seen to reduce the normalized accumulated stress at damage location, as well as facilitate the use of strain in the stochastic DLV method. More detailed discussion follows in the remainder of this section.

Table 6.1 Simulation cases.

Case	Acceleration		Strain		
	Number of measurements	Node	Number of measurements	Element	Strain in the damaged element measured?
1	0	N/A	11	15~25	Y
2	10	5~7, 19, 20	0	N/A	N
3	12	5~7, 19~21	0	N/A	N
4	14	4~7, 19~21	0	N/A	N
5	16	4~7, 18~21	0	N/A	N
6	12	5~7, 19~21	1	16	Y
7	12	5~7, 19~21	3	16, 23, 24	Y
8	12	5~7, 19~21	6	16, 20, 22~25	Y
9	12	5~7, 19~21	8	16, 18, 20~25	Y
10	12	5~7, 19~21	11	15~25	Y
11	12	5~7, 19~21	1	17	N
12	12	5~7, 19~21	3	15, 17, 19	N
13	12	5~7, 19~21	6	15, 17~21	N
14	12	5~7, 19~21	8	15, 17~22, 25	N
15	12	5~7, 19~21	10	15, 17~25	N

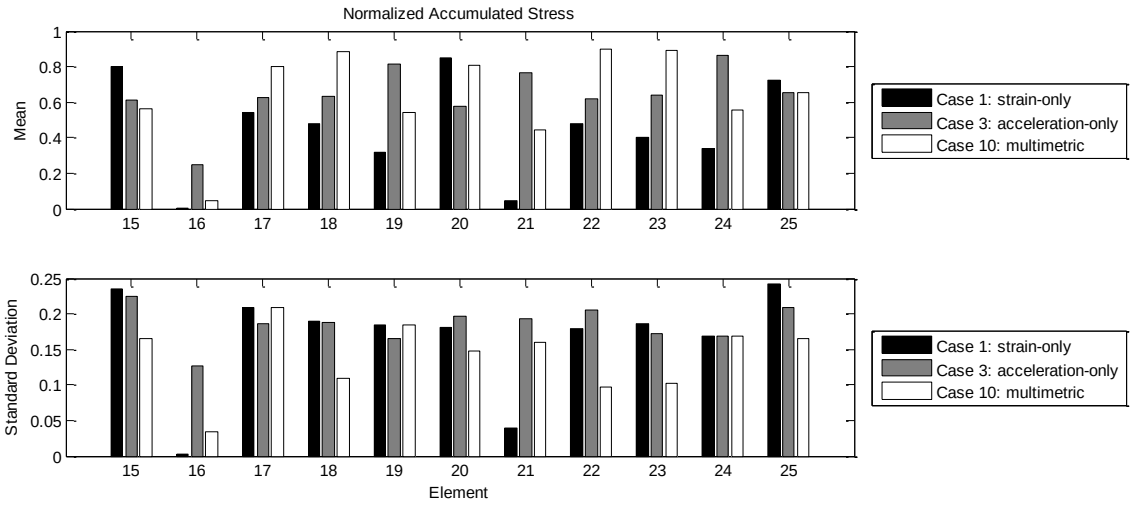
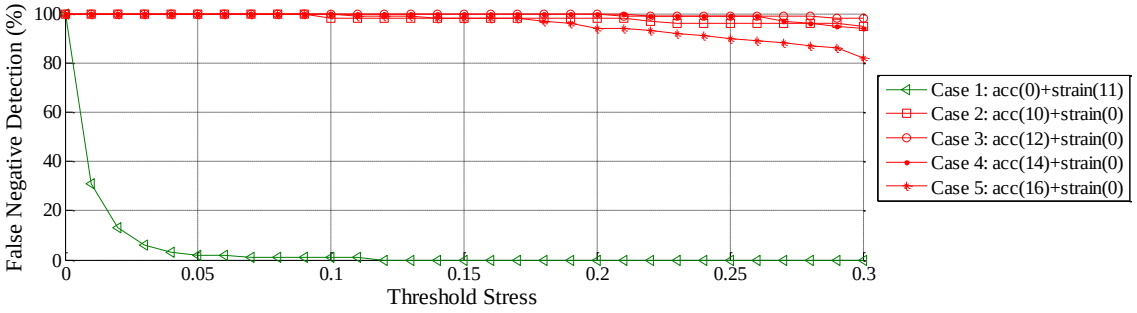


Figure 6.2 Mean and standard deviation of normalized accumulated stress (40% stiffness reduction in element 16).

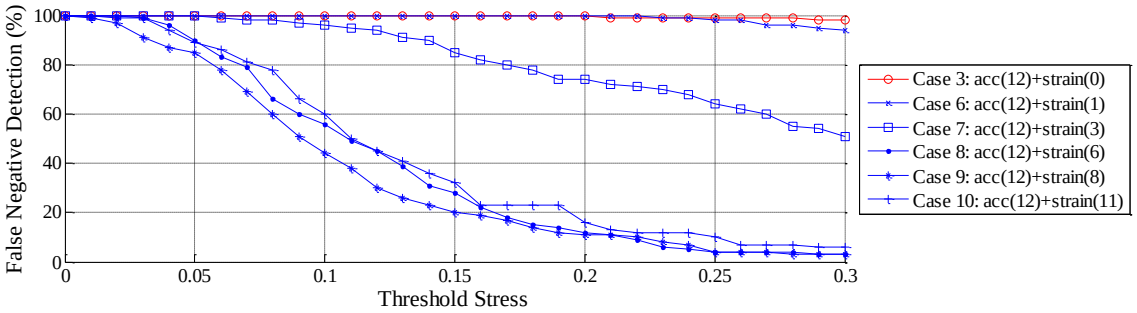
6.3.2 Performance evaluation: false negative and positive detections

False negative and positive detections of damage are considered to demonstrate the efficacy of the proposed multimetric sensing approach. A false negative detection occurs when the algorithm cannot detect a damaged element, *i.e.*, reports it as undamaged. A false positive detection occurs when the algorithm reports an undamaged element as damaged. From the structural engineering perspective, the false negative detection is more critical, because undetected damaged elements may have severe consequences, even resulting in structural collapse. On the other hand, false positive detections can needlessly heighten concern about a safe structure. Note that in the DLV method the zero stress does not necessarily mean damage; the term 'damage candidate' is used. Practically, when damage candidate elements are identified based on the threshold stress, further actions should be taken to verify damage and repair if necessary. Thus, the false positive detection is defined to evaluate how often false *damage candidates* are identified. For the simulations reported herein, the numbers of false negative and positive detections are tracked.

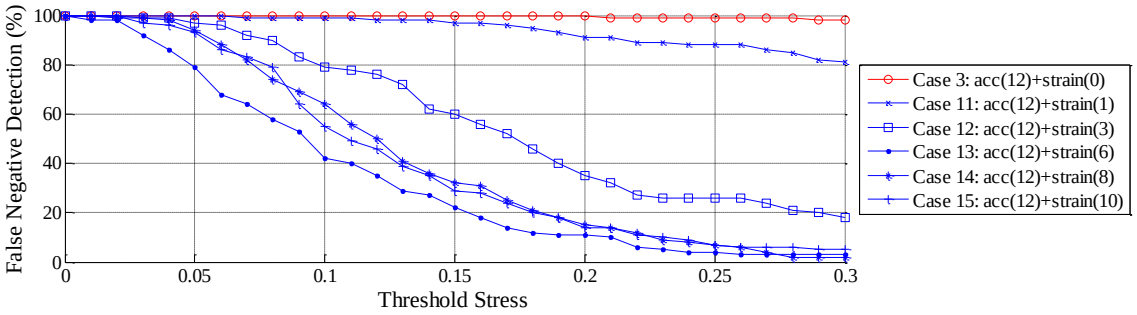
Figure 6.3 shows the percentage of false negative detections for 10% stiffness reduction for the cases (a) only strain or only acceleration, (b) multimetric data when strain in the damaged element is measured, and (c) multimetric data when strain in the damaged element is not measured. Compared to the acceleration-only cases (Cases 2~5), the false detection rate is drastically reduced if multimetric data is employed. In particular, the false detection of all acceleration-only cases is nearly 100% (see Figure 6.3a), which is shown to be significantly reduced by the use of multimetric data (see Figure 6.3b, c). Note that the strain-only case (Case 1) has a smaller number of false negatives than the acceleration-only cases (see Figure 6.3a). However, the strain-only case has a critical limitation, as described in the previous section, that strain in the damaged element must be measured to localize damage.



(a) Strain and acceleration employed independently.



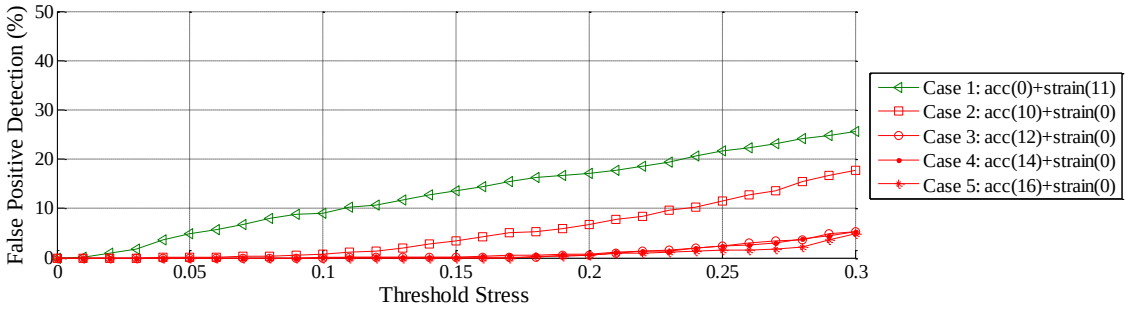
(b) Multimetric data when strain in the damaged element is measured.



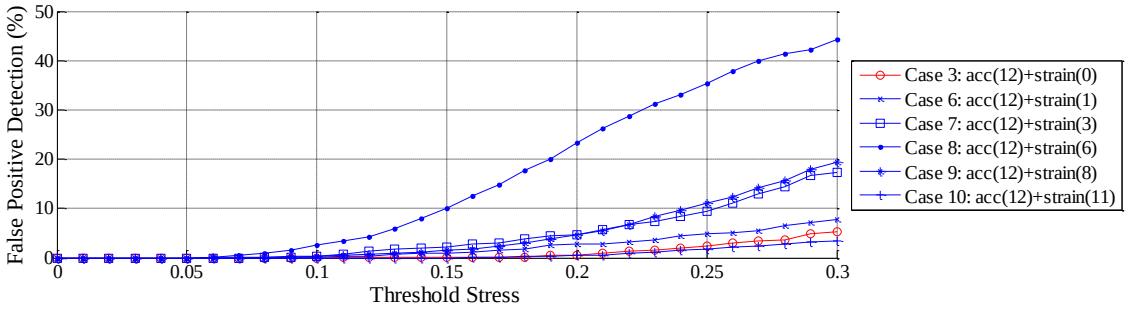
(c) Multimetric data without measuring strain in the damage element.

Figure 6.3 False negative detection (10% stiffness reduction).

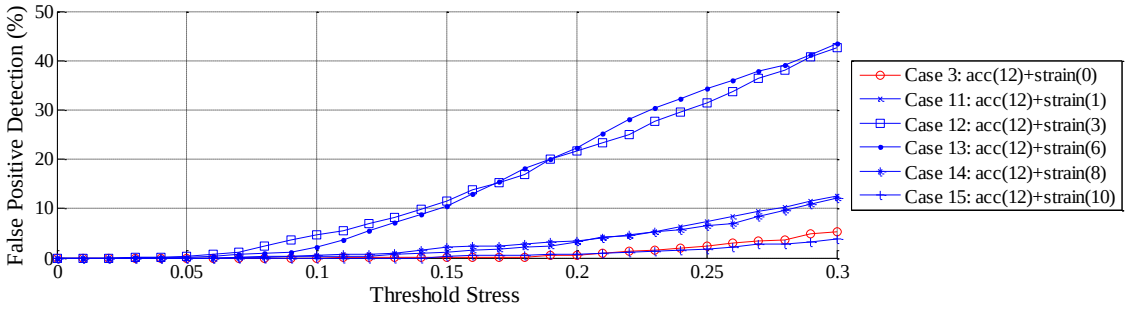
While multimetric data can reduce the false negative detection, multimetric data does not always guarantee a smaller number of false positives than for the case when strain or acceleration is used separately. In cases 8, 11, and 12 of Figure 6.4, the number of false positive detections is greater than for the case when only accelerations were used (*e.g.*, Case 4). Because false negative detection is more critical than false positive detection, use of multimetric data is considered to be more effective than use of either strain or acceleration separately. By selecting a proper threshold stress, both the false positive and negative detections can be reduced to an acceptable level.



(a) Strain and acceleration employed independently.



(b) Multimetric data when strain in the damaged element is measured.

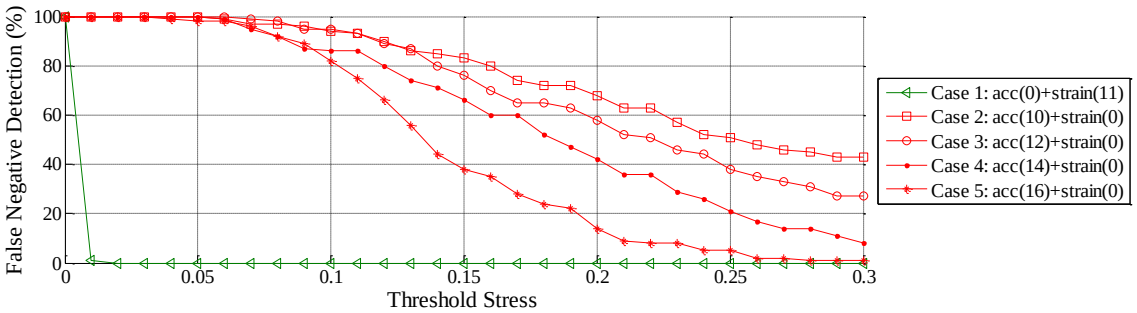


(c) Multimetric data without measuring strain in the damage element.

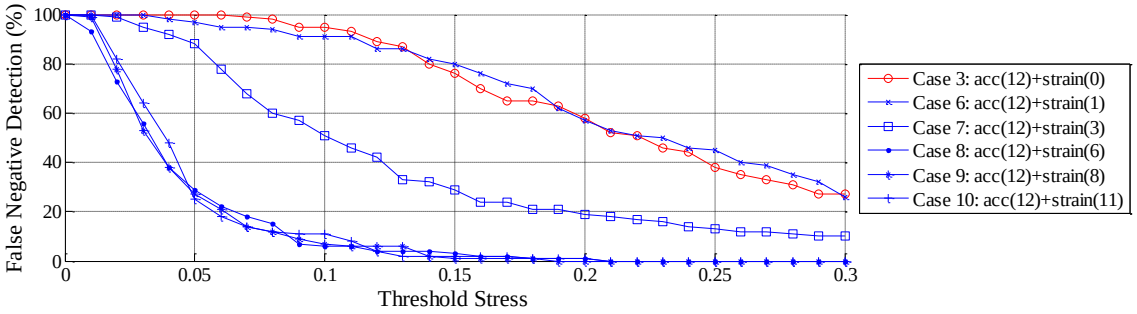
Figure 6.4 False positive detection (10% stiffness reduction).

The 40% stiffness reduction case shows the same trend as in the 10% case. The false negative detection rates are significantly reduced when the multimetric data is employed as can be seen in Figure 6.5 while the false positive detection rates tend to increase in Figure 6.6.

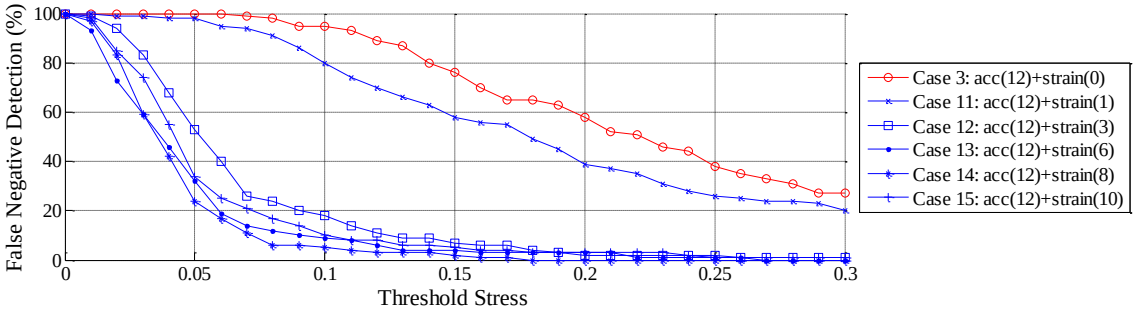
Note that when strain at the damaged element is not measured (Case 11~15), the numbers of false negative detections in Figure 6.3c and Figure 6.5c are somewhat greater than for the cases when it is measured (Case 10~13) in Figure 6.3b and Figure 6.5b, showing seemingly better performance. However, the damage decision for non-measured elements becomes less reliable, because the DLV obtained using strain data is applied only to the measured elements, making the cumulative stress at the unmeasured elements smaller when normalized. Thus, unmeasured elements have a higher chance of false positive detections as shown in Figure 6.4 and Figure 6.6. Although the smaller number of false negative detections is considered normally better, false positive detection also should be taken into account to appropriately assess structural damage.



(a) Strain and acceleration employed independently.

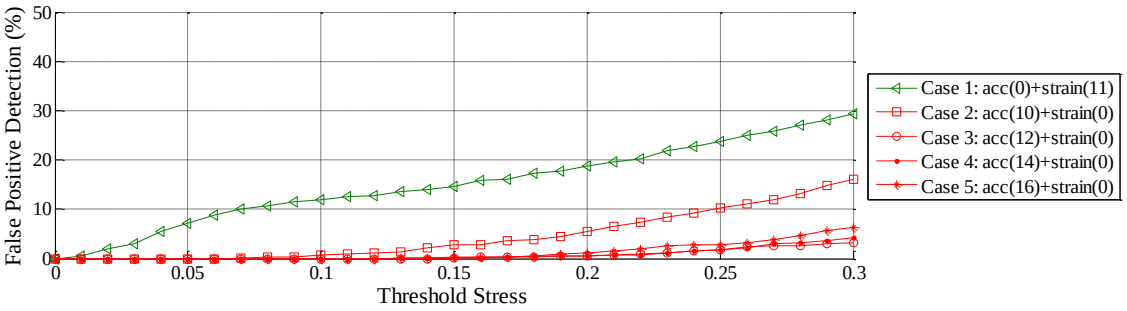


(b) Multimetric data when strain in the damaged element is measured.

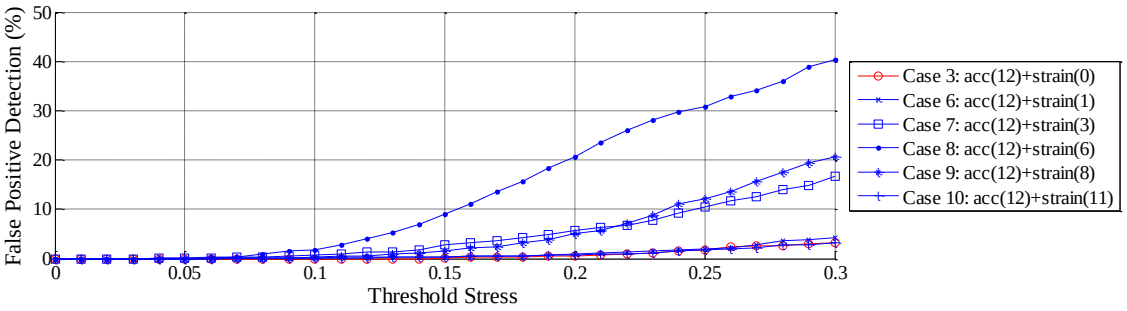


(c) Multimetric data without measuring strain in the damage element.

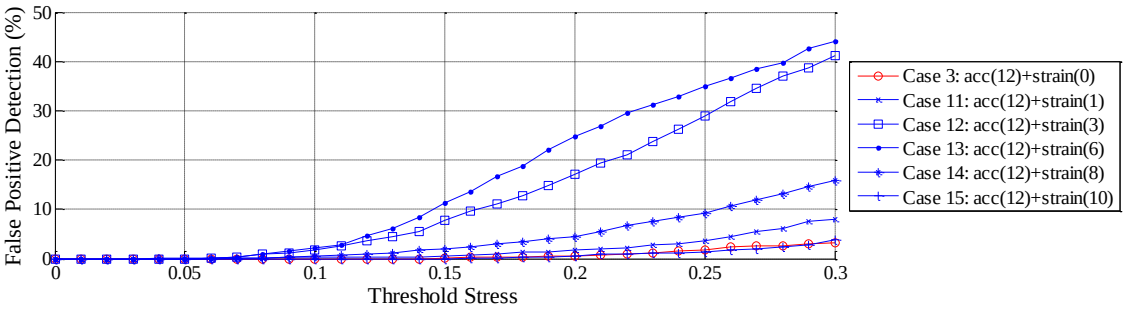
Figure 6.5 False negative detection (40% stiffness reduction).



(a) Strain and acceleration employed independently.



(b) Multimetric data when strain in the damaged element is measured.



(c) Multimetric data without measuring strain in the damage element.

Figure 6.6 False positive detection (40% stiffness reduction).

Damage detection results from numerical simulation have been presented in this section to evaluate the performance of the proposed approach utilizing multimetric data for structural damage detection. Two different damage severity cases (10% and 40% stiffness reductions at a horizontal element) have been considered. Damage at a diagonal element has produced the similar trend as in the horizontal element although it is not presented herein due to space limitations. From the numerical study, multimetric data has been shown to improve the performance of structural damage detection.

6.4 Summary

This chapter described a damage detection approach based on the stochastic DLV method using the multimetric data. A symmetric strain flexibility matrix was introduced and the stochastic DLV method was modified to accommodate acceleration and strain measurements simultaneously. The damage detection approach was validated by extensive numerical simulation using a planar truss model.

The damage detection strategy based on the multimetric data outperformed the stochastic DLV method that uses only a single type of data. In particular, the limitation of the strain-only case that the strain in the damaged element needs to be measured was resolved with the aid of acceleration. This result shows the multimetric damage detection strategy is promising for SHM. The next chapter illustrates full-scale experimental validation of decentralized approaches described in this study.

FULL-SCALE VALIDATION

The decentralized approaches, implemented on the Imote2 sensor platform using the ISHMP Services Toolsuite, are experimentally validated in this chapter using two full-scale testbeds: (i) the Irwin Indoor Practice Field in the University of Illinois at Urbana-Champaign, and (ii) the Jindo Bridge, a cable-stayed bridge located in South Korea. Robustness of the WSSN should be proven in the harsh environment from hardware and software aspects to validate the applicability of smart sensors to monitoring civil infrastructure. The nature of the spatially distributed WSSN over wide areas provided by these test beds is well-suited to demonstrating the efficacy of the decentralized approaches.

7.1 Decentralized modal analysis at the football field

7.1.1 Irwin Indoor Practice Field

The decentralized modal analysis is applied in the filed testing at the Irwin Indoor Practice Facility located on the campus of the University of Illinois at Urbana-Champaign (see Figure 7.1). The facility, which embraces a reduced-size football field inside, has a semi-parabolic dome structure with an arched box truss spanning the length of the football field as shown in Figure 7.1b. The box truss is approximately 94.5 m in length and reaches up to an internal height of 15 m. The large scale of the facility provides a unique test bed for decentralized modal analysis using wireless smart sensors.

(a) Outside. (b) Inside.

Figure 7.1 Irwin Indoor Practice Facility.

7.1.2 Sensor deployment

A total of 14 Imote2 sensor nodes are prepared as shown in Figure 7.2. Each sensor node consists of an Imote2, a SHM-A sensor board, an external antenna, 3 D-Cell batteries, an enclosure with an aluminum plate bolted to the bottom. The Imote2 nodes are installed at the center truss of the football field (see Figure 7.3). A lift truck is used to reach the ceiling (see Figure 7.4) and clamp the sensor nodes as shown in Figure 7.5. All nodes including the gateway and sensor nodes have antennas vertically oriented to ensure wireless communication.

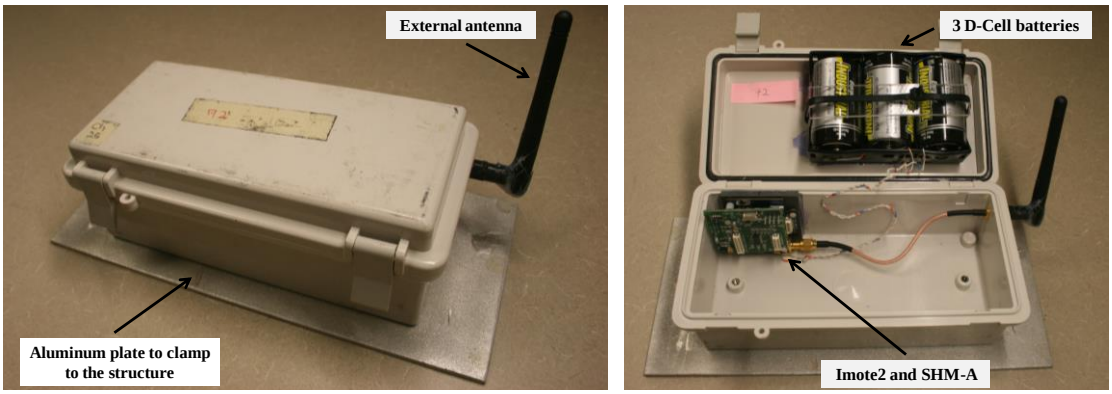


Figure 7.2 Imote2 sensor node assembly.

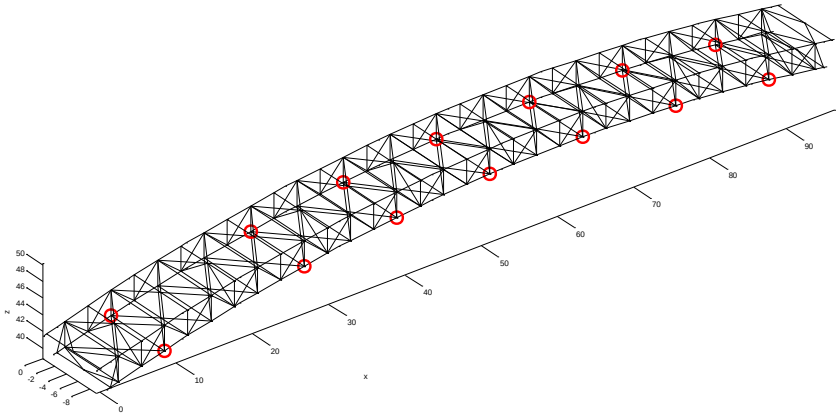


Figure 7.3 Sensor locations indicated as circles.

Figure 7.4 Lift truck for sensor installation.

Figure 7.5 Imote2 sensor node attached to the center truss.

The effort for the sensor installation can be significantly reduced due to the nature of the wireless smart sensors. Most of the sensor deployment time (about 6 hours) is spent on the lift truck operation as clamping the sensor nodes to the structure is quickly done. If the centralized data acquisition system with wired sensors is employed in the experiment, wiring all sensor nodes distributed over the 94 m-long truss at the height of 15 m from the ground should be costly and challenging. As such, the WSSN provides a convenient means for dynamic testing.

7.1.3 Decentralized data acquisition and processing

DecentralizedDataAggregation is utilized to obtain correlation functions as in the laboratory experiment previously described. The network consists of 4 local groups, each of which has a cluster-head (S6, S10, and S9 for groups 1, 2, and 3, respectively) and 5 leaf nodes. Ambient accelerations in the vertical direction are measured at each sensor node, and correlation functions are subsequently calculated on the Imote2 sensor nodes with respect to reference data measured at cluster-heads. Then, the cluster-heads send the group's correlation functions to the gateway node to save in the base station. Figure 7.7 shows two typical correlation functions estimated in group 2. Parameters for sensing and data processing are summarized as:

- Sampling frequency: 100 Hz
- Cutoff frequency: 50 Hz
- Data channel: Vertical
- Number of FFT: 1024
- Number of overlaps between windows: 512
- Number of averages: 10
- Detrended

Note that the correlation functions can be retained in each cluster-head rather than collecting in the base station if further in-network processing such as damage detection or system identification is desired.

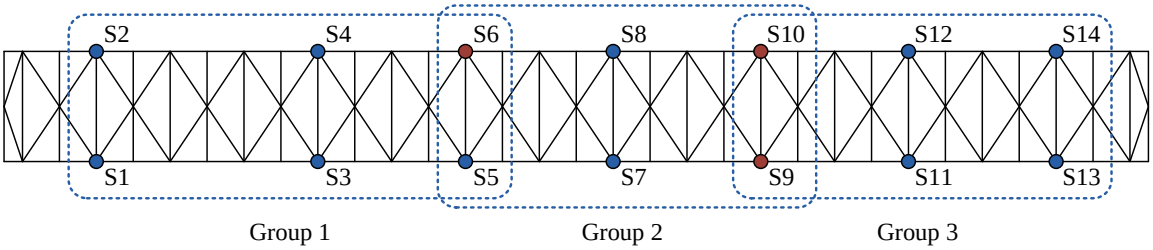


Figure 7.6 Sensor topology at the center truss (S6, S10 and S9 are cluster-heads of Groups 1, 2, and 3, respectively).

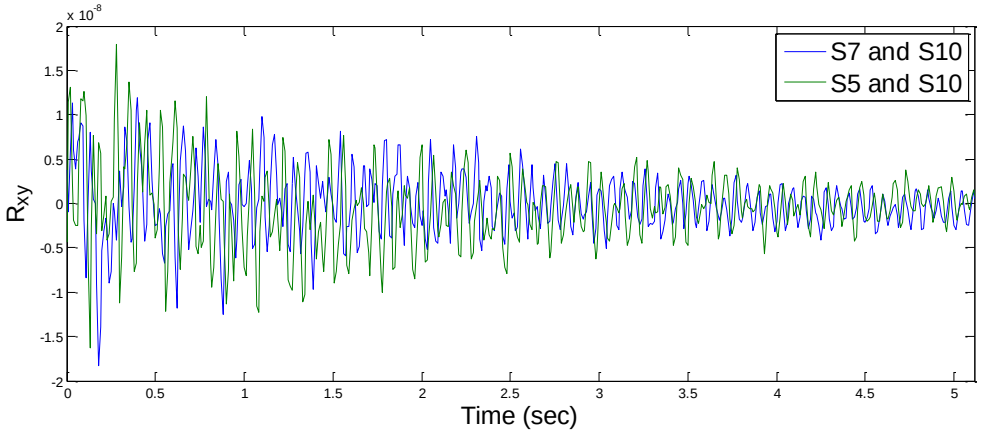
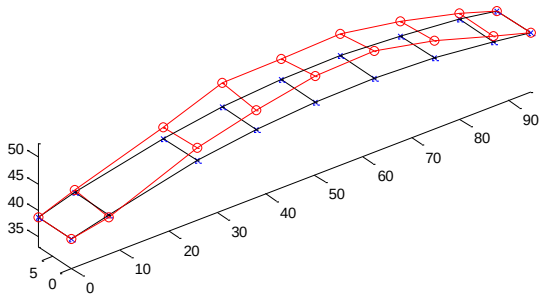
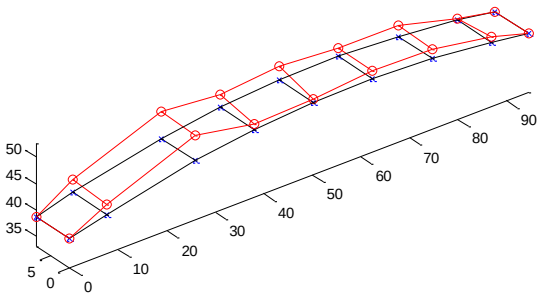


Figure 7.7 Correlation functions $R_{S7,S10}$ and $R_{S5,S10}$ in Group 2.

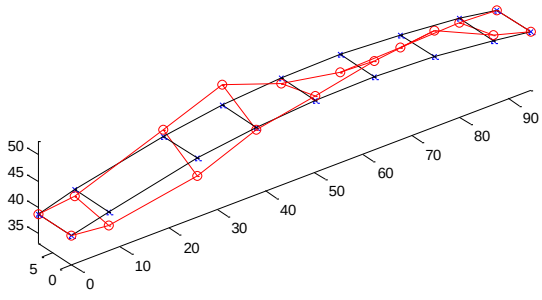
Global modal properties are estimated in the base station using the collected correlation functions as shown in Figure 7.8. The identified modes correspond to the peaks found in cross spectra between the reference accelerations from the cluster heads and accelerations from leaf nodes. Taking the inverse FFT of the correlation functions estimated at each group, the cross spectra are calculated. The absolute value of the cross spectra is normalized and averaged for each group as shown in Figure 7.9. The peaks at the identified frequencies are commonly found in all three groups, supporting that the identified frequencies indicate true modes.



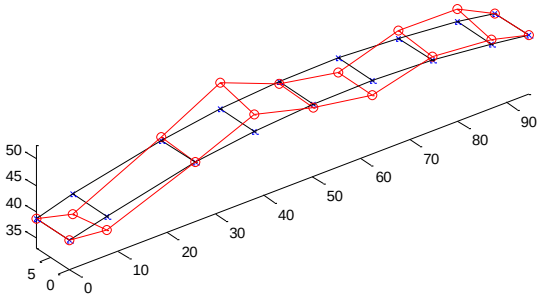
(a) 2.2137 Hz



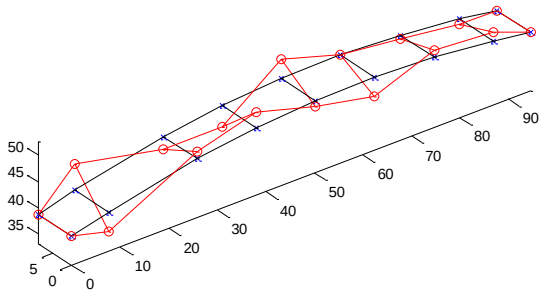
(b) 2.8351 Hz



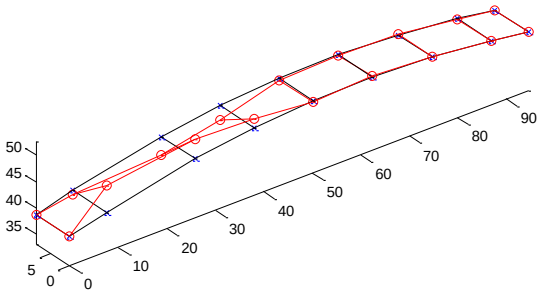
(c) 5.6155 Hz



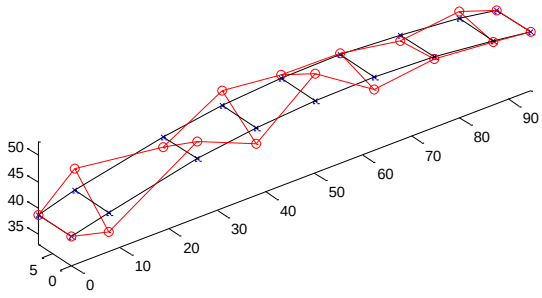
(d) 7.8907 Hz



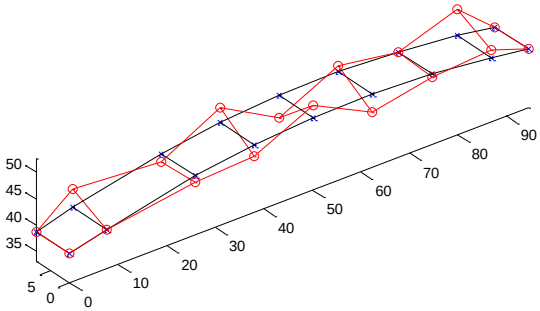
(e) 11.6111 Hz



(f) 15.7276 Hz



(g) 23.3831 Hz



(h) 29.6967 Hz

Figure 7.8 Global mode shapes and corresponding natural frequencies.

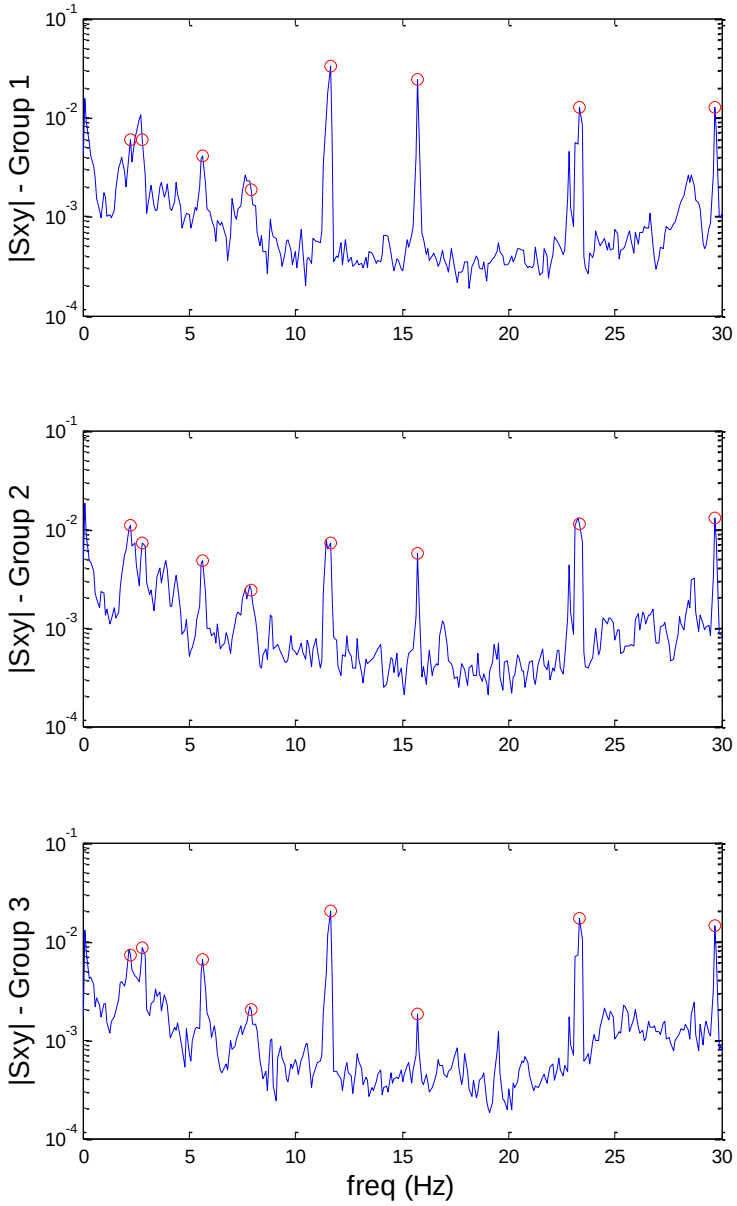


Figure 7.9 Absolute value of cross spectrum (circles represent identified frequencies).

7.1.4 Summary

Decentralized modal analysis was applied to identify global modal properties of the Irwin Indoor Practice Field using the network of Imote2 smart sensors. *DecentralizedData-Aggregation* for sensing and in-network data processing could successfully collect correlation functions decentrally estimated in each subdivided groups. The global modal properties were identified from the correlation functions at the base station. Furthermore, the WSSN was seen to be quite efficient in the short term dynamic testing, significantly reducing the installation efforts. The next section will discuss the efficacy of the decentralized approaches for the WSSN from the long-term monitoring perspectives.

7.2 Jindo Bridge deployment

The second year WSSN deployment at the Jindo Bridge is part of a collaborative project between South Korea (KAIST; Seoul National University), Japan (University of Tokyo) and the USA (University of Illinois at Urbana-Champaign; Texas Tech University). In the first year of the project, a total of 70 Imote2 sensor nodes with two base stations were deployed to realize the first large-scale, autonomous WSSN for SHM (Rice *et al.* 2010; Jang *et al.* 2010a; Cho *et al.* 2010b; Nagayama *et al.* 2010). The deployment focused on demonstrating the performance and applicability of the Imote2 sensor platform and the ISHMP software in the full-scale test bed in a harsh environment. The research effort has been continued to the second year to realize a larger, autonomous, power-harvesting network of sensors with decentralized data processing strategies as well as the centralized data acquisition. The primary goals of the deployment are as follows:

- Realize the power-harvesting network of sensors for long-term monitoring
- Validate the performance of the high-sensitivity sensor board (SHM-H) developed at the University of Illinois at Urbana-Champaign (Jo *et al.* 2010)
- Validate the capability of autonomous operation of the WSSN
- Validate the applicability of decentralized modal analysis to bridge structures
- Monitor tension forces of the cables and validate associated software applications
- Validate the multi-hop communication protocol of the ISHMP Service Toolsuite

After providing an overview of the Jindo Bridge deployment, this section will describe the decentralized approaches in WSSN developed in this study.

7.2.1 Jindo Bridges

The Jindo Bridges pictured in Figure 7.10 are twin cable-stayed bridges constructed in 1984 (the 1st Jindo Bridge) and 2005 (the 2nd Jindo Bridge) to connect Jindo Island and the town of Haenam, located at the southeastern part of the Korean peninsula. The 2nd Jindo Bridge, on the left in Figure 7.10, is the test bed in the WSSN deployment. 'Jindo Bridge' indicates the 2nd Jindo Bridge in the rest of this document. The Jindo Bridge features three continuous spans (344 m of mid span, 70 m of each side span) and a total of 60 steel cables that support the bridge deck (see Figure 7.11). As the bridge has no

Figure 7.10 Jindo Bridges (1st: right, 2nd: left).

structural part under the sea, the bridge scour is not a concern while it could be a serious problem due to the strong tidal current. Instead, wind- or traffic- induced vibration can be a potential threat to structural health for this lightly damped structure.

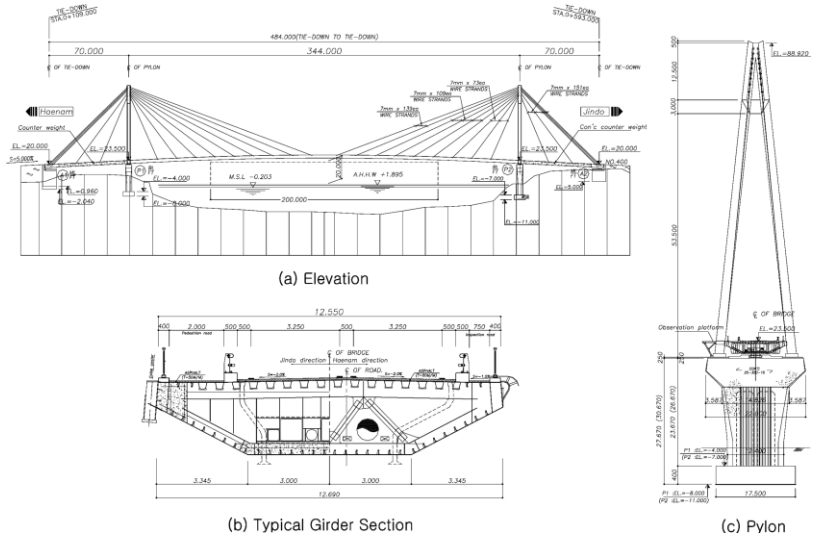


Figure 7.11 Drawing of the 2nd Jindo Bridge.

7.2.2 WSSN deployment

The WSSN deployed on the Jindo Bridge, consisting of 113 Imote2 sensor nodes (see Figure 7.12), is the world's largest network of smart sensors for SHM to date. To efficiently operate the large network, the WSSN is divided into 4 subnetworks controlled by 2 base stations (Jindo base station at the Jindo side and Haenam base station at the Haenam side). Each base station manages two networks, each of which is for cable and deck. The subnetworks are summarized in Table 7.1. This section describes the installation of the WSSN components, i.e., gateway and sensor nodes and base stations.

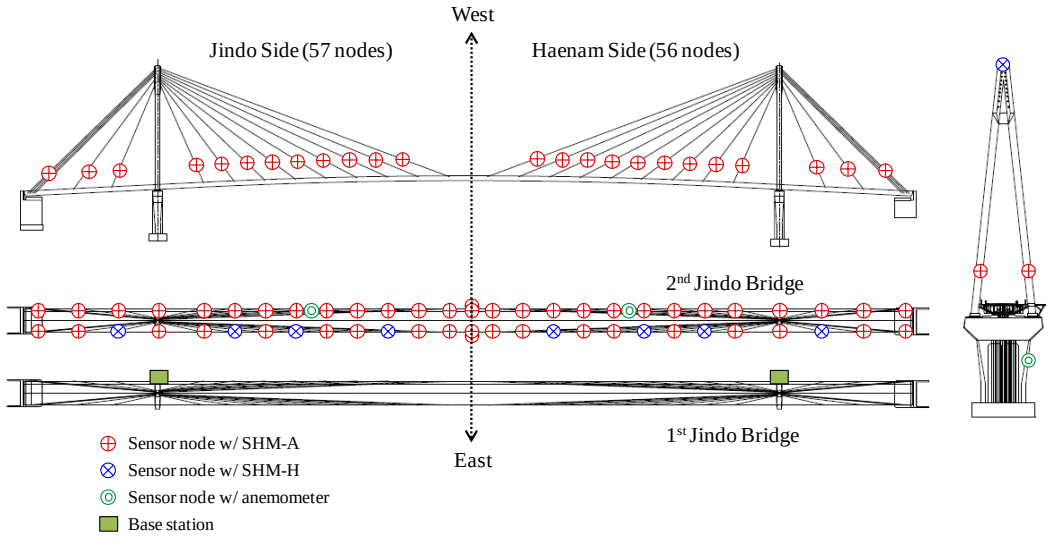


Figure 7.12 Sensor deployment.

Table 7.1 Subnetworks of the WSSN at the Jindo Bridge.

	Haenam base station				Jindo base station				Total	
	Deck Haenam		Cable Haenam		Deck Jindo		Cable Jindo			
Nodes	31		26		30		26		113	
Accelerometer	29		26		29		26		110	
SHM-A	24	SHM-H 5	26	0	24	5	26	0	100	10
Anemometer	2		0		1		0		3	
Solar-powered	30		26		30		26		112	
Wind-powered	1		0		0		0		1	
Communication protocol	Single-hop		Single-hop		Single-hop		Multi-hop		N/A	

Sensor node

In total, 113 sensor nodes (100 nodes with SHM-A, 10 nodes with SHM-H, and 3 nodes with anemometers) have been deployed on the Jindo Bridge (see Figure 7.12 and Table 7.1Figure 7.12 Sensor deployment.). To achieve long-term monitoring, all sensor nodes have equipped with solar panels and rechargeable batteries as wells as environmentally hardened enclosures. The sensor node shown in Figure 7.13 consists of an Imote2, either SHM-A or SHM-H sensor board, a rechargeable battery, an external antenna, and an enclosure. As opposed to the 1st year's deployment that mostly employed 3 D-cell batteries per node, a rechargeable battery has been adopted with solar panels (see Figure 7.14). Two magnets are bolted down to the bottom of the enclosure for nodes to be attached to the steel deck of the bridge.

The SHM-H requires a different configuration as shown in Figure 7.13. The SHM-H has been designed based on the SHM-A with an increased sensitivity with low measurement noise for the vertical acceleration while sacrificing the cost effectiveness (Jo *et al.* 2010). Because the measurement range of the SHM-H has been selected as 0.8g – 1.2g for the vertical acceleration, the SHM-H should be placed upward so that SHM-H measures 1g. As the sensor nodes with SHM-H have been deployed underneath the deck upside down, the SHM-H is attached to the lid of the enclosure as shown in Figure 7.13.

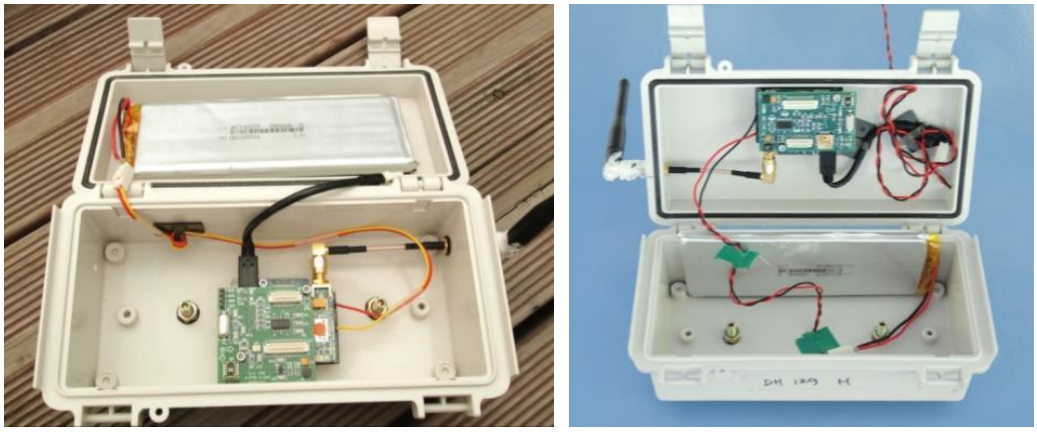


Figure 7.13 Sensor nodes with a SHM-A (left) and a SHM-H (right) sensor board.

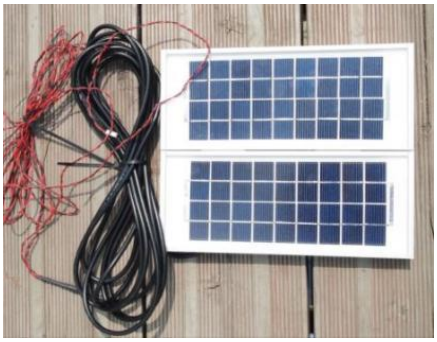


Figure 7.14 Solar panel.

Figure 7.15 shows how the sensor nodes with solar panels and power cables are arranged underneath the deck. The sensor nodes are attached to the steel deck using the magnets bolted to the bottom of the enclosure. Two solar panels are installed on the wind fair of the bridge to allow the direct sun light on the solar panels for more efficient power harvesting. Sensor nodes and solar panels are connected with power cables shown in Figure 7.14. Although the solar panels are exposed to the direct sun light for only a few hours with some angle in this configuration, the power harvesting of the sensor nodes has shown to be enough for the operation of the WSSN.

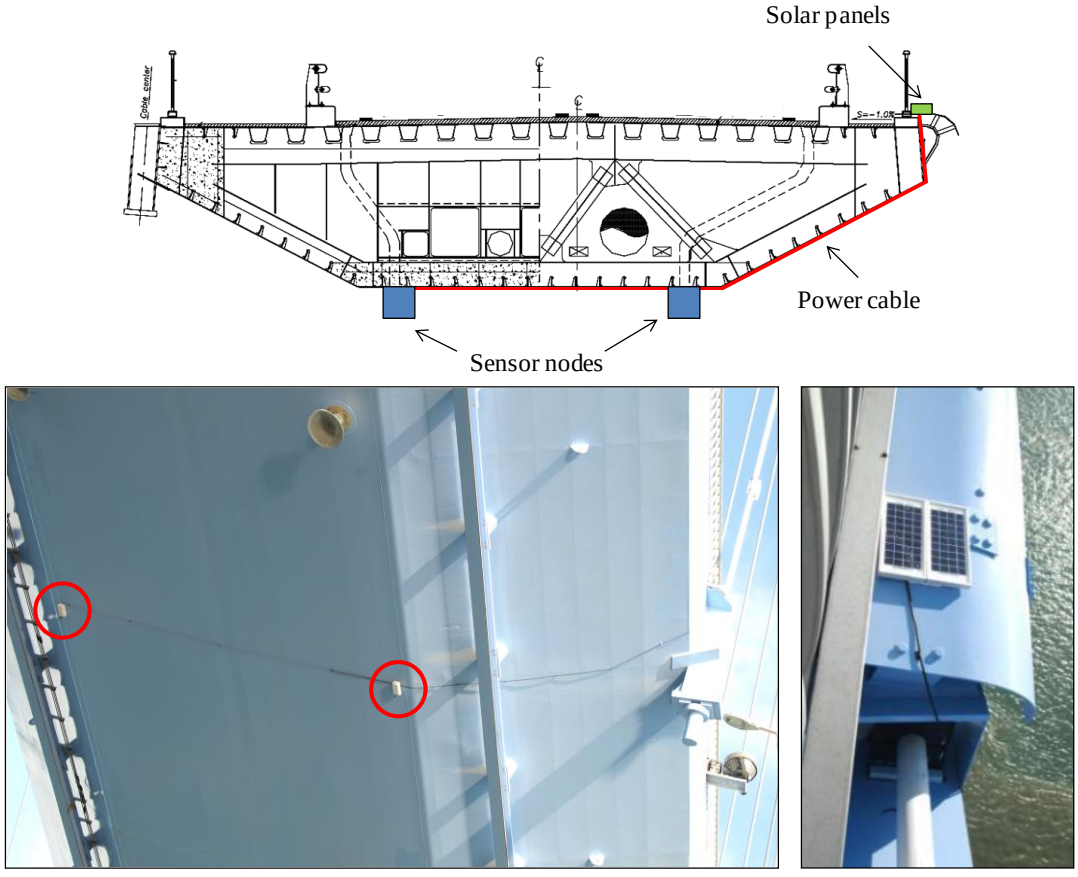


Figure 7.15 Sensor deployment: schematic view of installation (top), sensor nodes with power cables (bottom left), and solar panels on the wind fairing (bottom right).

For the cable node, the solar panel is directly attached on top of the enclosure as shown in Figure 7.16. A plastic plate is bolted down to the enclosure and installed on the cable using the U-shaped steel rod. In total, 52 cable nodes are installed on the cables in the same manner.

Figure 7.16 Cable node.

Base station and gateway node

The base stations, pictured in Figure 7.17, are industrial grade PC running on Windows XP with an Uninterrupted Power Supply (UPS) housed in the environmentally hardened case. The base stations can be remotely accessed from local computers via internet connection provided by an Internet Service Provider (ISP). Two gateway nodes, also enclosed in the case, are connected to the base station. To improve radio communication between the gateway nodes and sensor nodes, larger 7 dBi antennas (see Figure 7.17) with boosters are used for the gateway nodes, compared to sensor nodes' 2 dBi antennas.

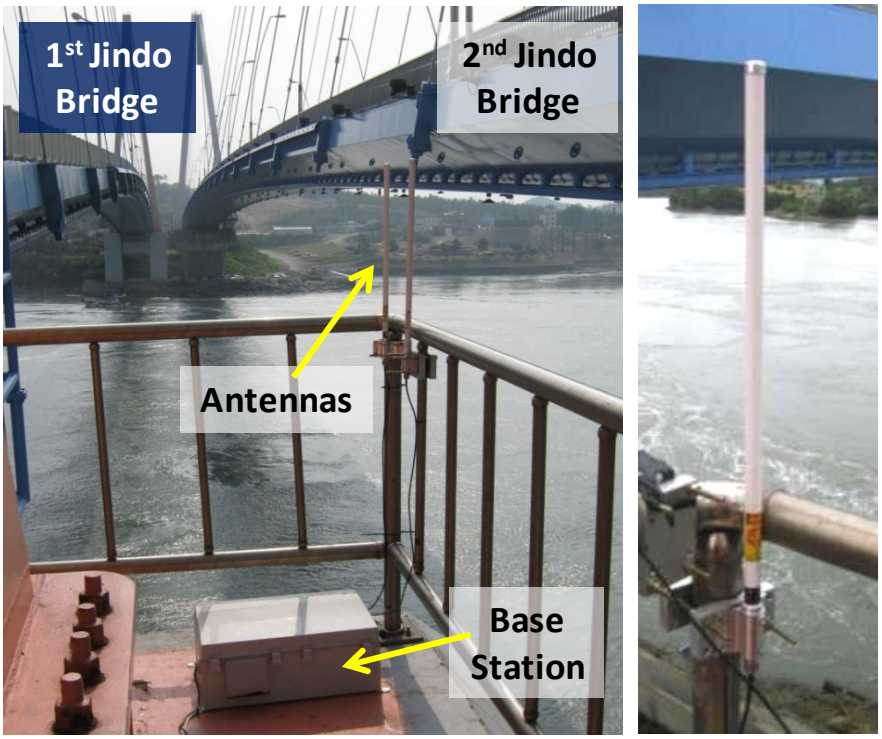


Figure 7.17 Base station and external antennas for the gateway nodes.

The base stations have been installed on the piers of the 1st Jindo Bridge as shown in Figure 7.17 and Figure 7.12. As the height of the bridge pier is lower than the deck, the antennas at the base stations and most sensor nodes under the deck are on the direct line of sight, resulting in excellent communication. However, the cables nodes on the west side of the bridge (the far side from the base station) cannot be seen from the base station; communication with these nodes is quite degraded while the cable nodes on the east side are well responsive. With the available sensor nodes, the WSSN has autonomously operated, providing information valuable for monitoring the bridge; decentralized modal analysis and cable tension monitoring are presented in the following sections.

7.2.3 Software configuration

Software is prepared for the WSSN to autonomously conduct designated tasks such as centralize/decentralized data acquisition, monitoring the status of the WSSN, and estimation of cable tensions. Applications installed on each network component are summarized in Table 7.2. The Jindo Bridge deployment aims at long-term, autonomous monitoring that is made possible by efficient power management using sleep cycling and autonomous operation using *AutoMonitor*.

As the smart sensors are typically battery-powered, power management to prolong the operation period is utmost important. Although all sensor nodes in the Jindo Bridge deployment are equipped with solar panels for power harvesting, power management is still required as the solar panels are insufficient for a continuous operation.

The strategy adopted in the ISHMP Services Toolsuite is to utilize Imote2's deep sleep mode, in which only the clock of the processor is powered while all other components are powered down (Rice and Spencer 2009). Sensor nodes are put to the deep sleep mode for a period of time T_s and wake up for another period of time T_w. In the deep sleep mode, the sensor nodes cannot receive any packets while consuming little power. With the two time periods $T_s >> T_w$, the sensor nodes are in the deep sleep mode for most of the operation time, significantly reducing power consumption compared to the case when the sensor nodes always stay on. If the sensor nodes receive any packets during the time period T_w, the sensor nodes stay with all components power on and are ready to conduct tasks that the gateway node command. This sleep cycling is implemented as *SnoozeAlarm* in the ISHMP Services Toolsuite. For more detailed information regarding the sleep cycling, see Rice and Spencer (2009).

Another important feature of the Jindo WSSN is the autonomous operation controlled by *AutoMonitor*. The primary operations of *AutoMonitor* can be categorized as event-triggered process and periodic process. The event-triggered process utilizes ThresholdSentry that periodically wakes up designated nodes (called sentry node) to measure data (i.e., acceleration or wind velocity) and alarms the gateway node if the maximum data is greater than specified threshold values. Subsequently, AutoMonitor at the gateway node run a specified application that is either *RemoteSensing* or *DecentralizedDataAggregation* in this deployment. The periodic process runs, as the name indicates, on a regular basis according to the user-specified time interval. The periodic processes of the Jindo WSSN include *CableTensionEstimation* and *AutoCommand* for monitoring the network status and environment such as battery levels, charging status, and temperature.

Table 7.2 Applications in the Jindo WSSN.

Network component	Application/Service	Description
Base station	*Team Viewer*	A free program for Windows that allows remote control and file transfer
	autocomm	A terminal program for interfacing with the Imote2 through the Imote2 Interface Board
Gateway node	*AutoMonitor*	An Imote2 application for the gateway node that controls the sensor nodes for autonomous operation
	ThresholdSentry	A service that periodically wakes up designated sensor nodes, takes short data, and reports whether the maximum data exceeds specified thresholds or not.
	SnoozeAlarm	A service for power management. Sensor nodes sleep for a period of time and wake up for a short period time to listen a command. They stay awake if a command is received.
	RemoteSensing	An Imote2 application that implements central data acquisition
	DecentralizedDataAggregation	An Imote2 application for decentralized correlation/random decrement function estimation
	CableTensionEstimation	An Imote2 application for estimation of cable tension forces based on independent processing
	AutoCommand	A part of *AutoMonitor* for monitoring the network status and environment (voltage level, charging status, and temperature)
Deck node	*RemoteSensing*	See above
	DecentralizedDataAggregation	See above
	SnoozeAlarm	See above
	ThresholdSentry	See above
Cable node	*RemoteSensing*	See above
	CableTensionEstimation	See above
	SnoozeAlarm	See above
	ThresholdSentry	See above

Hardware and software aspects of the WSSN deployed on the Jindo Bridge were described here. In the following sections, analyses on the bridge using the collected data are presented, focusing on centralized and decentralized modal analysis and estimation of cable tensions.

7.2.4 Decentralized modal analysis

Tailored to the modal testing using the WSSN, decentralized modal analysis estimates global modal properties based on the local information, obtained using the decentralized in-network data processing. As previously mentioned, this decentralized approach is especially useful for a large sensor network, in which the central data collection becomes inefficient due to data inundation and excessive power consumption. To validate the performance in a full-scale bridge structure, decentralized modal analysis is first simulated on a local computer using centrally collected sensor data, and then conducted with the local information condensed by in-network processing on the Imote2 sensor nodes.

Simulated decentralized modal analysis with centrally collected sensor data

RemoteSensing has collected acceleration time history data from each network. As the most fundamental natural modes of the Jindo Bridge are found in the 0 – 5 Hz range (Cho *et al*. 2010), the sample rate of 25 Hz with a cutoff frequency of 10 Hz (the lowest value of the default sampling rates) is selected. Time history data, taken while typhoon Kompasu is passing nearby, is shown in Figure 7.19: maximum acceleration levels are 9.37mg, 25.64mg, 26.88mg, and 29.99mg for N26 (top left, side span), N21 (top right, main span), N17 (bottom left, main span), and N14 (bottom right, main span), respectively (see Figure 7.20 for node numbers). Due to the typhoon, the bridge is well excited with a large input excitation.

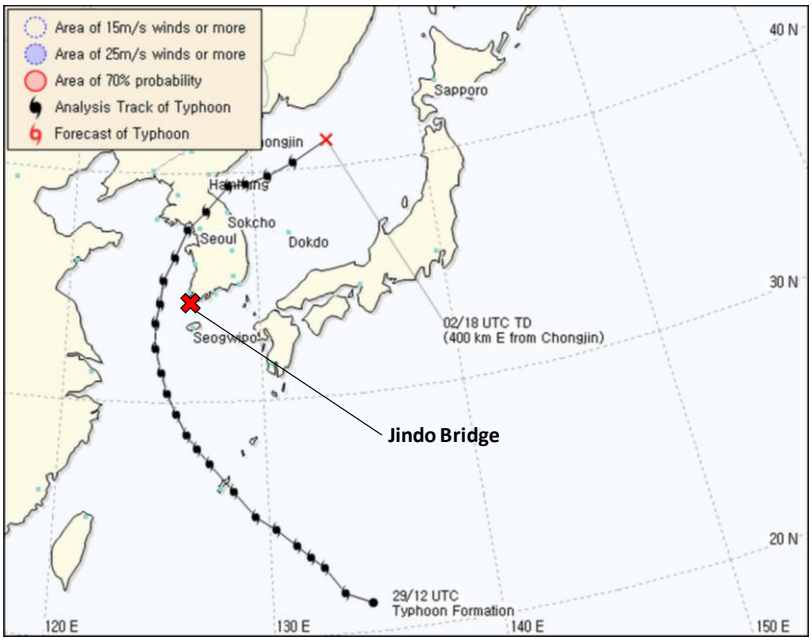


Figure 7.18 Typhoon Kompasu (9/2/2010 at the Jindo Bridge).

Decentralized modal analysis is conducted with the sensor topology shown in Figure 7.20. Each network consists of 4 subgroups with a cluster head and 7 or 9 leaf nodes, two of which are overlapping nodes. Note that N14 and N15 placed at the same location can provide phase information of two networks. The high sensitivity sensor boards (SHM-H) are assigned to all cluster heads. Jo *et al*. 2010 has indicated that given the limited number of high sensitivity sensors, using them as reference nodes (herein cluster heads serve as reference) in decentralized system identification is an efficient way to improve the identification results. This fact can be seen by comparing acceleration time history and power spectrum measured from SHM-A and SHM-H. When the acceleration level is high for both SHM-A and SHM-H to capture most vibration characteristics, the difference is not significant (see Figure 7.21). Note that measured accelerations for N26 and N54 in Figure 7.21 are similar to each other due to symmetry of the structure. On the contrary, the acceleration in Figure 7.22 is too low for SHM-A to accurately measure; thus the first several peaks between 0 and 1 Hz are under the noise floor.

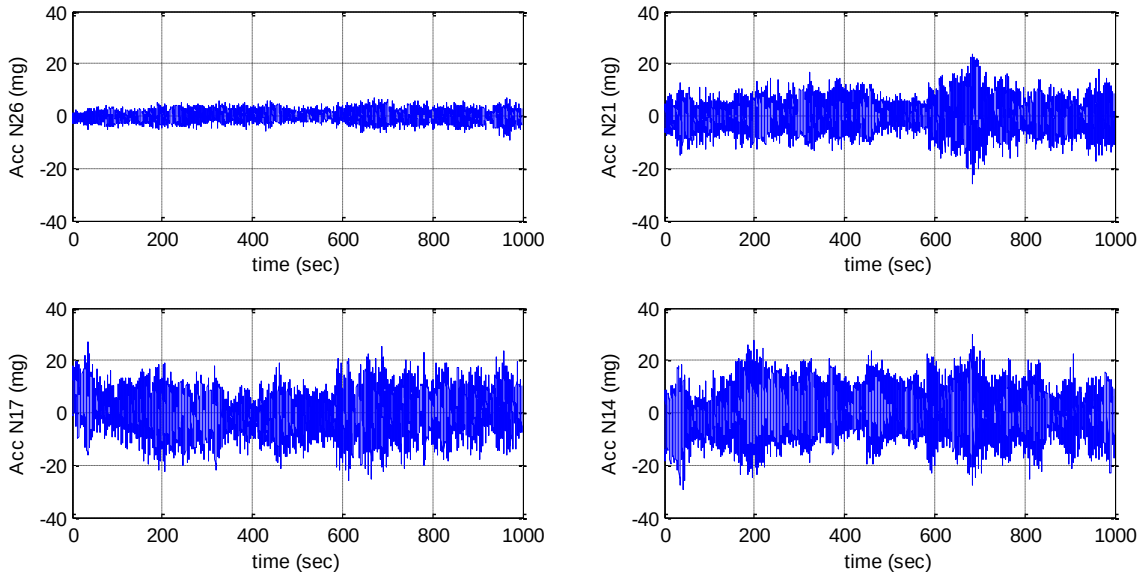


Figure 7.19 Time history data (vertical acceleration) from the Deck-Jindo Network.

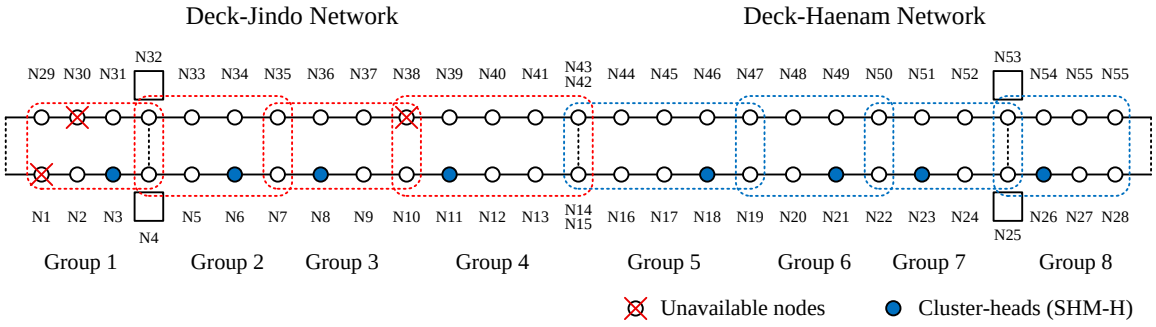


Figure 7.20 Sensor topology.

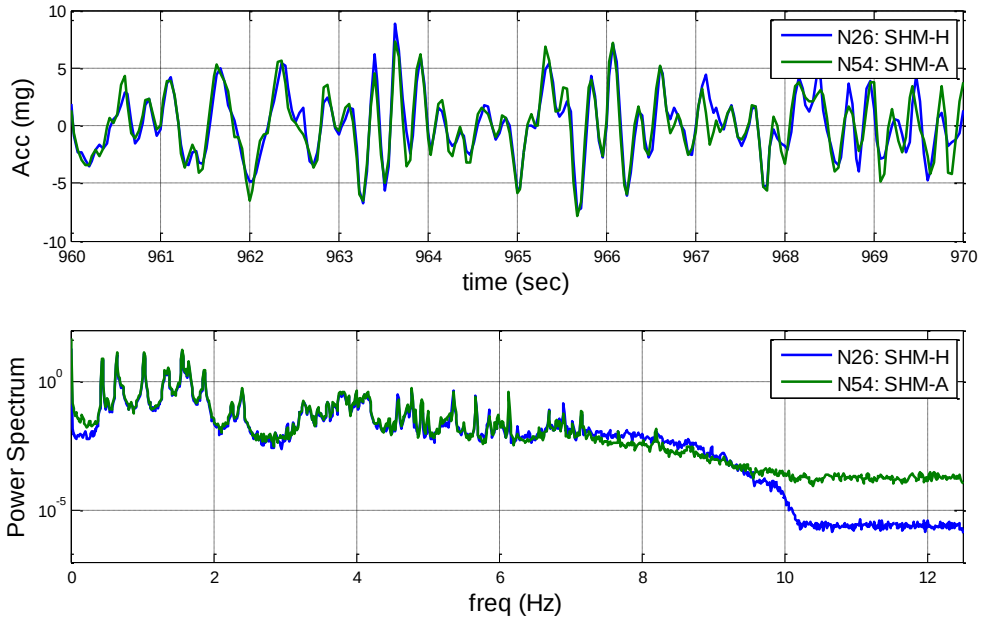


Figure 7.21 Time history data and power spectrum: high acceleration level.

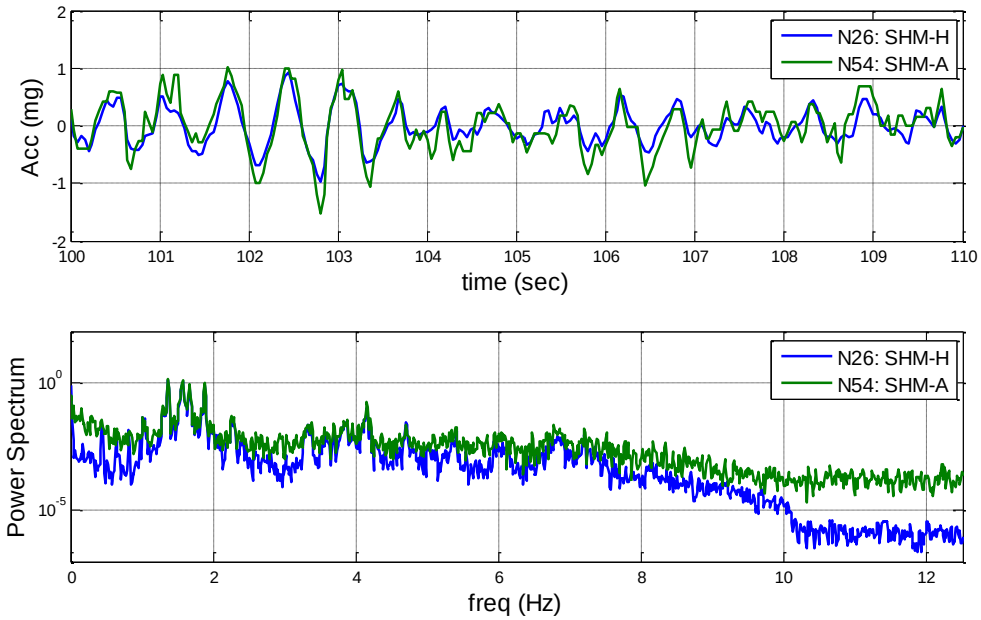
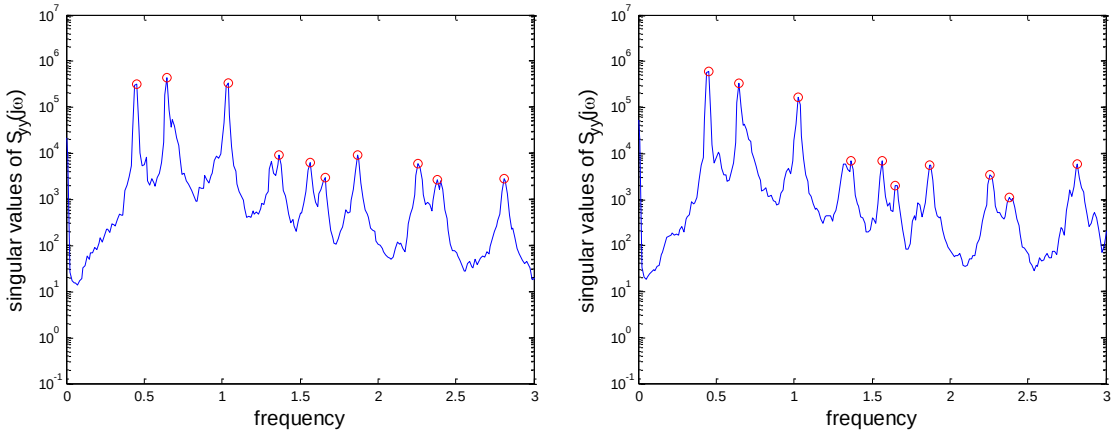


Figure 7.22 Time history data and power spectrum: low acceleration level.

Traditional modal analysis is also conducted using Frequency Domain Decomposition (FDD) (Brincker *et al*. 2001) to provide reference modal properties. Figure 7.23 shows 1st singular values in FDD, which can be considered as PSD (Brincker *et al*. 2001), in the frequency region of 0 – 3 Hz, where most fundamental natural modes of the Jindo Bridge are found (Cho *et al*. 2010). Because measured accelerations from the two networks are not time synchronized to each other, modal properties are calculated

separately in each network. The mode shape values corresponding to the pairs of overlapping nodes at the mid span (i.e., N14 and N15; N42 and N43) are then utilized to combine each network's mode shapes using Equation (4.7) as in the decentralized modal analysis.



(a) Jindo side (b) Haenam side

Figure 7.23 1st singular values in FDD (equivalent to PSD).

Combining the local mode shapes based on the overlapping nodes can result in unreasonable global mode shapes for a certain modes that have near-zero values at the mid span (see Figure 7.24). Because mode shapes values at the overlapping nodes are close to zero for the anti-symmetric modes, small errors introduced to these nodes can significantly distort the combined mode shape. As a result, the mode shapes for the Jindo and Haenam sides in Figure 7.24 are in phase with different magnitudes. To obtain appropriate mode shapes in such cases, the modes shapes are assumed to be anti-symmetric. Scale factors for the Jindo and Haenam sides to be out of phase with the same magnitudes of mode shapes are calculated in the least squares sense.

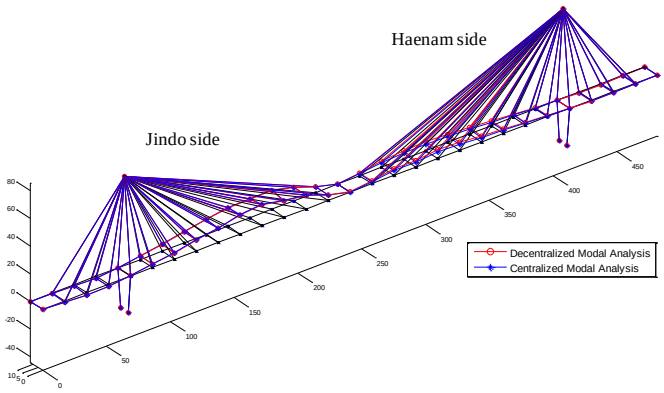


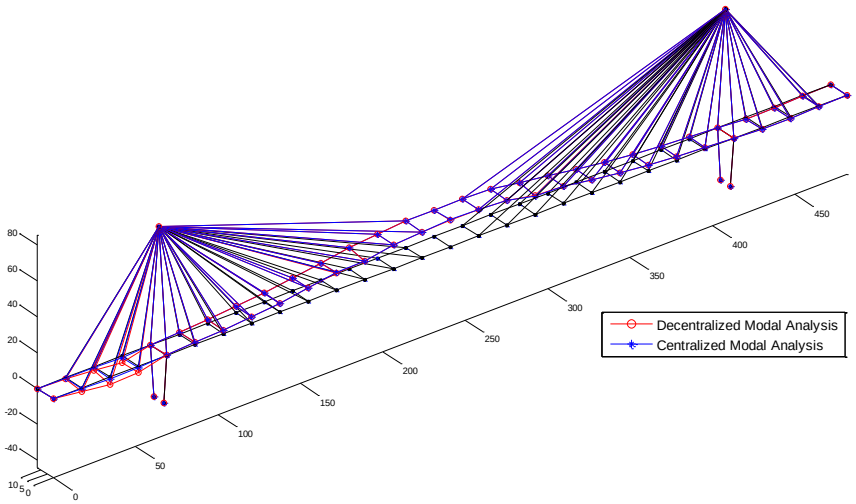
Figure 7.24 Combined global mode shapes with zero values at the mid span (0.6452 Hz).

The global modal properties estimated by centralized/decentralized modal analysis are compared in Table 7.3 and Figure 7.25. Note that anti-symmetric modes are reasonably estimated (see Figure 7.25b, d, and g). The natural frequencies for centralized

and decentralized approaches are close to each other; MAC as the accuracy measure for mode shapes indicates the global mode shapes are found in proximity. However in Figure 7.25, noticeable difference between two mode shapes is commonly seen in the side span at the Jindo side. Because the overlapping nodes N4 and N32 are located near the bridge pylon where the deck sits on the pier, N4 and N32 serve as nodal points of most modes. As discussed previously, overlapping nodes N4 and N32 with small mode shape values can introduce significant errors to combined mode shapes. In particular, the 1.6602 Hz mode with the largest error as shown in Table 7.3 has all overlapping nodes near the nodes of the mode shape (see Figure 7.25f). Excluding the side span at the Jindo side, MAC can be increased as shown in the last column of Table 7.3. From the natural frequencies and global mode shapes compared in Table 7.3 and Figure 7.25, decentralized modal analysis appears to estimate modal properties accurately.

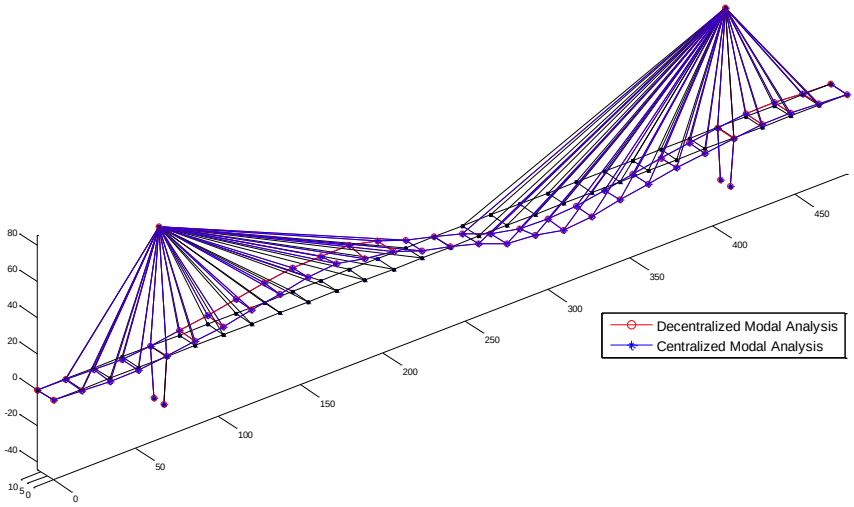
Table 7.3 Estimated natural frequencies and MAC between global mode shapes.

No.	Natural frequencies		MAC	
	Centralized	Decentralized	Complete mode shapes	w/o side span at Jindo network
1	0.4456	0.4462	0.9905	0.9999
2	0.6436	0.6452	1.0000	1.0000
3	1.0328	1.0316	0.9824	0.9954
4	1.3644	1.3660	0.9836	0.9923
5	1.5645	1.5617	0.9200	0.9552
6	1.6602	1.6581	0.4545	0.4545
7	1.8715	1.8724	0.9406	0.9959
8	2.2698	2.2620	0.9442	0.9808
9	2.3804	-	-	-
10	2.8133	2.8153	0.9240	0.9340

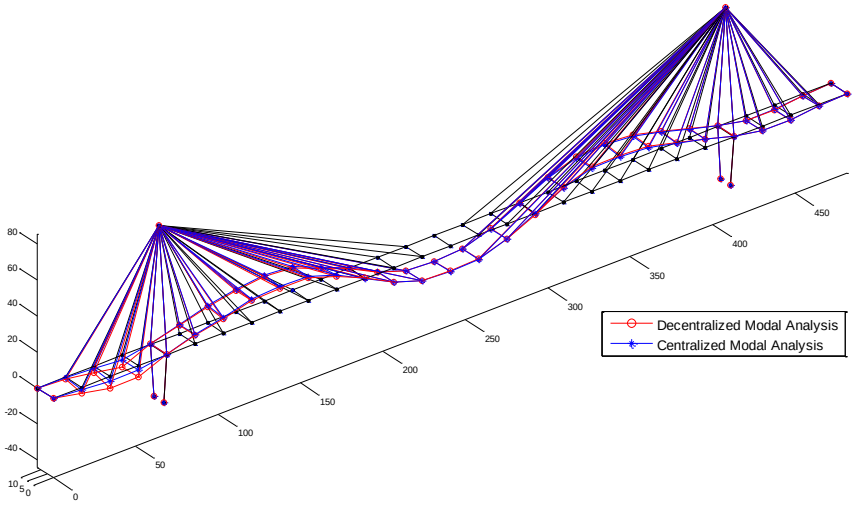


(a) 0.4456 Hz (MAC: 0.9905).

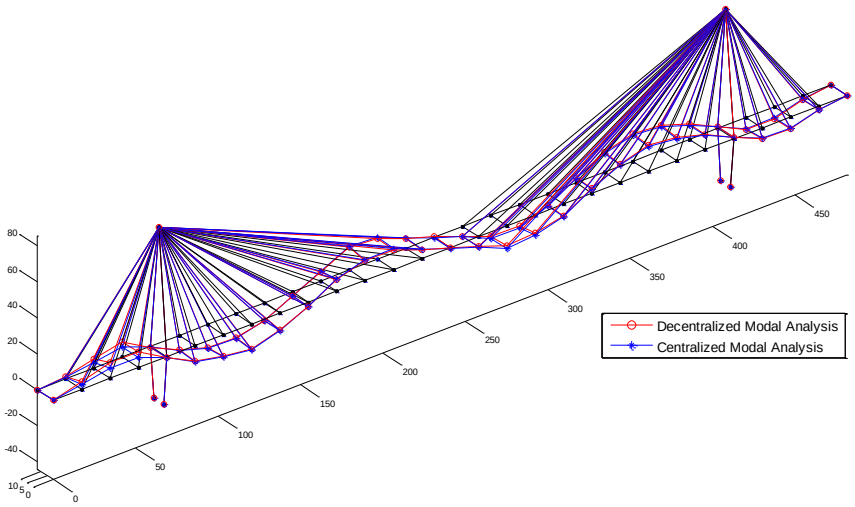
Figure 7.25 Global mode shapes and corresponding natural frequencies.



(b) 0.6452 Hz (MAC: 1.0000).

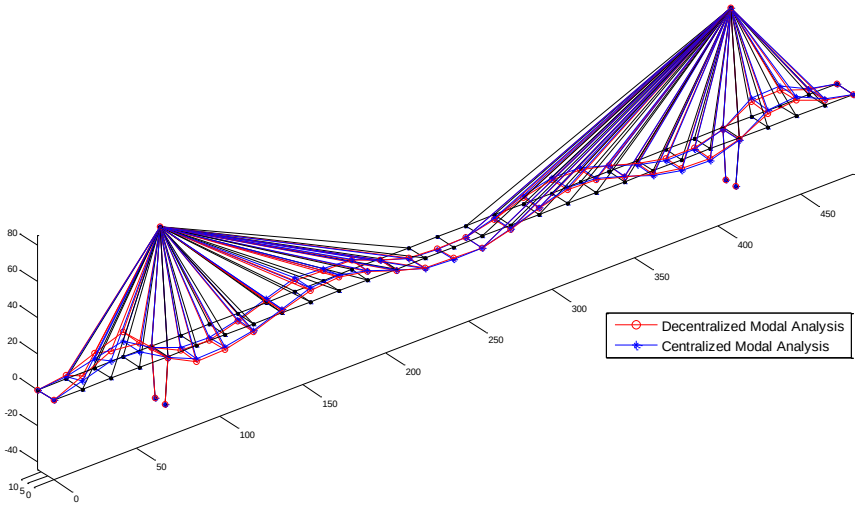


(c) 1.0328 Hz (MAC: 0.9824).

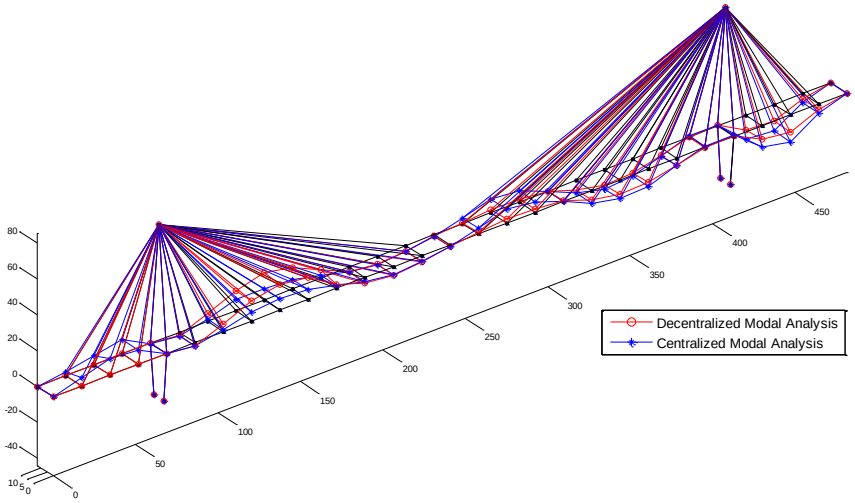


(d) 1.3644 Hz (MAC: 0.9836).

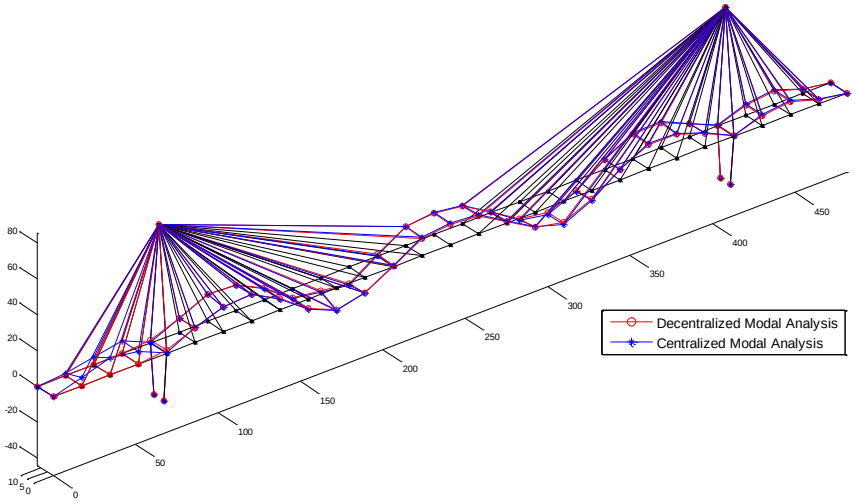
Figure 7.25 (cont.)



(e) 1.5645 Hz (MAC: 0.9200).

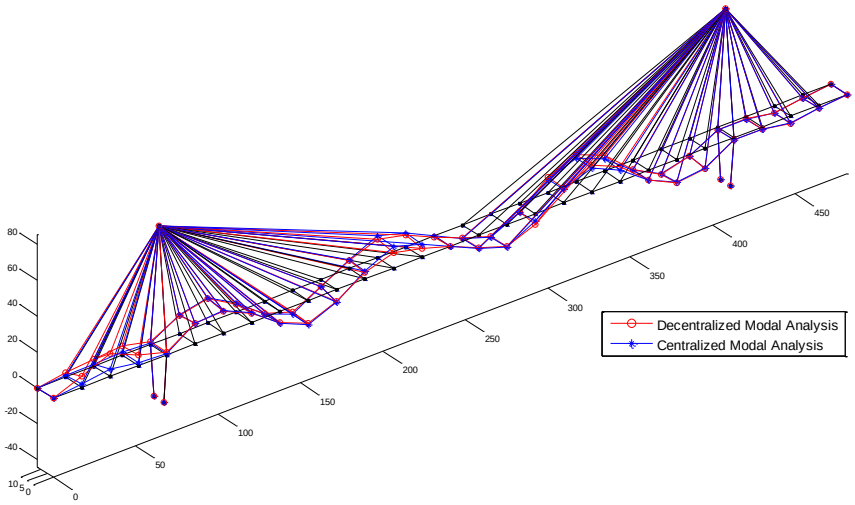


(f) 1.6602 Hz (MAC: 0.4545).

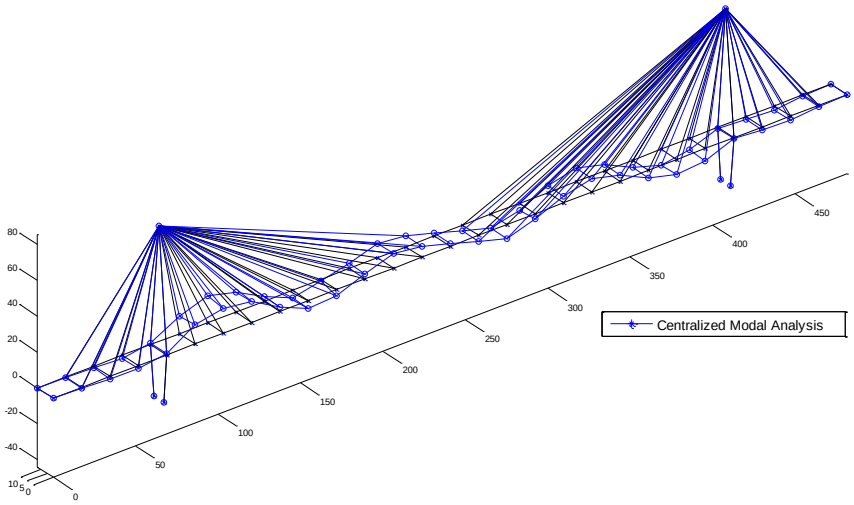


(g) 1.8715 Hz (MAC: 0.9406).

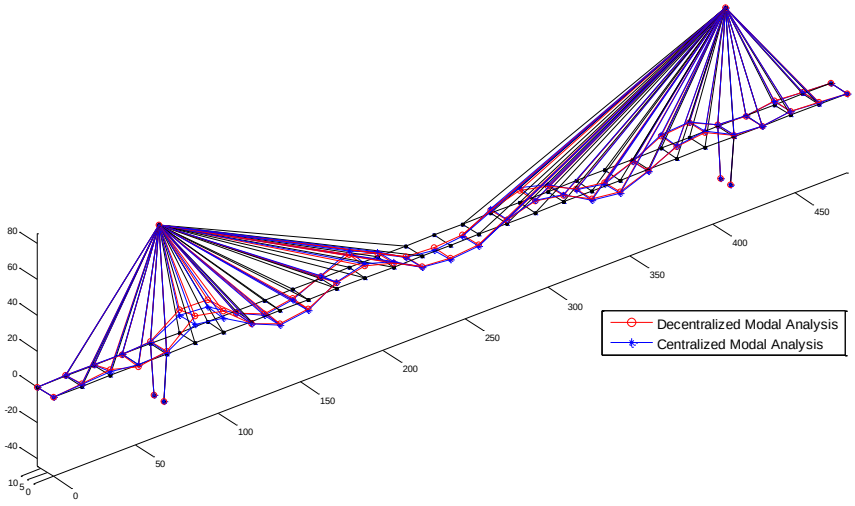
Figure 7.25 (cont.)



(h) 2.2698 Hz (MAC: 0.9442).



(i) 2.3804 Hz (MAC is not available).



(j) 2.8133 Hz (MAC: 0.9240).

Figure 7.25 (cont.)

Decentralized modal analysis with in-network data processing

Decentralized modal analysis is conducted using the Imote2's on-board computing capability for in-network data processing. *DecentralizedDataAggregation* measures and processes acceleration responses of the bridge, producing correlation functions in the base station at the Jindo Bridge. The correlation functions, outcome of the in-network data processing, are transferred to a local computer in the University of Illinois at Urbana-Champaign for estimation of global modal properties. The procedure of decentralized modal analysis for the Jindo Bridge deployment is schematically shown in Figure 7.26.

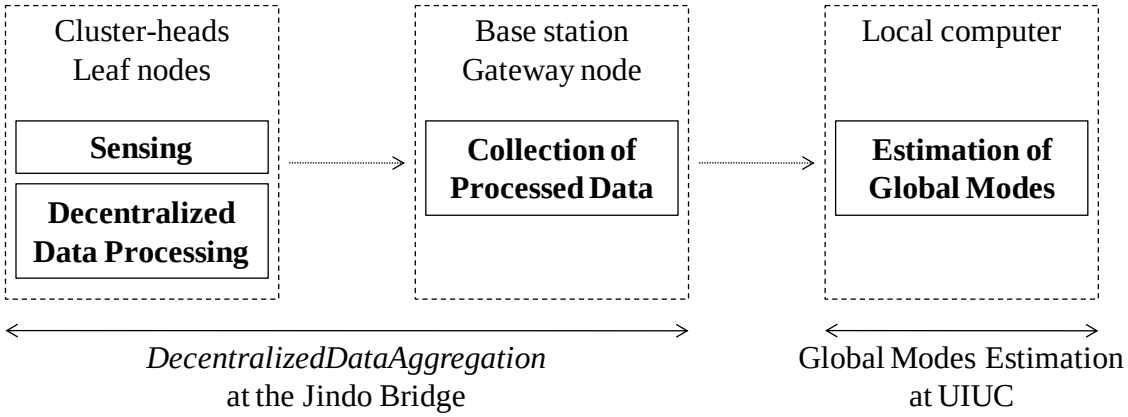


Figure 7.26 Flowchart of decentralized modal analysis for the Jindo Bridge deployment.

As the first step of the decentralized modal analysis, *DecentralizedDataAggregation* is used to obtain local information (i.e., correlation functions) with the same sensor topology shown in Figure 7.20. As the Jindo Bridge deployment is still in the debugging phase, only partial information is currently available; only the Deck-Haenam network is considered in the analysis. Cluster-heads in each group are sensor nodes with the SHM-H sensor boards. Having cluster-heads that provides high-precision acceleration, system identification can be significantly improved (Jo *et al.* 2010). The sensor nodes indicated as 'X' in Figure 7.20 are unavailable due to poor data communication or low battery. Ambient accelerations are measured under the normal condition, contrary to the previous case that the typhoon is the main excitation source. Parameters used in *DecentralizedDataAggregation* are summarized in Table 7.4.

DecentralizedDataAggregation has successfully conducted the whole procedure, consisting of sensing, in-network processing, and collection of the processed data in the large-scale network of sensors distributed over the broad area. Figure 7.28 shows sample correlation functions. The fault-tolerance has served as designed when some of the nodes experienced problems such as poor radio communication or low battery. The robustness of *DecentralizedDataAggregation* and its components (i.e., *SensingUnit, ReliableComm*, *RemoteCommand*) in the field testing is validated.

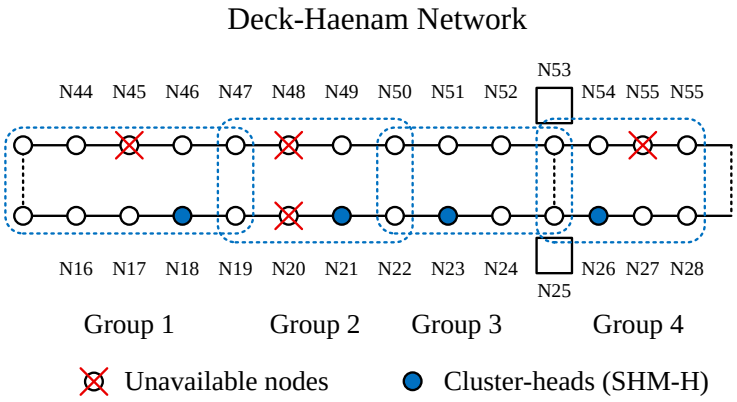


Figure 7.27 Sensor Topology.

Table 7.4 Data processing parameters for *DecentralizedDataAggregation*.

Sampling frequency	Cutoff frequency	Data channel	Number of FFT	Number of overlaps	Number of averages	Detrend
25 Hz	10 Hz	Vertical	1024	512	10	Yes

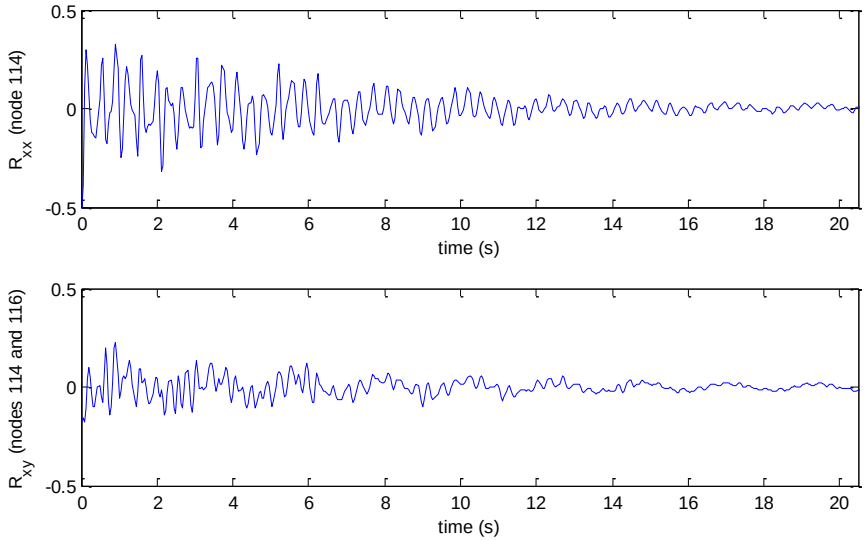
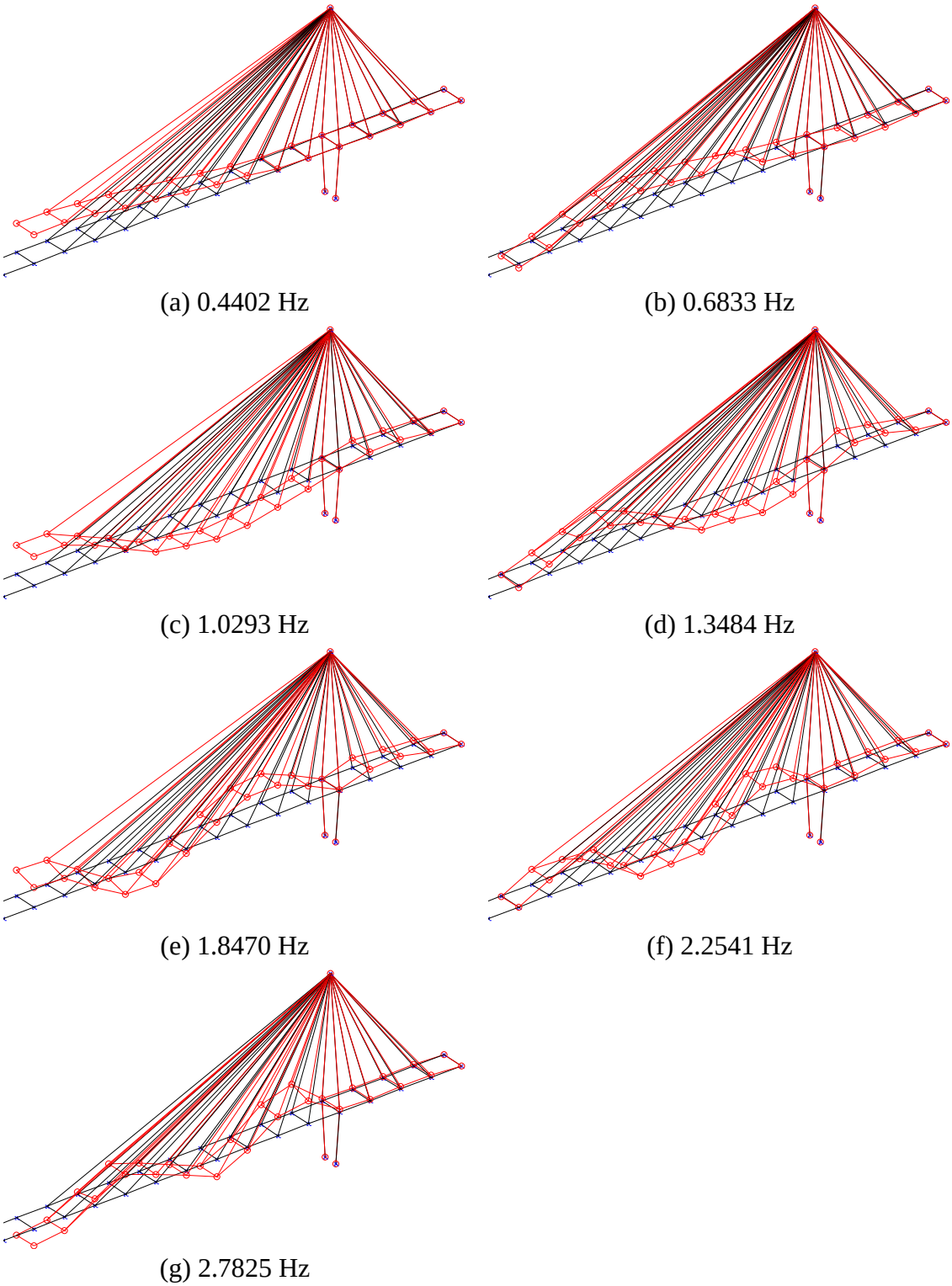


Figure 7.28 Auto- and cross correlation functions estimated in Group 1.

Global modal properties are estimated using the correlation functions obtained from *DecentralizedDataAggregation*. NExT/ERA is applied to each group to obtain local modal properties that are subsequently used to determine true modes and combine global mode shapes as found in Figure 7.29. In total, 7 natural modes are successfully estimated in the range of 0 – 3 Hz, and consistent with those obtained from the centralized approach previously described. Contrary to the previous case with typhoon Kompasu, the natural modes of 1.5645 Hz, 1.6602 Hz, and 2.3804 Hz are unidentified because these modes are not well excited as can be seen in the average absolute cross spectrum of each group

shown in Figure 7.30. As such, the performance of decentralized modal analysis is validated in the Jindo Bridge deployment.



(a) 0.4402 Hz

(b) 0.6833 Hz

(c) 1.0293 Hz

(d) 1.3484 Hz

(e) 1.8470 Hz

(f) 2.2541 Hz

(g) 2.7825 Hz

Figure 7.29 Global mode shapes from decentralized data aggregation.

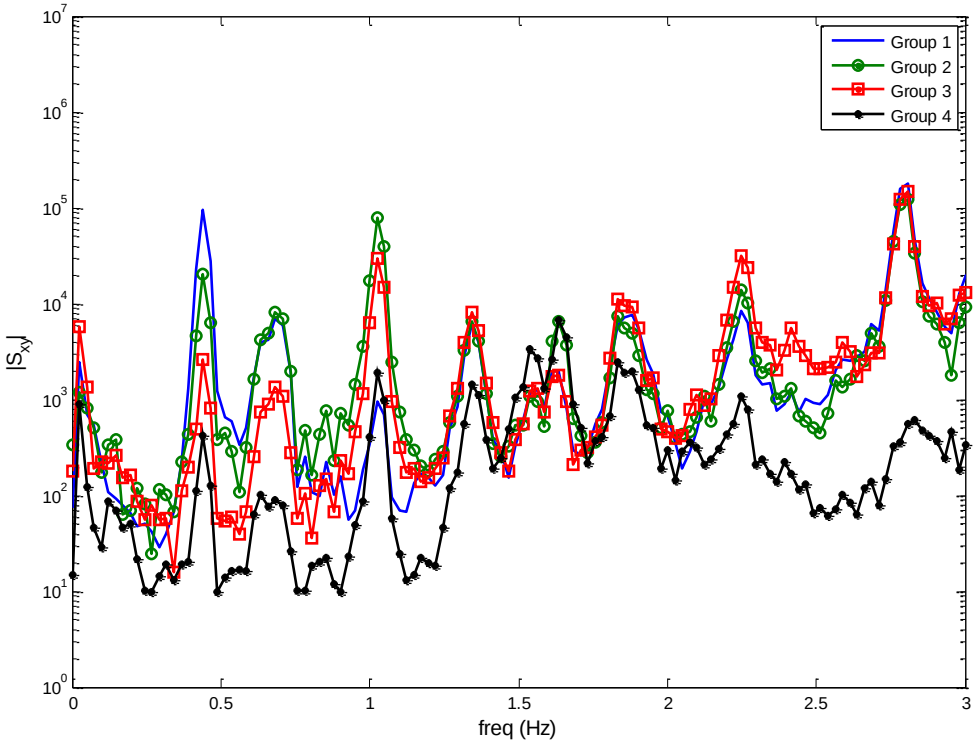


Figure 7.30 Average absolute cross spectrum.

7.2.5 Estimation of cable tension

The tension forces of the cables are estimated on the sensor nodes shown in Figure 7.31. Because the Jindo Bridge deployment is currently in the debugging phase as previously mentioned, only the sensor nodes located on the south-east side of the Cable-Haenam network are utilized in the cable tension monitoring (see Figure 7.31). The sensor node C10 is excluded from the network because C10 is unresponsive. *AutoMonitor* at the gateway node runs *CableTensionEstimation* on a regular basis with the predefined time interval (24 hours), saving the retrieved cable tension values in the base station.

Prior to running *CableTensionEstimation*, accelerations of the cables are collected using *RemoteSensing* to determine approximate natural frequencies that are required for the peak-picking method implemented in *CableTensionEstimation*. Local axis is defined as shown in Figure 7.32, and the corresponding power spectrum of cable accelerations are given for node C12 in Figure 7.33. Both *x* and *z* directional accelerations have the same, well separated peaks with an almost constant intervals, while the *y* directional acceleration along the cable has little dynamics of the cable. Because using both *x*- and *z*-axis is redundant for identification of natural frequencies, only *z*-axis is used for *CableTensionEstimation*. Natural frequencies for each cable node as well as properties of cables (see Table 7.5) are saved in the gateway node to use for estimation of each cable tension.

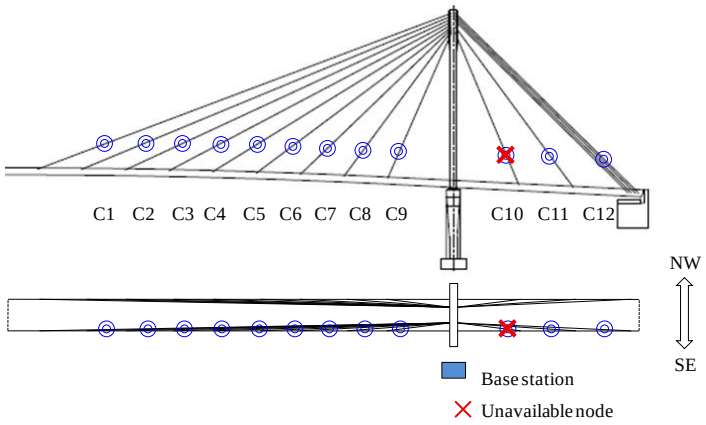


Figure 7.31 Cable nodes at the Cable-Haenam network.

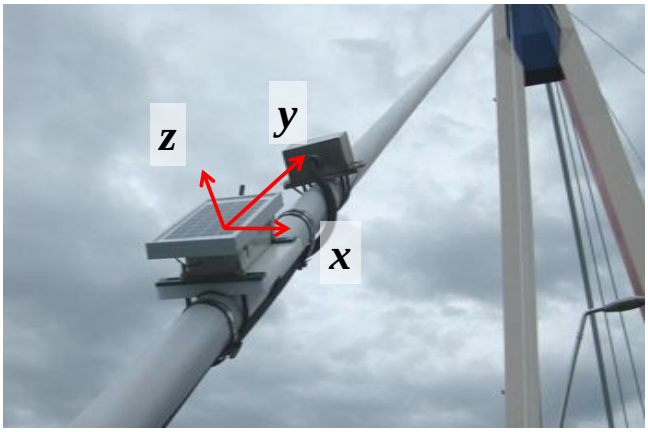


Figure 7.32 Local axis for a cable node (*x*: horizontal, *y*: along the cable, *z*: vertical to the cable and *x*-axis).

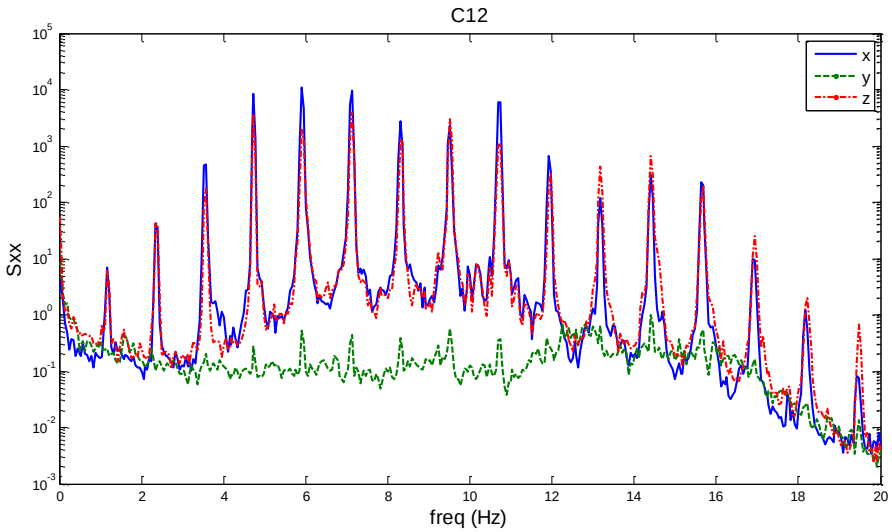


Figure 7.33 Power spectrum of cable vibration.

Table 7.5 Cable properties.

Cable		Total length (m)	Effective length (m) Park *et al*. (2008)	Unit mass (ton/m)
ID	Type			
C1	Φ7×139	174.150	169.686	0.0439
C2	Φ7×139	157.880	152.704	0.0439
C3	Φ7×109	141.755	136.869	0.0347
C4	Φ7×109	126.090	120.886	0.0347
C5	Φ7×109	110.940	106.201	0.0347
C6	Φ7×109	96.515	92.172	0.0347
C7	Φ7×73	83.165	79.011	0.0231
C8	Φ7×73	71.620	68.369	0.0231
C9	Φ7×73	62.625	58.170	0.0231
C10	Φ7×151	65.000	63.327	0.0476
C11	Φ7×151	78.600	76.761	0.0476
C12	Φ7×151	101.445	99.801	0.0476

Cable tensions in the debugging phase (9/8/10 and from 9/21/10 to 9/26/10) are tracked using *CableTensionEstimation* in conjunction with *AutoMonitor* as shown in Table 7.6. Note that some tension values have not been obtained due to poor communication. To check the validity of the estimation, cable tensions from *CableTensionEstimation* are compared to references that are the design tension and estimated tension from Park *et al.* (2008); mean tension values from *CableTensionEstimation* measured in Figure 7.34 are consistent with the references.

Table 7.6 Summary of cable tension forces.

ID	Design Tension	Park *et al*. (2008)	*CableTensionEstimation*					
			9/8/10 17:33	9/21/10 13:30	9/23/10 5:31	9/24/10 2:22	9/25/10 3:09	9/26/10 2:30
C1	202	223.3	222.5	-	222.6	220.4	222.6	219.3
C2	174	183.1	184.5	184.5	183.5	181.9	183.5	182.8
C3	160	164.3	165.7	168.2	165.9	165.2	165.9	165.6
C4	141	139.9	144.2	142.6	145.2	144.0	144.1	143.7
C5	122	132.8	132.9	134.2	133.8	133.2	131.7	133.0
C6	108	106.5	112.1	-	-	-	110.2	111.8
C7	90	86.2	89.2	-	86.8	-	86.8	88.0
C8	73	77.5	82.4	-	81.8	82.1	80.1	81.5
C9	70	59.0	60.8	59.62	58.8	58.8	58.8	59.1
C10	271	293.1	-	-	-	-	-	-
C11	271	304.8	300.1	301.0	295.4	304.5	295.4	282.1
C12	237	258.7	273.7	273.8	269.7	269.5	269.6	267.9

The estimated cable tensions shown in Figure 7.35 are within about 2% deviation from mean values in most cases. This variation is caused due to not only actual tension

changes but also errors introduced by the FFT-based data processing. As the simple peak-picking method has been used with power spectrum, natural frequencies are found only on the discrete spectral lines that are integer multiples of Δf (sampling rate divided by the number of FFT points). Thus, with small change in real natural frequency, the peak locations can be moved to the adjacent spectral lines. Fluctuation of the natural frequencies for node C11 shown in Figure 7.36 is considered due to the error from discretization as well as actual natural frequency changes. In particular, the discretization error in lower frequency modes has more impact on cable tension estimation as the ratios between Δf and natural frequencies are more significant in lower modes. The discretization error can be reduced by using a smaller sampling rate or a larger number of FFT points in power spectrum calculation.

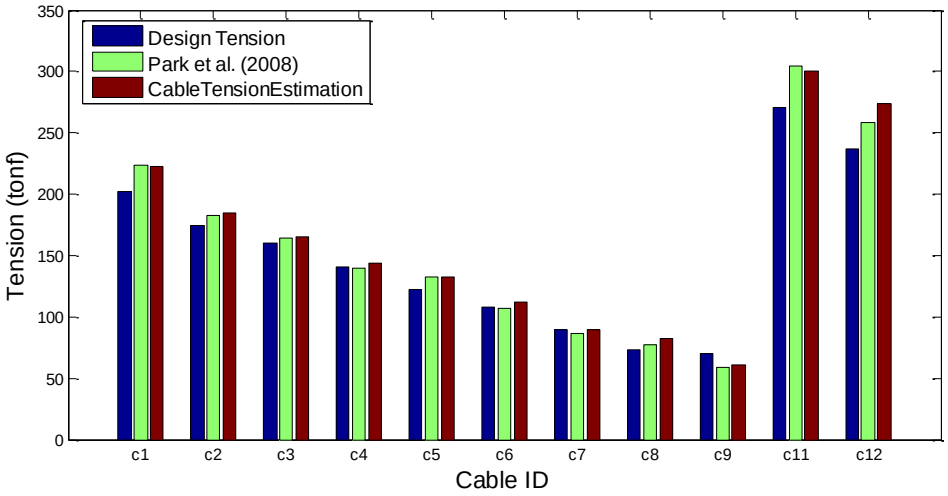


Figure 7.34 Comparison of the design tension, estimation by Park *et al*. (2008), and average tension of *CableTensionEstimation*

Table 7.7 Mean and maximum deviation of the estimated tension forces.

ID	Mean of cable tension (9/8/10 – 9/26/10)	Maximum deviation from the mean value (%)
C1	221.39	1.08
C2	2	0.78
C3	165.99	1.33
C4	143.82	0.93
C5	133.12	1.08
C6	111.10	0.91
C7	87.67	1.77
C8	81.55	1.73
C9	59.31	2.45
C10	-	-
C11	298.65	1.97
C12	270.55	1.31

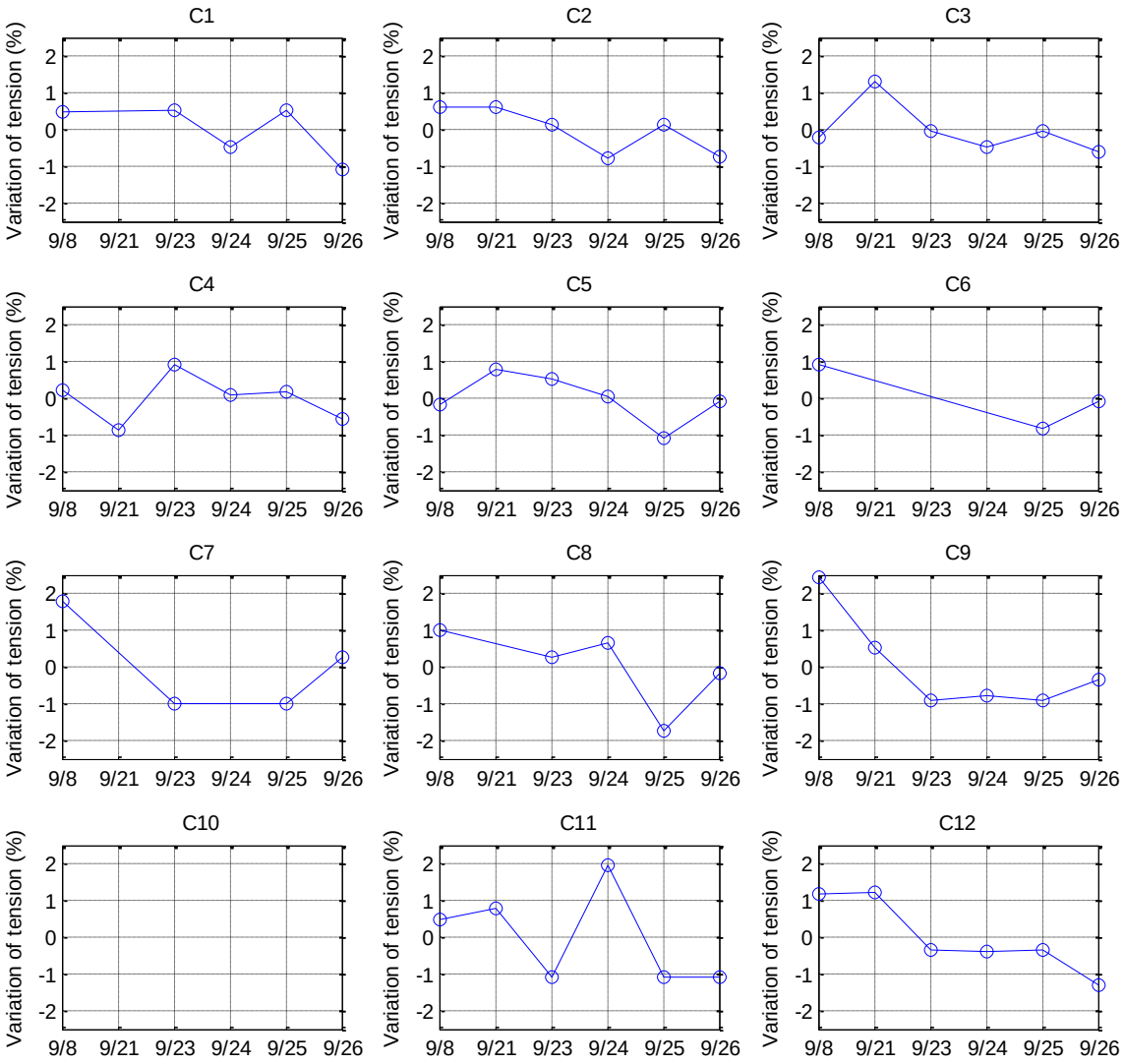


Figure 7.35 Variation of estimated cable tension with respect to mean values (Note that data from node C10 is not available).

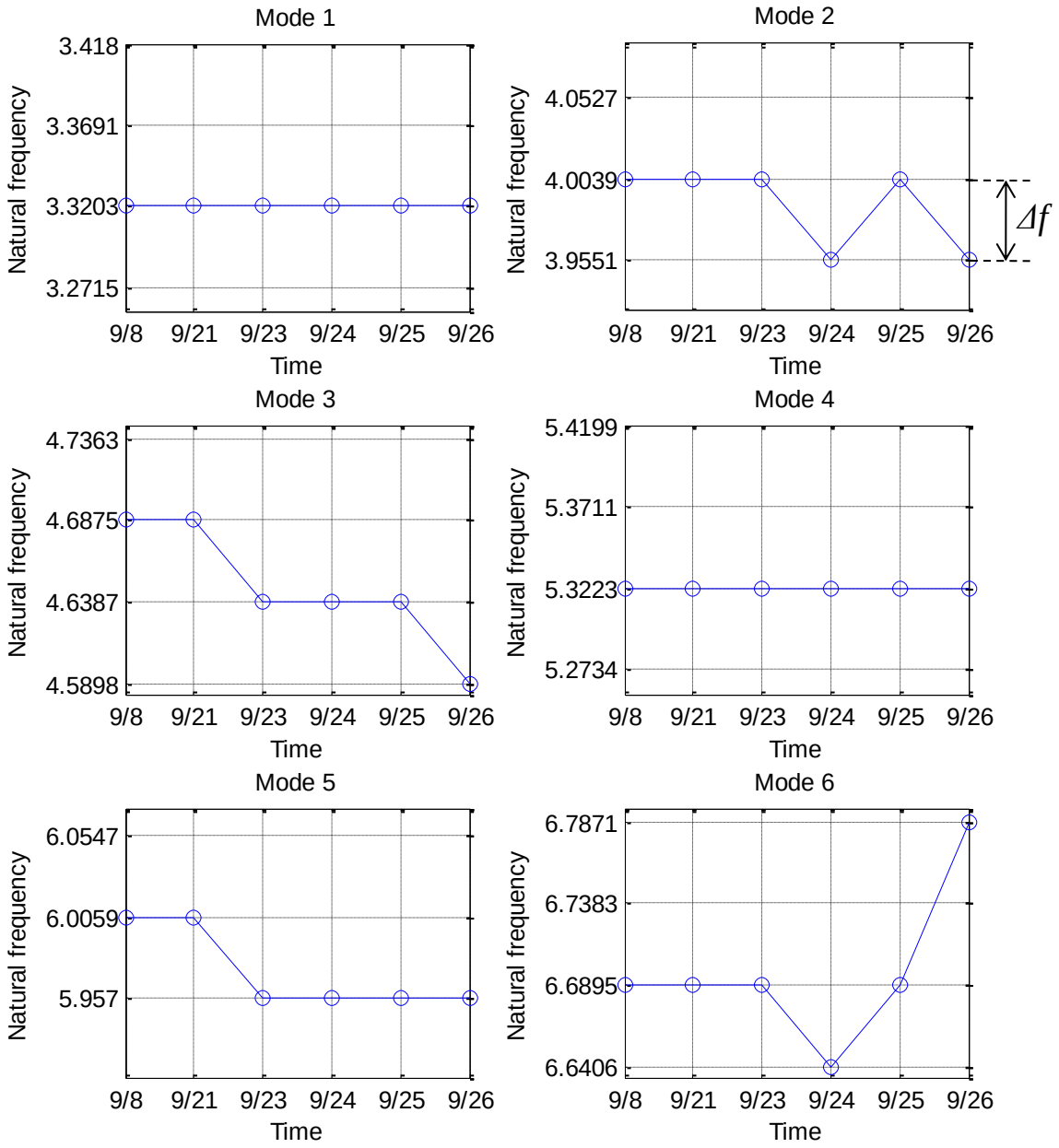


Figure 7.36 Change in lower six natural frequencies for node C11 (*Δf* is the frequency resolution, sampling rate divided by the number of FFT points).

Cable tensions could be successfully obtained using *CableTensionEstimation* autonomously ran by *AutoMonitor*. As the deployment on the cables is primarily focused on realization of the first autonomous WSSN for cable tension monitoring, the WSSN can be improved in both hardware and software aspects (e.g., different antenna for better communication, cable tension algorithm that considers the cable sag, and improved fault tolerance).

7.2.6 Summary

The WSSN on the Jindo Bridge was introduced for validation of the decentralized approaches. The hardware/software aspect of the deployment was described. For modal analysis, *DecentralizedDataAggregation* was able to successfully collect correlation functions from the large size of sensor network, proving the robustness and applicability to full-scale testing. The collected local information output the natural frequencies and global mode shapes that were shown to be accurate, compared to those from centralized data collection. In addition, the WSSN showed the potential of smart sensors for autonomous monitoring of cable tensions. The estimated cable tensions were shown to be consistent with the references as well as can be used as the valuable information for the maintenance purposes.

This chapter focused on providing experimental results to validate the efficacy and applicability of the WSSN to full-scale civil infrastructure. Based on the results shown in this chapter, conclusions and future research are provided in the next chapter.

CONCLUSIONS AND FUTURE STUDIES

8.1 Conclusions

This dissertation outlined the research on realization of the flexible, versatile wireless smart sensor network (WSSN) for monitoring and identification of civil infrastructure. To extend the current WSSN beyond emulating the traditional centralized approach typically used in the wired system, decentralized strategies have been adopted as an alternative to overcome the limitations and benefit the powerfulness of smart sensors. This dissertation consisted of introduction, background, theoretical development of the decentralized approaches, implementation on smart sensors, and experimental validation.

Extensive background on Structural Health Monitoring (SHM), modal analysis, and smart sensors has been provided. The smart sensor has been recognized as a promising alternative to overcome the intrinsic limitations that the traditional SHM systems have. For the smart sensor network to be implemented in the full-scale civil structures, scalability was shown to be essential, which can be achieved by the decentralized sensor network. While the decentralized approaches have been primarily developed for SHM, such algorithms from the modal analysis perspective were shown to be still lacking. In addition, introducing multimetric sensing was seen to have the potential to further enhance the performance of the WSSN.

To achieve the goal of this research, which is to develop a flexible, versatile WSSN based on the decentralized strategies, this research considered (1) decentralized modal analysis that enables the decentralized WSSN to be used for estimation of global modal properties, (2) efficient decentralized system identification in the WSSN, and (3) multimetric sensing to enrich essential information of the state of structures. Numerical simulation was conducted to verify the efficacy of the decentralized and multimetric sensing approaches.

Decentralized modal analysis has developed as an automated, decentralized strategy for modal analysis using smart sensors. The WSSN ultimately requires in-network processing utilizing smart sensor's onboard computing capability to efficiently use limited resources such as available bandwidth and battery. Thus, the processed data in the decentralized hierarchical network contains only local information of each local sensor community. Decentralized modal analysis provides a means to combine the local information to obtain the global picture of a structure. The numerical simulation using the plate and truss models shown in Figure 4.5 and Figure 4.13, respectively, has shown that the decentralized modal analysis can find the global modal properties accurately.

System identification methods were investigated for more efficient use of smart sensors in local data processing. Natural Excitation Technique (NExT) has been known as an efficient data processing method particularly for the decentralized WSSN, providing correlation functions that can be further used as the input to Eigensystem Realization Algorithm (ERA). While NExT has been used for decentralized in-network processing, study in Random Decrement Technique (RDT) revealed that RDT was more efficient in terms of wireless communication than NExT that was an important issue in

the WSSN. In NExT, cluster-heads in each local sensor community broadcasts a measured signal as a reference for correlation function estimation; cluster-heads in RDT sends the triggering information that is much smaller than the raw sensor data. As the sending reference data takes a significant part of radio communication in NExT, reducing the size of the reference data in RDT results in an important impact on the overall communication requirement. Numerical study with the 3D truss model shown in Figure 5.5 showed that if RDT was employed, data communication could be reduced to 22%, compared to the NExT case.

The third contribution of this study in the theoretical development is the use of multimetric data for structural damage detection. The fundamental idea behind this approach is that damage detection can be significantly enhanced by using a heterogeneous mix of measurands containing both local and global information. Here in this study, the Stochastic DLV method, an output-only, flexibility-based damage detection method, was extended to accommodate acceleration and strain in combination. The accuracy and reliability of the damage detection result in the presence of measurement noise could be improved due to the multimetric data in the numerical study.

The decentralized approaches developed for the WSSN were implemented on the Imote2 smart sensor platform based on the ISHMP Services Toolsuite. A wide variety of WSSN applications and services are available; thus the ISHMP Services Toolsuite reduces the efforts necessary for software development. To realize the decentralized approaches, several services and applications were developed using the ISHMP Services Toolsuite: *SensingUnit*, *IndependentProcessingPSD*, *CableTensionEstimation*, *DecentralizedDataAggregation*, and *GlobalModesEstimation*. *SensingUnit* is a basic service component for network-wide sensing, supporting two types of networks, centralized and decentralized networks. This service was used by the other three applications. *IndependentProcessingPSD*, an example implementation of independent processing, was designed to calculate the power spectrum of measured data. *CableTensionEstimation* was developed implementing the vibration-based cable tension estimation. It performs pick-picking on the power spectrum from *IndependentProcessingPSD* to obtain natural frequencies that are subsequently used in calculating cable tensions. *DecentralizedDataAggregation* is an application that performs decentralized in-network processing and outputs the processed data, correlation and RD functions. The processed data can be further used for local system identification or damage detection within the WSSN, or collected at the base station for other purposes such as global modal property estimation by the *GlobalModesEstimation*.

The decentralized approaches and software implementations were validated by a series of laboratory- and full-scale experiments: the 3D truss structure, the Irwin Indoor Practice Field, and the Jindo Bridge. As the first example, the 3D truss structure at the University of Illinois at Urbana-Champaign was utilized in the laboratory testing. A total of 14 Imote2 sensors with *DecentralizedDataAggregation* were installed on the truss, providing correlation and RD functions locally estimated in each subdivided sensor community. Global modal properties were successfully obtained both from correlation and RD functions, while RDT was seen to be more efficient in terms of data communication than NExT.

The second experiment was conducted in the Irwin Indoor Practice Facility at the University of Illinois at Urbana-Champaign. The steel arch box truss at the center of the

roof was selected as the test bed to conduct decentralized modal analysis. 14 Imote2 sensors with *DecentralizedDataAggregation* were deployed to measure ambient accelerations, process the sensor data in each subdivided local sensor community, and collect the processed data at the base station. The processed data only containing local information of each community was subsequently combined to produce the global modal properties of the center truss; several fundamental natural modes were successfully estimated. The experiment at the Practice Facility has showed the validity of decentralized modal analysis as well as the software reliability of *DecentralizedDataAggregation* and the basic services used in *DecentralizedDataAggregation* (i.e., *UnifiedSensing*, *SensingUnit*, *RemoteCommand*, and *ReliableComm*)

Ongoing efforts at the Jindo Bridge deployment, the world-largest smart sensor deployment for SHM to date, have demonstrated the performance and potential of the WSSN in the large-scale network of smart sensors. Both hardware and software were carefully prepared to conduct autonomous, long-term monitoring of the Jindo Bridge. The Jindo Bridge deployment features autonomous operation, power harvesting capability for all sensor nodes, multi-hop communication, decentralized data aggregation, cable tension monitoring, and high-precision acceleration measurement using the new SHM-H sensor board. This study mainly focused on investigating the efficacy of the decentralized approaches (i.e., decentralized modal analysis and cable tension monitoring) on the performance of the WSSN. As an example of decentralized coordinated processing, decentralized modal analysis is applied to estimate the global modal properties of the Jindo Bridge: *DecentralizedDataAggregation* collected local information using in-network processing within the network, and the global modal properties were combined at the local computer. Comparison between centralized and decentralized approaches showed a good agreement. Tension forces of the cables were monitored using *CableTensionEstimation* based on the decentralized independent processing. Monitoring cable tension showed the variation of the forces in terms of time and temperature. As such, the decentralized approaches and the software system based on the ISHMP Services Toolsuite were successfully validated at the Jindo Bridge deployment.

The decentralized strategies for the WSSN and multimetric sensing for damage detection have showed the significant impact on the capability of SHM for civil infrastructure. In particular, the decentralized approaches and their software implementations have intensively investigated from the theoretical development along with numerical simulation to a series of laboratory- and full-scale experiments. The efficacy of the decentralized approaches validated in this study is expected to be more distinct for a larger size and spatial distribution of sensor network that the centralized data collection is no longer applicable. In addition, multimetric data has been shown to enhance the accuracy and reliability of damage detection; if the WSSN is intended to monitor structural damage, multimetric data can be adopted for more reliable damage detection. Findings of this study will serve as a foundation that enables civil infrastructure monitoring using smart sensors to make a step forward in the field of SHM research.

8.2 Future studies

Advances in both hardware and software aspects of smart sensor technology have enabled the unprecedented large-scale WSSN at the Jindo Bridge. While the Jindo Bridge deployment has been an excellent test bed in which significant steps in the field of smart sensor research have been made, it also has revealed new issues that should be addressed for smart sensors to be more widely used for monitoring civil infrastructure. Herein, several issues for improvement of the WSSN are presented.

8.2.1 Multimetric sensing

As an example for the use of multimetric data, combination of acceleration and strain is considered to improve damage localization in this research. The Stochastic DLV method, which was extended to accommodate multimetric data in this study, was shown to be effective in the decentralized computing environment. Thus, the proposed damage detection approach for multimetric data is also well-suited to the large-scale WSSN. Laboratory experiments should be first conducted to verify the efficacy of the approach in a well-controlled environment. To implement on WSSNs, both hardware and software should be prepared. For the Imote2 sensor platform, acceleration sensor boards are already commercially available while appropriate strain sensor boards are not yet developed. To make use of acceleration and strain measurements in combination associated with smart sensors, strain sensor boards compatible with the current acceleration sensor boards should be developed. As the damage detection approach for multimetric data is based on the local system identification results, the *DecentralizedSysID* application that combines *DecentralizedDataAggregation* and the ERA service in the ISHMP Services Toolsuite can serve as a basis for the software development. Successful implementation of the decentralized damage detection strategy using multimetric data will significantly improve the performance of the WSSN.

In addition to the combined use of acceleration and strain for damage detection, GPS provides a unique opportunity for smart sensors to measure displacement. Despite usefulness of the displacement response, displacement has not been widely employed in SHM because traditional displacement sensors such as the Linear Variable Differential Transformer (LVDT) requires a reference point, which makes deployment particularly difficult for large-scale civil structures. However, recent advances in GPS technologies have allowed displacement measurement to be more viable in the SHM. Indeed, researchers have been using GPS in civil engineering applications (Leach and Hyzak 1992; Lovse *et al.* 1995; Brown *et al.* 1999; Roberts *et al.* 1999; Fujino *et al.* 2000a; Fujino *et al.* 2000b; Ogaja *et al.* 2003; Brownjohn *et al.* 2005). GPS chips in the market are relatively inexpensive and consume little power, so they can be operated readily with batteries. As such, a sensor board for the Mica Mote on which a GPS sensor is commercially available. However, the GPS-based displacement measurement is not sufficiently accurate for most SHM applications

One approach to improve the accuracy of GPS-based displacement measurements, this research proposes to address this problem to combine acceleration with GPS-based displacement measurements using a model-based Kalman estimator. Generally, displacement measurement of a physical system exhibits higher accuracy in the low frequency region while acceleration measurement is more reliable in the relatively high

frequency region. Knowing the different characteristics of these two measurements, combination of acceleration and displacement with a model-based Kalman estimator is considered to obtain accurate displacement estimate. For this purpose, development of a numerical model for the structure under consideration followed by model updating is necessary to appropriately design the model-based Kalman estimator to achieve improved displacement estimation.

8.2.2 Rare event monitoring

To date, most research for monitoring full-scale civil structures has been conducted based on ambient vibration that are relatively common and long duration. However, the smart sensors have potential to be used to monitor rare events such as earthquakes and heavy vehicle loadings. The critical issues that should be addressed to enable the rare event monitoring are related to power management. In the Jindo Bridge deployment, the sensor nodes are put in the deep sleep mode during the most of the operation time to save power. Thus, capturing the sudden events is not supported due to the time it takes to wake up nodes and prepare sensing. Monitoring such events with battery consumption low may not be possible with the current Imote2 sensor platform; realizing this feature may require both hardware and software supports.

One possible solution is an alarm system that powers on the mote for measurement if a certain triggering event occurs. A low-power sensor separately powered from the rest of the mote can be utilized to continuously measure vibration responses, and wake up the main microprocessor and sensor module if a certain condition is met (e.g., the maximum vibration level is greater than a predefined threshold). A disadvantage of this approach is difficulties in determining the condition. For example, setting the threshold value too low, normal events might be considered to be rare events, waking up the sensor node. On the contrary, too large threshold value may allow the main sensor module to measure the event after important large vibration passed. Thus, the triggering condition should be carefully selected. Although the rare event monitoring using smart sensors is a challenging task, it will greatly increase versatility of the WSSN for SHM.

8.2.3 Over-the-air programming

The WSSN for civil infrastructure monitoring will ultimately consist of hundreds or thousands of sensor nodes with huge spatial distribution over a structure. Reprogramming the sensor nodes, which can be required for maintenance such as software improvement and bug fixes, is a daunting task in the current Imote2 sensor platform because each node should be connected to a PC with a USB cable to reprogram. For large-scale civil structures, accessing each sensor node for reprogramming is not tractable.

This issue can be resolved by over-the-air programming that distributes new software images to each sensor node via wireless communication. In the Imote2 sensor platform, applications are stored in the designated address of the flash memory and loaded to RAM by *bootloader* on the startup. The application image can be wirelessly transmitted from the gateway node to the sensor nodes, and then saved in the flash memory. Updating the address where the new application located for *bootloader*, the new one is loaded to RAM. Implementation of the over-the-air programming for Imote2 will lower costs in maintenance of the deployed WSSN.

8.2.4 Fatigue prediction

Fatigue is an important issue in steel structures; the capability of monitoring fatigue accumulation can provide valuable information for timely maintenance practices. The fatigue accumulation can be predicted based on damage accumulation formulas using measured time history responses. In this approach, however the fatigue accumulation can be estimated only at the measurement points. To resolve this limitation, Papadimitriou *et al.* (2010) used the Kalman filter to estimate structural responses at unmeasured locations, which can lead to prediction of the fatigue accumulation. Numerical simulation using a simple spring-mass model and a truss model has showed the accuracy of the approach. However, several issues should be addressed for this approach to be incorporated with the WSSN. The prediction approach for the fatigue accumulation should be proven to work in the distributed computing environment intrinsically found in the WSSN. In addition, the prediction approach is based on the uniaxial stress that needs to be extended to the multiaxial stress to apply for general structures. Experimental verifications are also necessary for real-world applications. The estimation of fatigue accumulation can be incorporated with other damage detection approaches for more accurate characterization of civil structures.

REFERENCES

Allemang, R.J. and Brown, D.L. (1982) "A correlation coefficient for modal vector analysis," Proc. International Modal Analysis Conference, 110-116.

Asmussen J.C. (1997) "Modal analysis based on the Random Decrement Technique: Application to Civil Engineering structures," Ph.D. Dissertation, Aalborg University.

Begg, R.D., Mackenzie, A. C., Dodds, C. J. and Loland, O. (1976) "Structural integrity monitoring using digital processing of vibration signals," Proc. 8th Annual Offshore Technology Conference, Houston, TX, 305-311.

Bendat, J.S. and Piersol, A.G., 1993, "Engineering applications of correlation and spectral analysis," Second Edition, Wiley, New York.

Bernal, D. (2002) "Load vectors for damage localization," J. Engineering Mechanics, ASCE, 128(1), 7-14.

Bernal, D. and Gunes, B. (2004) "Flexibility-based approach for damage characterization: Benchmark application," J. Engineering Mechanics, ASCE, 130(1), 61-70.

Bernal, D. (2006) "Flexibility-based damage localization from stochastic realization results," J. Engineering Mechanics, ASCE, 132(6), 651-658.

Bernal, D. (2007a) "Damage localization from the null space of changes in the transfer matrix," AIAA Journal, 45(2), 374-381.

Bernal, D. (2007b) "Damage localization in systems with unknown inputs," Key Engineering Materials, 347, 107-112.

Brincker, R., Krenk, S., Kirkegaard, P.H. and Rytter, A. (1992) "Identification of the dynamical properties from correlation function estimates," Bygningsstatikse Meddelelser, Danish Society for Structural Science and Engineering, 63(1), 1-38.

Brincker, R. (1995) "Note about the Random Decrement Technique," Aalborg University.

Brincker, R., Zhang, L. and Andersen, P. (2001) "Modal identification of output-only systems using frequency domain decomposition," Smart Materials and Structures, 10, 441-445.

Brown, C.J., Karuna, R., Ashkenazi, V., Roberts, G.W. and Evans, R.A. (1999) "Monitoring of structures using GPS," Proc. the Institution of Civil Engineers Structures and Buildings, 1, 97–105.

Brownjohn, J.M.W., Pan, T.C. and Cheong, H.K. (1998) "Dynamic response of Republic Plaza, Singapore," The Structural Engineer, 76(11), 221-226.

Brownjohn, J.M.W. and Pan, T.C. (2001) "Response of tall buildings to weak long distance earthquakes," Earthquake Engineering and Structural Dynamics, 30, 709-729.

Brownjohn, J.M.W. (2005) "Lateral loading and response for a tall building in the nonseismic doldrums," Engineering Structures, 27, 1801–1812.

Brownjohn, J.M.W. (2007) "Structural health monitoring of civil infrastructure," Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences, 365(1851), The Royal Society, 589–622.

Caffrey, J., Govindan, R., Johnson, E., Krishnamachari, B., Masri, S., Sukhatme, G., Chintalapudi, K., Dantu, K., Rangwala, S., Sridharan, A., Xu, N. and Zuniga, M. (2004) "Networked sensing for structural health monitoring," Proc. 4th International Workshop on Structural Control, New York, NY, June 10–11, 57–66.

Caicedo, J.M., Clayton, E., Dyke, S.J., Abe, M. and Tokyo, J. (2002) "Structural health monitoring for large structures using ambient vibrations," Proc. of the ICANCEER Conference, Hong Kong, August, 15-20.

Celebi, M., Purvis, R., Hartnagel, B., Gupta, S., Clogston, P., Yen, P., O'Connor, J. and Franke, M. (2004) "Seismic instrumentation of the Bill Emerson Memorial Mississippi River Bridge at Cape Girardeau (MO): A cooperative effort," 4th International Seismic Highway Conference, Memphis, TN.

Chance, J., Tomlinson, G.R. and Worden, K. (1994) "A simplified approach to the numerical and experimental modeling of the dynamics of a cracked beam," Proc. 12th International Modal Analysis Conference, 778-785.

Chang, P.C. and Liu, S.C. (2002) "Recent research in nondestructive evaluation of civil infrastructures," J. Materials in Civil Engineering, 15, 298, (doi: 10.1061/(ASCE)0899-1561(2003)15:3(298)).

Chang, P.C., Flatau, A. and Liu, S.C. (2003) "Review paper: health monitoring of civil infrastructure," Structural Health Monitoring, 2(3), 257-267.

Cho, S., Lynch, J.P., Lee, J.-J. and Yun, C.-B. (2010a) "Development of an automated wireless tension force estimation system for cable-stayed bridges," J. Intelligent Material Systems and Structures, 21(3), 361-376, (doi: 10.1177/1045389X09350719).

Cho, S., Jo., H., Jang, S., Park, J., Jung, H.J., Yun, C.B., Spencer, Jr., B.F. and Seo, J.-W. (2010b) "Structural health monitoring of a cable-stayed bridge using smart sensor technology: Data analyses," Smart Structures and Systems, 6(5-6), 461-480.

Cole, H.A. (1968) "On-the-analysis of random vibrations," Paper No. 68-288. American Institute of Aeronautics and Astronautics.

Dallard, P., Fitzpatrick, A.J., Flint, A., Le Bourva, S., Low, A., Ridsdill Smith, R.M. and Willford, M. (2001) "The London Millennium Footbridge," The Structural Engineer, 79(22): 17-33.

Doebling, S.W., Farrar, C.R., Prime, M.B. and Shevitz, D.W. (1996) "Damage identification and health monitoring of structural and mechanical systems from changes in their vibration characteristics: a literature review," Los Alamos National Laboratory Report, LA-13070-MS.

Ewins, D.J. (1986) "Modal testing: Theory and practice," Research Studies Press, Ltd., Staunton, Somerset, U.K.

Fang, I.-K., Chen, C.-R and Chang, I.-S. (2004) "Field static load test on Kao-Ping-Hsi cable-stayed bridge," J. Bridge Engineering, 9, 531.

Felber, A.J., 1993, "Development of a hybrid bridge evaluation system," Ph.D. Dissertation, University of British Columbia, Vancouver, Canada.

Fox, C. H.J. (1992) "The location of defects in structures: A comparison of the use of natural frequency and mode shape data," Proc. 10th International Modal Analysis Conference, 522-528.

Fujino, Y., Abe, M., Shibuya, H., Yanagihara, M., Sato, M., Nakamura, S. and Sakamoto, Y. (2000a) "Forced and ambient vibration tests and vibration monitoring of Hakucho Suspension Bridge," Transportation Research Record: J of the Transportation Research Board, 1696, 57-63.

Fujino, Y., Murata, M., Okano, S., Takeguchi, M. (2000b) "Monitoring system of the Akashi Kaikyo Bridge and displacement measurement using GPS," Proc. of SPIE, non-destructive evaluation of highways, utilities, and pipelines IV, 229–236.

Gao, Y., Spencer Jr., B.F. and Bernal, D. (2004) "Experimental verification of the damage locating vector method," Proc. 1st Int. Workshop on Advanced Smart Materials and Smart Structures Technology, Honolulu, Hawaii, January 12-14.

Gao, Y., Spencer Jr., B.F. and Ruiz-Sandoval, M. (2005) "Distributed computing strategy for structural health monitoring," Structural Control and Health Monitoring, 43(1), 488-507.

Glisic, B., Inaudi, D., Hoong, K.C. and Lau, J.M. (2003) "Monitoring of building columns during construction," Proc. 5th Asia Pacific Structural Engineering and Construction Conference, Johor Bahru, Malaysia, 593-606.

Gul, M. and Catbas, N. (2008) "Ambient vibration data analysis for structural identification and global condition assessment," J. Engineering Mechanics, ASCE, 134(8), 650–662.

Hermans, L. and Van Der Auweraer, H. (1999) "Modal testing and analysis of structures under operational conditions: Industrial applications," Mechanical Systems and Signal Processing, 13(2), 193-216.

Ibrahim, S.R. and Mikulcik, E.C. (1973) "A time domain modal vibration test technique," Shock and vibration test technique; 43(4), 21-37.

Ibrahim, S.R. and Pappa, R.S. (1982) "Large modal survey testing using the Ibrahim time domain identification technique," J. Spacecraft and Rockets, 19(5), 459-465.

James, G.H., Carne, T.G. and Lauffer, J.P. (1993) "The natural excitation technique (NExT) for modal parameter extraction from operating wind turbines," Sandia Report, SAND92-1666, Sandia National Laboratories, Albuquerque, NM.

Jang, S., Jo, H., Cho, S., Mechitov, K.A., Rice, J.A., Sim, S.-H., Jung, H.J., Yun, C.B., Spencer, Jr., B.F. and Agha, G. (2010a) "Structural health monitoring of a cable-stayed bridge using smart sensor technology: Deployment and evaluation," Smart Structures and Systems, 6(5-6), 439-459.

Jang, S., Sim, S.-H., Jo, H. and Spencer, Jr., B.F. (2010b) "Decentralized bridge health monitoring using wireless smart sensors," Proc. SPIE, Vol.7647, 76473I.

Jo, H., Rice, J.A., Spencer, Jr., B.F. and Nagayama, T. (2010) "Development of high-sensitivity accelerometer board for structural health monitoring," Proc. SPIE, 7647, 764706.

Juang, J.N. and Pappa, R.S. (1985) "An eigensystem realization algorithm for modal parameter identification and model reduction," J. Guidance Control and Dynamics, 8, 620-627.

Kijewski-Correa, T., Su, S., Abittan, E. and Antsaklis, P.J. (2006) "On the use of heterogeneous, wireless sensor networks for damage assessment in bridges under unknown excitations," Proc. 4th World Conference on Structural Control and Monitoring.

Kim, B.H. and Park, T. (2007) "Estimation of cable tension force using the frequency-based system identification method," J. Sound and Vibration, 304, 660-676.

Kim, B.H., Park, T., Shin, H. and Yoon, T.Y. (2007) ‘‘A comparative study of the tension estimation methods for cable supported bridges,’’ Int. J. Steel Structures, 7(1), 77-84.

Ko, J.M. and Ni, Y.Q. (2005) "Technology developments in structural health monitoring of large-scale bridges," Engineering Structures, 27, 1715–1725.

Law, S.S., Li, X.Y., Zhu, X.Q. and Chan, S.L. (2005) "Structural damage detection from wavelet packet sensitivity," Engineering Structures, 27(9), 1339-1348.

Leach, M. and Hyzak, M. (1992) "Results from a bridge motion monitoring experiment," Proc. 6th International Geodetic Symposium on Satellite Positioning, The Ohio State University, 801–810.

Lee, J.J., Lee, J.W., Yi, J.H., Yun, C.B. and Jung, H.Y. (2005) "Neural networks-based damage detection for bridges considering errors in baseline finite element models" J. Sound and Vibration, 280(3-5), 555-578.

Linderman, L.E., Rice, J.A., Barot, S., Spencer, Jr., B.F., Bernhard, J.T. (2010) "Characterization of wireless smart sensor performance," Newmark Structural Engineering Laboratory (NSEL) Report Series, No. 21, University of Illinois at Urbana-Champaign, Urbana, Illinois (http://hdl.handle.net/2142/15101).

Lovse, J.W., Teskey, W.F., Lachapelle, G. and Cannon, M.E. (1995) "Dynamic deformation monitoring of tall structure using GPS technology," J. Survey Engineering, 121(1), 35–40.

Lynch, J.P., Sundararajan, A., Law, K.H., Kiremidjian, A.S., Carryer, E., Sohn, H. and Farrar, C. (2003) "Field validation of a wireless structural monitoring system on the Alamosa Canyon Bridge," SPIE 10th Annual International Symposium on Smart Structures and Materials, San Diego, CA, USA, March 2-6.

Lynch, J.P., Sundararajan, A., Law, K.H., Kiremidjian, A.S. and Carryer, E. (2004a) "Embedding damage detection algorithms in a wireless sensing unit for operational power efficiency," Smart Materials and Structures, 13(4), 800–810.

Lynch, J.P., Parra-Montesinos, G., Canbolat, B.A. and Hou, T-C. (2004b) "Real-time damage prognosis of high-performance fiber reinforced cementitious composite

structures," Proc. Advances in Structural Engineering and Mechanics (ASEM'04), Seoul, Korea, September 2–4.

Lynch, J.P. and Loh, K. (2006) "A summary review of wireless sensors and sensor networks for structural health monitoring," Shock and Vibration Digest, 38(2), 91–128.

Maeck, J., Peeters, B. and De Roeck, G. (2001) "Damage identification on the Z24 Bridge using vibration monitoring analysis," Smart Material and Structure, 10, 512–517.

Maia, N.M.M. and Silva, J.M.M. (2001) "Modal analysis identification techniques," Philosophical Transactions: Mathematical, Physical and Engineering Sciences, 359(1778), Experimental Modal Analysis, 29-40.

Mehrabi, A.B. and Tabatabai, H. (1998) "Unified finite difference formulation for free vibration of cables," J. Structural Engineering, ASCE, 124 (11), 1313–1322.

Miller, T.I., Spencer, Jr., B.F.; Li, J., Jo, H. (2010) "Solar energy harvesting and software enhancements for autonomous wireless smart sensor networks," Newmark Structural Engineering Laboratory (NSEL) Report Series, No. 22, University of Illinois at Urbana-Champaign, Urbana, Illinois (http://hdl.handle.net/2142/16300).

Montgomery, D.C. and Runger, G.C. (1994) "Applied statistics and probability for engineers," John Wiley & Sons, Inc.

Nagayama, T., Sim, S.-H., Miyamori, Y. and Spencer, B.F., Jr. (2007) "Issues in SHM employing smart sensors," Smart Structures and Systems, 3(3), 299-320.

Nagayama, T. and Spencer Jr., B.F. (2007) "Structural health monitoring using smart sensors," Newmark Structural Engineering Laboratory (NSEL) Report Series, No. 1, University of Illinois at Urbana-Champaign, Urbana, Illinois (http://hdl.handle.net/2142/3521).

Nagayama, T., Moinzadeh, P., Mechitov, K., Ushita, M., Makihata, N., Ieiri, M., Agha, G., Spencer, Jr., B.F., Fujino, Y. and Seo, J.-W. (2010) "Reliable multi-hop communication for structural health monitoring," Smart Structures and Systems, 6(5-6), 481-504.

Nataraja, R. (1983) "Structural integrity monitoring in real seas," Proc. 15th Annual Offshore Tech. Conference, 221-228.

Ni, Y.Q., Ko, J.M. and Zheng, G. (2002) "Dynamic analysis of large-diameter sagged cables taking into account Flexural Rigidity," J. Sound and Vibration, 257(2), 301-319, (doi:10.1006/jsvi.2002.5060).

Nitta, Y., Nagayama, T., Spencer Jr., B.F. and Nishitani, A. (2005) "Rapid damage assessment for the structures utilizing smart sensor MICA2 MOTE," Proc. of 5th International Workshop on Structural Health Monitoring, Stanford, CA., 283-290.

Ogaja, C., Wang, J. and Rizos, C. (2003) "Detection of wind-induced response by wavelet transformed GPS solutions," J. Survey Engineering, 129(3), 99-104.

Pakzad, S.N., Fenves, G.L., Kim, S. and Culler, D.E. (2008) "Design and implementation of scalable wireless sensor network for structural monitoring," J. Infrastructure Systems, 14(1), 89-101.

Pandey, A.K., Biswas, M. and Samman, M.M. (1991) "Damage detection from changes in curvature mode shapes," J. Sound and Vibration, 145(2), 321-332.

Pandey, A.K. and Biswas, M. (1994) "Damage detection in structures using changes in flexibility," J. Sound and Vibration, 169(1), 3-17.

Papadimitriou, C., Fritzen, C.-P., Kraemer, P. and Ntotsios, E. (2010) "Fatigue prediction in entire body of metallic structures from a limited number of vibration sensors using Kalman filtering," J. Structural Control and Health Monitoring, (doi: 10.1002/stc.395).

Park, Y.S., Choi, S.M., Yang, W.Y., Hong, H.J. and Kim, W.H. (2008) "A study on tension for cables of a cable-stayed bridge damper is attached," J. Korean Society of Steel Construction, 20(5), 609-616.

Park, J.H. (2009) "Development of autonomous smart sensor nodes for hybrid structural health monitoring of large structures," Ph.D. Dissertation, Pukyung National University, Korea.

Peeters, B. and De Roeck, G. (2001) "Stochsatic system identification for operational modal analysis: A review," J. Dynamic Systems, Measurements, and Control; 132, 659-667.

Penrose, R. (1955) "A generalized inverse for matrices," Proc. the Cambridge Philosophical Society, 51, 406-413.

Rahimi, M., Hardik, S., Sukhatme, G.S., Heideman, J. and Deborah, E. (2003) "Studying the feasibility of energy harvesting in a mobile sensor networks," Proc. IEEE International Conference on Robotics and Automation. Taipai. Taiwan; 1, 19-24.

Rice, J.A. and Spencer, Jr., B.F. (2009) "Flexible smart sensor framework for autonomous full-scale structural health monitoring," Newmark Structural Engineering Laboratory (NSEL) Report Series, No. 18, University of Illinois at Urbana-Champaign, Urbana, Illinois (http://hdl.handle.net/2142/13635).

Rice, J.A., Mechitov, K., Sim, S.-H., Nagayama, T., Jang, S.A., Kim, R., Spencer, Jr., B.F., Agha, G. and Fujino, Y. (2010) "Flexible smart sensor framework for autonomous structural health monitoring," Smart Structures and Systems, 6(5-6), 423-438.

Roberts, G.W., Dodson, A.H., Ashkenazi, V. (1999) "Twist and deflection: Monitoring motion of Humber Bridge," GPS World, 10, 24–34.

Rodrigues, J., Brincker, R. and Andersen, P. (2004) "Improvement of frequency domain output-only modal identification from the application of the Random Decrement Technique," Proc. 23rd Int. Modal Analysis Conference. Deaborn. USA.

Shih, C.Y., Tsuei, Y.G., Allemang, R.J., Brown, D.L. (1988) "Complex mode indicator function and its application to spatial domain parameter estimation," Mechanical Systems and Signal Processing, 2(4), 367-77.

Shimada, T. (1994) "Estimating method of cable tension from natural frequency of high mode," Proc. JSCE, 501/1-29, 163-171.

Sim, S.-H. and Spencer, Jr., B.F. (2009) "Decentralized strategies for monitoring structures using wireless smart sensor networks," Newmark Structural Engineering Laboratory (NSEL) Report Series, No. 19, University of Illinois at Urbana-Champaign, Urbana, Illinois (http://hdl.handle.net/2142/14280).

Sim, S.-H., Spencer, Jr., B.F., Zhang, M. and Xie, H. (2010a) "Automated decentralized modal analysis using smart sensors," J. Structural Control and Health Monitoring, (doi: 10.1002/stc.348).

Sim, S.-H., Carbonell-Marquez, J.F., Spencer, Jr., B.F. and Jo, H. (2010b) "Decentralized Random Decrement Technique for efficient data aggregation and system identification in wireless smart sensor networks," Probabilistic Engineering Mechanics, 26, 81-91.

Sim, S.H., Spencer, Jr., B.F., and Nagayama, T. (2011) "Multimetric sensing for structural damage detection," J. Engineering Mechanics, ASCE, 137(1), pp. 22-30, doi:10.1061/(ASCE)EM.1943-7889.0000199.

Siringoringo D.M. and Fujino, Y. (2006) "Observed dynamic performance of the Yokohama-Bay Bridge from system identification using seismic records," J. Structural Control and Health Monitoring, 13(1), 226-244.

Small, E.P., Philbin, T., Fraher, M. and Romack, G.P. (1999) "The current status of bridge management system implementation in the United States," 8th International Bridge Management Conference, Denver, Colorado.

Sohn, H. and Farrar, C.R. (2001) "Damage diagnosis using time series analysis of vibration signals," Smart Materials and Structures, 10, 446-451.

Sohn, H. and Law, K.H. (2001) "Damage diagnosis using experimental Ritz vectors," J. Engineering Mechanics, ASCE, 127(11), 1184-1192.

Sohn, H., Farrar, C.R., Hemez, F.M., Shunk, D.D., Stinemates, D.W. and Nadler B.R. (2003) "A review of structural health monitoring literature: 1996-2001," Los Alamos National Laboratory Report, LA-13976-MS.

Spencer Jr., B.F., Ruiz-Sandoval, M. and Kurata, N. (2004) "Smart sensing technology: Opportunities and challenges," J. Structural Control and Health Monitoring.

Srinivasan, M.G. and Kot, C.A. (1992) "Effects of damage on the modal parameters of a cylindrical shell," Proc. 10th International Modal Analysis Conference, 529-535.

Straser, E.G. and Kiremidjian, A.S. (1998) "A modular, wireless damage monitoring system for structures," Technical Report 128, John A. Blume Earthquake Engineering Center, Stanford University, Stanford, CA.

Studer, M. and Peters, K. (2004) "Multi-scale sensing for damage identification," Smart Materials and Structures, 13, 283-294.

Tanner, N.A., Wait, J.R., Farrar, C.R. and Sohn, H. (2003) "Structural health monitoring using modular wireless sensors," J. Intelligent Material Systems and Structures, 14(1), 43–56.

Van Overschee, P. and De Moor, B. (1993) "Subspace algorithms for the stochastic identification problem," Automatica, 29, 649–660.

Vandiver, J.K., Dunwoody, A.B., Campbell, R.B. and Cook, M.F. (1982) "A mathematical basis for the random decrement vibration signature analysis technique," J. Mechanical Design, 104, 307–13.

Wang, M.L., Wang, G. and Zhao, Y. (2005) "Application of EM stress sensors in large steel cables," Sensing Issues in Civil Structural Health Monitoring, Chapter III, 145-154, (doi: 10.1007/1-4020-3661-2_15).

Wong K.Y. (2004) "Instrumentation and health monitoring of cable-supported bridges," J. Structural Control and Health Monitoring, 11(2), 91–124.

Yi, J.H. and Yun, C.B. (2004) "Comparative study on modal identification methods using output-only information," Structural Engineering and Mechanics, 17(3-4), 445-466.

Zienkiewicz, O.C. and Cheung, Y.K. (1964) "The finite element method for analysis of elastic isotropic and orthotropic slabs," Proc. Institution of Civil Engineers, 28, 471-488.

Zimmerman, A.T., Shiraishi, M., Swartz, R.A. and Lynch, J.P. (2008) "Automated modal parameter estimation by parallel processing within wireless monitoring systems," J. Infrastructure Systems, 14(1), 102-113.

Zui, H., Shinke, T. and Namita, Y.H. (1996) “Practical formulas for estimation of cable tension by vibration method,” J. Structural Engineering, ASCE, 122(6), 651–656.

COMMAND LINE USAGE OF APPLICATIONS

The Imote2 applications developed in this study are part of the open source ISHMP Services Toolsuite that can be downloaded from the ISHMP website (http://shm.cs.uiuc.edu). To take full advantage of the applications implementing decentralized approaches, installation of TinyOS 1.x and ISHMP Services Toolsuite 2.2.0 or higher is required. Following documents, found at the ISHMP website, are useful to install them as well as to get familiar with programming on TinyOS and ISHMP Services Toolsuite.

- Getting Started for Advanced Users and Developers guide (http://shm.cs.uiuc.edu)
- Imote2 for Structural Health Monitoring: User's Guide (http://shm.cs.uiuc.edu)
- TinyOS tutorial (http://www.tinyos.net/tinyos-1.x/doc/tutorial/) (up to Lesson 4)

Once TinyOS and ISHMP Services Toolsuite are installed on a PC, the Imote2 applications are needed to compiled and installed on each mote. Each application has own commands (called BluSH command) that should be properly used to have the application run. The BluSH commands are typically used to set up Imote2 node IDs and necessary parameters (e.g., sampling rate for any sensing application), and start an application. This appendix illustrates how to run two main applications used in this study (i.e., *CableTensionEstimation* and *DecentralizedDataAggregation*).

A.1 CableTensionEstimation

The procedure to run *CableTensionEstimation* consists of two steps as in all other applications: (1) step 1 for compiling and installation the application and (2) step 2 for running *CableTensionEstimation*.

Step 1: Preparation

The first step is to compile and install the application. Open a cygwin window and change the directory to where *CableTensionEstimation* is located as:

```
cd $SHMROOT/tools/CableTensionEstimation
```

Use a USB-MiniB cable to connect an Imote2 to the computer. Run the following command to compile and install:

```
make imote2 usbinstall
```

This command will compile *CableTensionEstimation*, and install the binary file on the connected Imote2. Next, run the following command simply to install the binary file of *CableTensionEstimation* on each subsequent Imote2:

```
make imote2 usbreinstall
```

Once installation is complete, run the following command in the cygwin window:

```
autocomm –d COMy
```

COMy represents the highest port listed in the Windows Device Manager. By pressing the <Enter> key a couple of times, a BluSH prompt should appear. If not, check the port number used for COMy and the connection between the interface board and the computer. Open another cygwin window, and type:

```
autocomm –n –o out.txt COMx
```

COMy is the second highest port listed in the Windows Device Manager. This command will save the PSD to the output file out.txt. The '–n' option suppresses printing outputs on the screen.

Step 2: Running CableTensionEstimation

Three BluSH commands should be used to specify the sensor nodes and the parameters for the PSD calculation, and to start the operation. At the first cygwin window, input:

```
SetCTENodes <nodeId> [nodeId] [nodeId]
```

To specify parameters necessary for tension estimation, input:

```
SetCTEParameters channelMask fs nfft navg overlap window normalize savepsd ppmethod estimation_method
```
(Note: this command is all one line.)

where

- channelMask specifies sensing channels: 1 for channel 1, 12 for channels 1 and 2, and 123 for channels 1, 2, and 3
- fs is the sampling frequency in Hz
- nfft is the number of FFT points (i.e., N_{FFT})
- navg is the number of averaging windows
- overlap is the number of overlapping points between consecutive spectral windows
- window specifies the spectral window: 1 for Welch window, 2 for Hamming window, and 3 for Hanning window
- normalize decides if the mean values of the measured data are removed: 1 to remove, and 0 not to remove
- savepsd decides if the estimated power spectrum is collected: 1 to save, and 0 not to save
- ppmethod specifies the peak-picking method: 1 for a method that is designed for cable vibrations without cable-structure interaction, and 2 for a method that requires knowledge of approximate peak locations
- estimation_method specifies the estimation method for cable tensions: 1 for Zui's formula, and 2 for the method utilizing the cable dynamics as well as least squares that used in this study

Then, cable properties need to be provided using the following command:

```
SetCTECableParameters NodeID E area moment length wc angle sag designT Cvalue
```
(Note: this command is all one line.)

where

- NodeID: ID for this node
- E: elastic modulus

- `area`: sectional area
- `moment`: moment of inertia
- `length`: effective length of cable
- `wc`: weight per length
- `angle`: angle of cable
- `sag`: span to sag ratio
- `design`: design tension force
- `Cvalue`: C value defined in Zui's formula

If the least squares-based method is used, only sectional area and weight design are required; use any small number (e.g., 0) for the rest of them. If `ppmethod` of 2 (the method that requires approximate peak locations), the following command should be used:

```
SetCTERefFreq NodeID width mode# freq mode# freq …
```

where

- width defines the ratio of search width to the sampling frequency. The highest point in this region is determined as the peak.
- mode#: mode number for the following natural frequency
- freq: natural frequency

To start *CableTensionEstimation*, type:

```
StartCTE
```

An example of running *CableTensionEstimation* is presented in the screenshots (see Figure A.1 for the gateway node, and Figure A.2 for the leaf nodes).

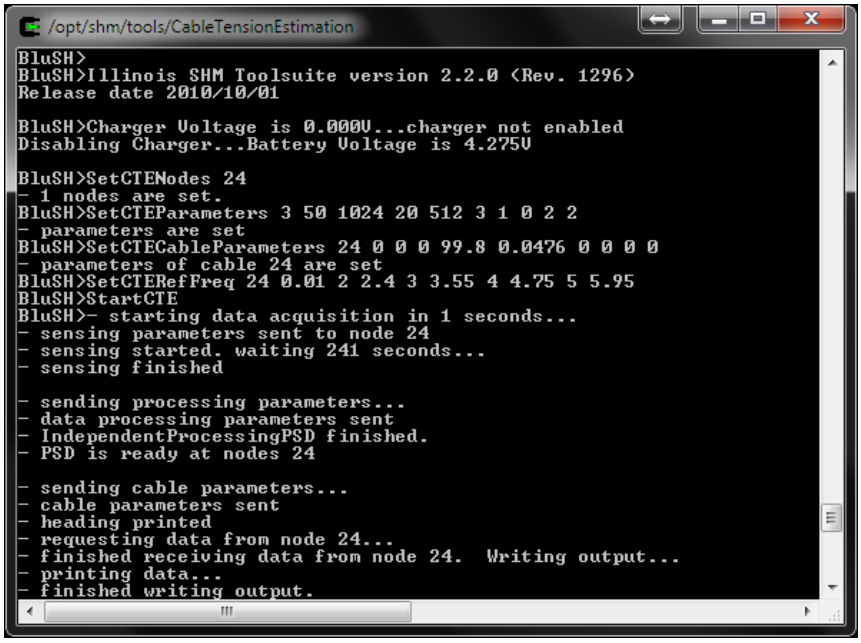


Figure A.1 Output messages from the gateway node in *CableTensionEstimation.*

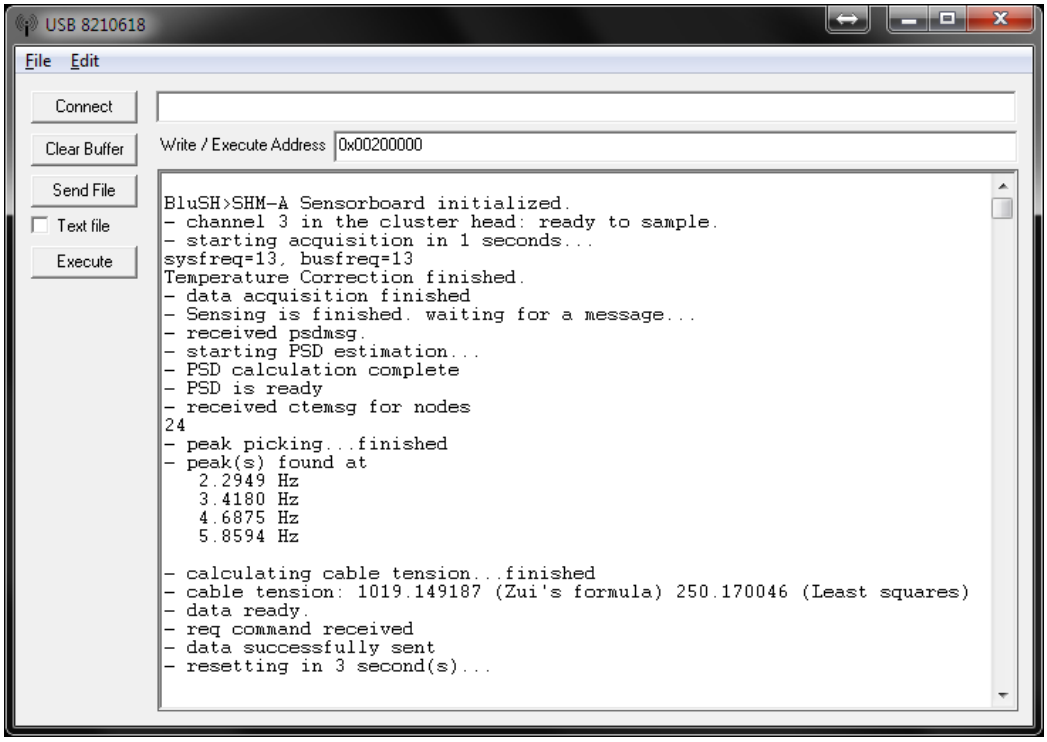


Figure A.2 Output messages from the leaf node in *CableTensionEstimation*.

A.2 DecentralizedDataAggregation

The following steps are required to run *DecentralizedDataAggregation*, as was the case for *CableTensionEstimation*.

Step 1: Preparation

The first step is to compile and install the application. Open a cygwin window and change the directory to where *DecentralizedDataAggregation* is located as:

```
cd $SHMROOT/tools/DecentralizedDataAggregation
```

Use a USB-MiniB cable to connect an Imote2 to the computer. Run the following command to compile and install:

```
make imote2 usbinstall
```

This command will compile *DecentralizedDataAggregation*, and install the binary file on the connected Imote2. Next, run the following command simply to install the binary file of *DecentralizedDataAggregation* on each subsequent Imote2:

```
make imote2 usbreinstall
```

Once installation is complete, run the following command in the cygwin window:

```
autocomm -d COMy
```

`COMy` represents the highest port listed in the Windows Device Manager. By pressing the <Enter> key a couple of times, a BluSH prompt should appear. If not, check the port

number used for COMy and the connection between the interface board and the computer. Open another cygwin window, and type:

```
autocomm –n –o out.txt COMx
```

`COMx` is the second highest port listed in the Windows Device Manager. This command will save the correlation functions to the output file `out.txt`.

Step 2: Running DecentralizedDataAggregation

Either SetDDANExTParameters or SetDDARDTParameters should be used at the BluSH prompt to provide necessary parameters for data processing.

```
SetDDANExTParameters nfft navg overlap window filter normalize
samplingRate cutoff refChannel channelMask saveData saveCF
```

(Note: this command is all one line.)

where

- `nfft` is the number of FFT points
- `navg` is the number of averaging windows
- `overlap` is the number of overlapping points between consecutive spectral windows
- `window` specifies the spectral window: 1 for Welch window, 2 for Hamming window, and 3 for Hanning window
- `filter` specifies the use of low pass filters: 1 to apply, and 0 not to apply
- `normalize` decides if the mean values of the measured data are subtracted: 1 to subtract, and 0 not to subtract
- `samplingRate` is the sampling frequency in Hz
- `cutoff` is the cutoff frequency for the low pass filter
- `refChannel` is the channel number for the reference signal of the cluster-head used in estimating correlation functions
- `channelMask` specifies sensing channels: 1 for channel 1, 12 for channels 1 and 2, and 123 for channels 1, 2, and 3
- `saveData` determines if the sensor data is collected at the base station for debugging purposes: 1 to collect, and 0 not to collect
- `saveCF` determines if the correlation function is collected at the base station. If the correlation function is desired to be used further in-network analysis, collection at the base station may not be desired. Use 1 to collect, and 0 not to collect.

```
SetDDARDTParameters data_len RDF_len trig_level_a
trig_level_b
```

`samplingRate refChannel channelMask saveData saveRDF` (Note: this command must be all on one line.)

where

- `data_len` is the number of data points in a channel
- `RDF_len` is the number of points in a random decrement function
- `trig_level_a` is the lower bound of the trigger level
- `trig_level_b` is the upper bound of the trigger level
- `samplingRate`, `refChannel`, `channelMask`, and `saveData` are as described previously.
- `saveRDF` determines if the random decrement function is collected at the base station. Use 1 to collect, and 0 not to collect.

Once the data processing parameters are set, the SetDDASensorTopology command should be used to configure the sensor topology, i.e.:

`SetDDASensorTopology <nodeId> <nodeId> ... <nodeId> 0 <nodeId> <nodeId> ... <nodeId> 0 ...` (Note: this command is all one line.)

Setting the sensor topology using SetDDASensorTopology is the same as was done using SetSUSensorTopology, as previously described. Another way to set the sensor topology is to call SetDDASensorTopology for each local sensor community, i.e.:

```
SetDDASensorTopology <nodeId> <nodeId> ... <nodeId>
SetDDASensorTopology <nodeId> <nodeId> ... <nodeId>
...
SetDDASensorTopology <nodeId> <nodeId> ... <nodeId>
```

For example, consider a sensor network that has the following topology:

- Community 1 nodeIds: 21, 87, 32
- Community 2 nodeIds: 51, 8, 21, 87
- Community 3 nodeIds: 2, 112, 42

where nodes 21, 51, and 2 are the cluster-heads for communities 1, 2, and 3, respectively. Thus, the corresponding input command should be:

```
SetDDASensorTopology 21 87 32 0 51 8 21 87 0 2 112 42
```

or

```
SetDDASensorTopology 21 87 32
SetDDASensorTopology 51 8 21 87
SetDDASensorTopology 2 112 42
```

To start *DecentralizedDataAggregation*, input:

```
StartDDA
```

At the completion of *DecentralizedDataAggregation*, '`(*) DDA finished at the base station`' should be seen in the cygwin window. Figure A.3 shows an example for running *DecentralizedDataAggregation*.

Raw sensor and processed data are saved in the output file specified in the second cygwin window using the options `saveData, saveCF, and saveRDF` for the `SetDDANExTParameters and SetDDARDTParameters` commands. The output file includes measured data from each node and processed data from each local sensor community as shown in Figure A.4. If ".m" is appended to the name, then these files can be run directly in MATLAB, resulting in variables in the workspace of the form `x[nodeId]` for measured data, `Rxy[nodeId]` for correlation functions, and `Dxy[nodeId]` for random decrement functions. For example:

- `x69` is a matrix with the measured data from node 69.
- `Rxy69` is a matrix with the correlation functions from the community for which node 69 is the cluster-head.
- `Dxy69` is a matrix with the random decrement functions from the community for which node 69 is the cluster-head.

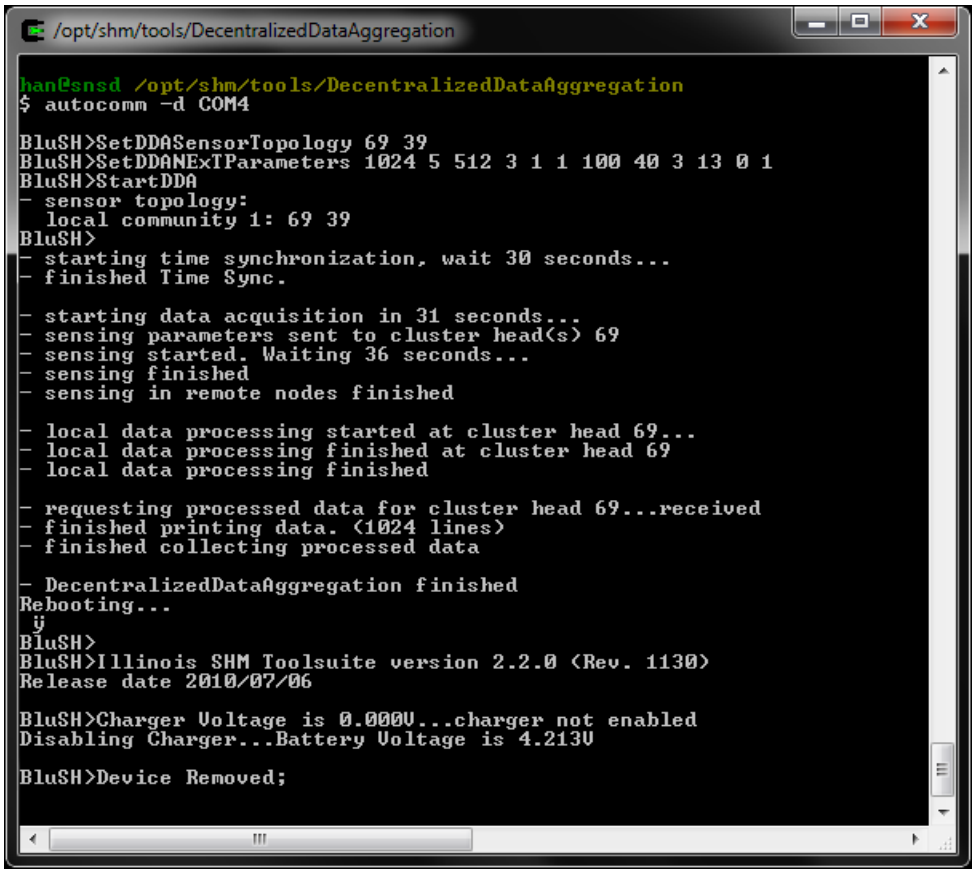


Figure A.3 Input commands in the gateway node for *DecentralizedDataAggregation*.

```
%Time History Data
x69=[
13837  20271
13837  20271
13838  20272
  ⋮      ⋮
13836  20276
];
%Time History Data
x39=[
13814  21497
13812  21498
13812  21502
  ⋮      ⋮
13809  21496
];
%CFE: cluster-head 69
  %node: 69,ch:1  %node: 69,ch:3  %node: 39,ch:1  %node: 39,ch:3
Rxy69=[
  6.19997626e-01  4.30631513e+00  4.23836029e-02  6.82955712e-01
  5.12753638e-01  3.06439151e+00  1.90794679e-02  6.17799897e-01
  3.76651171e-01  1.36226064e+00  9.23906115e-02  4.13684798e-01
        ⋮               ⋮               ⋮               ⋮
  3.80944256e-01  1.39077845e+00  1.24120981e-01  3.89017034e-01
];
```

Figure A.4 Example output file of *DecentralizedDataAggregation*.

RemoteCommand

RemoteCommand developed by K. Mechitov provides an efficient means for the nodes to interact with each other to collaboratively conduct a certain task. *RemoteCommand* provides a fault tolerant and efficient implementation of Remote Method Invocation (RMI) (Waldo 1998) for the ISHMP Services Toolsuite. The most common pattern in WSSN applications is that a node (let say *master* node) sends a command message to other nodes (*slave* node) that in turn perform a task (e.g., time synchronization, sensing, or data processing) and optionally returns results to the master node. *RemoteCommand*, well-suited to the process, can significantly reduce the programming effort by taking advantage of the unique features offered by *RemoteCommand*. This appendix outlines it.

The interface of *RemoteCommand* provides several commands and event handlers. The commands are supposed to be called for a specific purpose in a WSSN application that uses *RemoteCommand*, and the event handler is to be triggered when a certain event happens. Commands and event handlers provided by *RemoteCommand* are:

- command **registerCommand** is used to register a command, specifying initialization parameters.
- command **unregisterCommand** unregisters the command.
- command **executeCommand** is called by the master node, sending a command message to the slave nodes.
- command **executionDone** is called by the slave nodes when the designated tasks are done.
- command **stopCommand** is to stop the command.
- event handler **blushCommandCalled** is triggered when a BluSH command is called.
- event handler **commandSent** is triggered in the master node when the command message is successfully delivered to the slave nodes.
- event handler **responseSent** is triggered in the slave nodes when data is successfully sent back to the master node.
- event handler **commandExecuted** is triggered in both the master and slave nodes when the command **executionDone** is called.

Closely related to each other, the commands and event handlers should be used in a correct order and timing. Figure B.1 shows the control flow of *RemoteCommand*.

(1) **registerCommand** is called in the slave nodes to initialize the *RemoteCommand*.
   - The start function that runs when a command message is received should be specified.
   - Whether data is returned after executing the command or not should be determined.

(2) Calling **executeCommand** in the master node, a command message, possibly parameters for the task of the slave node, is sent.

(3) When the message is sent:

- **commandSent** is triggered in the master node, and the start function runs in the slave nodes. If no return data is specified, **commandExecuted** is triggered as well.
- The start function is used to conduct the task of the slave nodes (e.g., time synchronization, sensing, or data processing).

(4) When the task is finished, the slave nodes should call **executionDone**. Data is sent to the master node if requested to do so.

(5) When the data is sent:
- **commandExecuted** is triggered in the master node.
- **responseSent** is triggered in the slave nodes.

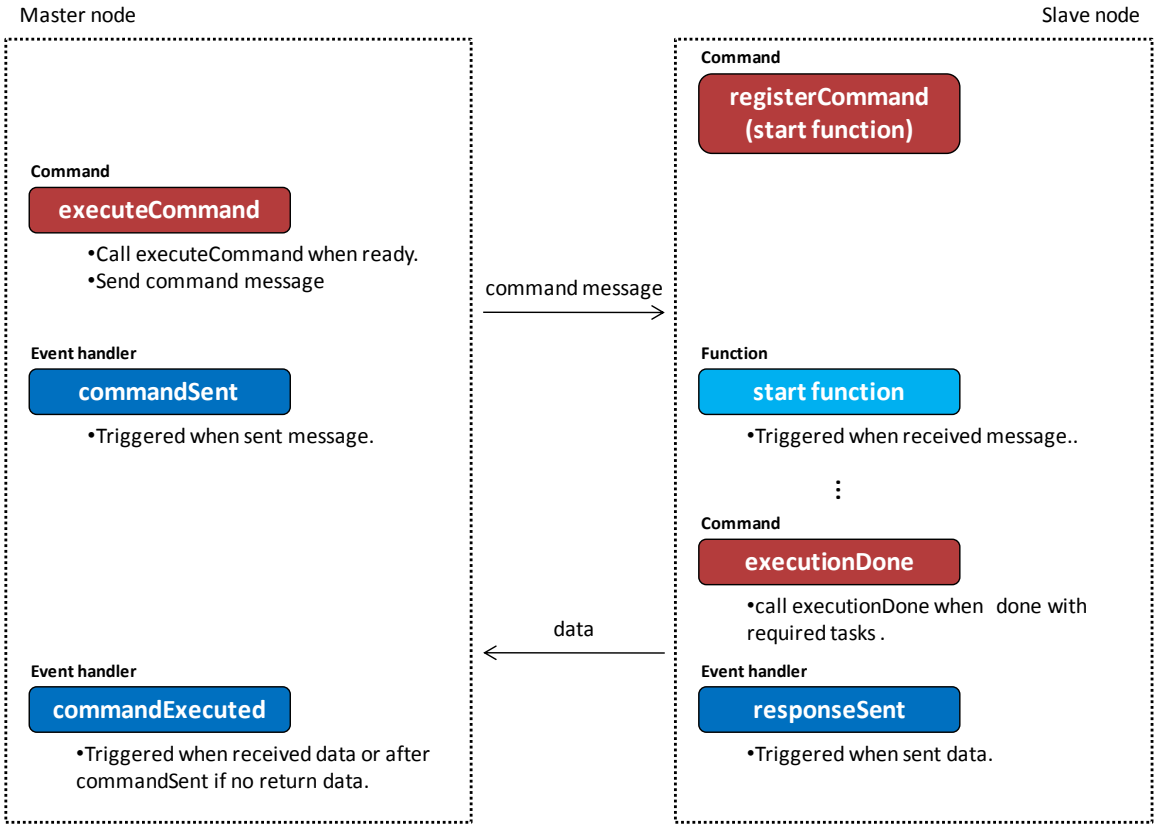


Figure B.1 Flow of *RemoteCommand*.

RemoteCommand is useful for the following examples.

(1) The gateway node sends parameters for sensing, and the leaf nodes start sensing when receiving the parameters.
(2) After sensing, the gateway node sends an acknowledgement, and the leaf nodes send sensor data when receiving the acknowledgement.
(3) The gateway node sends parameters for the power spectral density (PSD) estimation, and the leaf nodes start calculation when receiving the parameters.
(4) After the calculation, the gateway node sends an acknowledgement, and the leaf nodes send the calculated PSD when receiving the acknowledgement.

These four different tasks fundamentally have the same structure from the programming point of view, in that one node (master node) sends a command message (e.g., sensing parameters or acknowledgement) and the other node (slave node) starts a designated task when it receives the command message. *RemoteCommand* is well-suited to carry the tasks that have this type of structure, allowing the four different tasks to be implemented in a straightforward manner.

DECENTRALIZED CABLE TENSION ESTIMATION

Cables of cable-stayed bridges are one of the most critical members for structure health; monitoring tension forces of the cables provides valuable information for SHM of the cable-stayed bridges. Several methods for estimating the cable tension are available including the direct measurement using the load cell, non-contact technique using the electromagnetic (EM) stress sensor (Wang *et al.* 2005), and vibration-based methods. Due to the convenience and cost effectiveness of sensor installation, the vibration-based methods have been recognized to be efficient in practice. As summarized by Kim and Park (2007), the vibration-based methods can be categorized as (1) methods based on the flat taut string theory (sag-extensibility and bending stiffness are not considered), (2) methods based on the modern cable theory (only sag-extensibility is considered), (3) methods based on the equation of motion of a cable that considers only bending stiffness, and (4) methods in which both sag-extensibility and bending stiffness are considered (Zui *et al.* 1996; Mehrabi and Tabatabai 1998; Kim and Park 2007). Examples include Tsing Ma Bridge (Ni *et al.* 2002), Seohae Bridge (Kim *et al.* 2007), and Kao-Ping-His (Fang I.-K. *et al.* 2004).

Cho *et al.* (2010a) implemented a vibration-based method proposed by Zui *et al.* (1996) on smart sensors and experimentally verified in the laboratory using a string with both ends fixed. On the FFT of measured vibration data from the cable, the automated Peak-Picking method finds the first three natural frequencies that are required for Zui's formula. Estimated tension forces are then transmitted to the base station. This earliest study for cable tension estimation using smart sensors has been seen to be promising. However, using Zui's formula in an automated manner can involve difficulties in distinguishing cable dynamics from the cable-deck interaction. In the Jindo Bridge (484m-long cable-stayed bridge), peaks due to the cable-deck interaction are clearly present near the peaks for cable modes in the power spectrum of the cable vibration (Cho *et al.* 2010b), making it difficult to implement an automatic Peak-Picking method.

Here in this study, a closed form relationship between natural frequencies and the tension force is used for cable tension estimation. Assuming the tension force T is constant over the entire cable, the equation of motion of the inclined cable shown in Figure C.1 can be written as (Shimada 1994):

$$\frac{w}{g}\frac{\partial^2 z}{\partial t^2} + EI\frac{\partial^4 z}{\partial x^4} - T\frac{\partial^2 z}{\partial x^2} = 0 \tag{C.1}$$

where z is the deflection in the y-direction, w is the weight density per length, EI is the flexural rigidity, and g is the gravity. With the boundary condition of hinged ends, the solution is:

$$T = \frac{4wL^2}{g}\left(\frac{f_n}{n}\right)^2 - \frac{EI\pi^2}{L^2}n^2 \tag{C.2}$$

or

$$\left(\frac{f_n}{n}\right)^2 = \frac{Tg}{4wL^2} + \frac{EI\pi^2 g}{4wL^4} n^2 \tag{C.3}$$

where L is the length of the cable, f_n is the natural frequency, and n is the order of the natural mode. Linear regression with $(f_n/n)^2$ and n^2 leads to estimation of the tension force T and the flexural rigidity EI.

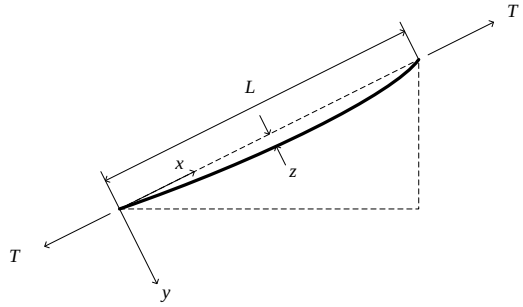


Figure C.1 Inclined cable.

Although this simple approach neglects the sag-extensibility, it can be a practical solution for an automated cable tension estimation system. In addition, because many higher frequency modes are included, reliable estimation can be expected.

List of Recent NSEL Reports

No.	*Authors*	*Title*	*Date*
022	Miller, T.I. and Spencer, B.F.	Solar Energy Harvesting and Software Enhancements for Autonomous Wireless Smart Sensor Networks	March 2010
023	Denavit, M.D. and Hajjar, J.F.	Nonlinear Seismic Analysis of Circular Concrete-Filled Steel Tube Members and Frames	March 2010
024	Spencer, B.F. and Yun, C.-B. (Eds.)	Wireless Sensor Advances and Applications for Civil Infrastructure Monitoring	June 2010
025	Eatherton, M.R.. and Hajjar, J.F.	Large-Scale Cyclic and Hybrid Simulation Testing and Development of a Controlled-Rocking Steel Building System with Replaceable Fuses	Sept. 2010
026	Hall, K., Eatherton, M.R., and Hajjar, J.F.	Nonlinear Behavior of Controlled Rocking Steel-Framed Building Systems with Replaceable Energy Dissipating Fuses	Oct. 2010
027	Yeo, D. and Jones, N.P.	Computational Study on 3-D Aerodynamic Characteristics of Flow around a Yawed, Inclined, Circular Cylinder	Mar. 2011
028	Phillips, B.M. and Spencer, B.F.	Model-Based Servo-Hydraulic Control for Real-Time Hybrid Simulation	June 2011
029	Linderman, L.E., Mechitov, K.A., and Spencer, B.F.	Real-Time Wireless Data Acquisition for Structural Health Monitoring and Control	June 2011
030	Chang, C.-M. and Spencer, B.F.	Multi-axial Active Isolation for Seismic Protection of Buildings	May 2012
031	Phillips, B.M. and Spencer, B.F.	Model-Based Framework for Real-Time Dynamic Structural Performance Evaluation	August 2012
032	Moreu, F. and LaFave, J.M.	Current Research Topics: Railroad Bridges and Structural Engineering	October 2012
033	Linderman, L.E., Spencer, B.F.	Smart Wireless Control of Civil Structures	January 2014
034	Denavit, M.D. and Hajjar, J.F.	Characterization of Behavior of Steel-Concrete Composite Members and Frames with Applications for Design	July 2014
035	Jang, S. and Spencer, B.F.	Structural Health Monitoring for Bridge Structures using Wireless Smart Sensors	May 2015
036	Jo, H. and Spencer, B.F.	Multi-scale Structural Health Monitoring using Wireless Smart Sensors	May 2015
037	Li, J. and Spencer, B.F.	Monitoring, Modeling, and Hybrid Simulation: An Integrated Bayesian-based Approach to High-fidelity Fragility Analysis	May 2015
038	Sim, S-H. and Spencer, B.F.	Decentralized Identification and Multimetric Monitoring of Civil Infrastructure using Smart Sensors	June 2015