Distributed computing strategy for damage monitoring employing smart sensors

B.F. Spencer, Jr.
Department of Civil and Environmental Engineering, UIUC, Urbana, IL 61801, USA
Y. Gao
DMJM Harris, Inc., 20 Exchange Place, New York, NY 10005, USA
G. Yang
Department of Civil and Environmental Engineering, UIUC, Urbana, IL 61801, USA

ABSTRACT: Recently, rapid advances in smart sensor technology have made damage monitoring using a dense array of sensors feasible, and therefore have the potential to change fundamentally the way civil infrastructure systems are monitored. Indeed, a 2002 National Research Council report noted that the use of networked systems of embedded computers and sensors throughout society could well dwarf all previous milestones in the information revolution. However, few damage monitoring algorithms exist that can allow the distributed computing paradigm offered by smart sensors to be employed for structural health monitoring (SHM). This paper begins with a review of current status of smart sensor technology, indicating some of the challenges and opportunities. Subsequently, a new distributed computing strategy (DCS) for SHM is proposed that is suitable for implementation on a network of densely distributed smart sensors. In this approach, a hierarchical strategy is proposed in which adjacent smart sensors are grouped together to form different communities. A flexibility-based damage detection method is employed to evaluate the condition of the local elements within these communities by utilizing only locally measured information. The damage detection results in these communities are then communicated with the surrounding communities; only the essential information is then sent back to a central station. The efficacy of the proposed strategy is experimentally verified using a 14-bay laboratory-scale truss structure.

1 INTRODUCTION

Structural health monitoring has recently attracted a significant amount of interests from researchers with many structural health monitoring (SHM) algorithms being developed, e.g. methods based on the measured frequency (Vandiver 1975, Cha & Tuck-Lee 2000), flexibility-based methods (Pandey & Biswas 1994, Bernal 2002), parameter estimation methods (Yao et al. 1992, Pothisiri & Hjelmstad 2003), neural network based methods (Kudva et al. 1991, Kirkegaard et al. 1995), etc. In general, most of the existing methods require central acquisition and data processing, i.e., the measured data needs to be transferred to a central station, either through wired or wireless communication.

Damage in structures is an intrinsically local phenomenon. Responses from sensors close to the damaged site are expected to be more heavily influenced than those remote to the damage. Therefore, to effectively detect arbitrary damage in a complicated structure, a dense array of sensors distributed over the entire structure will be required. However, if the measured data is to be centrally acquired, significant limitations will exist for conducting SHM either using wired or wireless sensors. Use of traditional wired sensors is difficult due to the cost of deploying and maintaining a large wiring plant. Similarly, transferring all the measured information to a central station using wireless senors is difficult because of power requirements and bandwidth limitations. In both cases, a tremendous amount of data is expected to be generated that would need to be sent to such a central station. Managing this large amount of data is challenging. Unnecessary information needs to be eliminated through data aggregation to efficiently utilize the network.

The recent development of the smart sensors has made health monitoring with a dense array of sensors feasible. A careful review of current smart sensing technology has been provided by Spencer et al. (2004). The essential feature of a smart sensor is the on-board microprocessor, which grants these sensors the "smart" characteristics. A portion of the computation can then be done at the sensor level for damage detection. Extraneous information can be discarded, reducing the information that needs to be transferred back to the central station.

This paper presents a new distributed computing strategy (DCS) for SHM that is suitable for implemen-

tation on a densely distributed smart sensor network. A brief overview of current smart sensor technology is first presented. A flexibility-based damage detection method that forms the basis for the DCS strategy is then reviewed, followed by formulation of the flexibility matrix from both forced and ambient vibration. Extension of a flexibility-based method to handle the continuous online monitoring is detailed. Damage detection using a small group of senors localized in damaged regions is investigated, which leads to the proposed DCS approach for SHM. Finally, to demonstrate the efficacy of the DCS approach, experimental verification using a 14-bay laboratory-scale truss structure is provided.

2 SMART SENSOR TECHNOLOGY

Smart sensor technology has been under rapid development in recent years. A smart sensor usually has four important features: (i) on-board central processing unit (CPU), (ii) small size, (iii) wireless, and (iv) the promise of being low-cost (Spencer et al. 2004). The essential difference between a smart sensor and a standard integrated sensor is its intelligence capabilities, i.e., the on-board microprocessor. Programs can be embedded in the sensor's microprocessor, which allows smart sensors to save data locally, perform desired computation, make "if-then" decisions, scan necessary information, send results quickly, etc. It can also control when and how long the sensor will be fully awake so as to minimize power consumption. The size of smart sensors has been decreasing with time. The use of micro-electromechanical systems (MEMS) has made the dream of having ubiquitous sensing, and in particular small 'smart' sensing, possible. The cost of the smart sensors has been decreasing to a level of tens of dollars, and with their increasing popularity, costs may be reduced to fractions of a dollar. Finally, all smart sensors to date are wireless, with data transmission based on radio frequency (RF) communication.

Some of the first efforts in developing smart sensors for application to civil engineering structures were presented by Straser and Kiremidjian (1996) and Kiremidjian et al. (1997). In their work, a wireless modular monitoring system (WiMMS) for SHM of civil engineering structures was proposed, which provides ease of installation, low cost, portability, and broad functionality. The sensor unit consists of a microprocessor, radio modem, data storage, and batteries. To reduce the battery consumption, the smart sensor can be either in the waiting mode or the operation mode. Numerous researchers have also worked extensively on this research area in the past decade, e.g., Mitchell et al. (1999), Lynch et al. (2001), Liu et al. (2001), etc.

The objective of the recently-created European Energy-Efficient Sensor Networks project (EYES, http://eyes.eu.org/) is to develop the architecture and technology that will enable the creation of a new generation of self-organizing and collaborative sensors. These sensors will be capable of effectively networking together, to provide a flexible platform to support a large variety of mobile-sensor network applications.

Under the substantial support of the US Defense Advanced Research Projects Agency (DARPA), researchers in University of California at Berkeley have developed an open hardware/software platform for smart sensing research. The main objective is to create massively distributed sensor networks, which consist of hundreds or thousands sensor nodes; these nodes have been termed as Smart Dust or Mote. The goal is to have small dust that is fully autonomous and is cubic millimeter in size. The first Mote, termed COTS Dust (Hollar 2000), incorporated communications, processing, sensors, and batteries into a package about a cubic inch in size. The generation of Mote following COTS Dust was Rene, developed in the summer of 2000. The third generation of Mote, the Mica node that improved memory capacity and used a faster microprocessor (4 MHz), was released in 2001 (Hill & Culler 2002). Since then, a series of development has been carried out to improve Mica platform that results in Mica2, Mica2Dot, and MicaZ. The latest generation of the Berkeley Mote is Telos platform, that seeks to achieve three major goals: ultra-lower power operation compared with previous Mote generations, easy to use, and robust hardware and software implementation (Polastre et al. 2005).

TinyOS is a component based operating environment designed for the Berkeley Mote. More specifically, it is designed to support the concurrency intensive operations required by networked sensors with minimal hardware requirements (http://today.cs.berkeley.edu/tos/). The autonomous characteristic of the smart sensor can be realized by developing programs using TinyOS and then running these program on the on-board microprocessor. This software is open source.

Intel Inc. announced the development of the Intel-Mote platform (Kling 2003). The objective is to create a new platform for smart sensing that provides a high level of integration as well as low-power operation and small physical size. This new platform fully supports TinyOS. The Berkeley-Mote and Intel-Mote platforms are expected to be an important impetus for smart sensing software and hardware developments.

While the opportunities offered by smart sensing for SHM are substantial, a number of critical issues need to be addressed before this potential can be realized (Spencer et al. 2004). In particular, effective SHM algorithms that can take advantage of the distributed

computing environment offered by smart sensor technology need to be developed.

3 DAMAGE LOCATING VECTOR METHOD

Damage detection methods based on the flexibility matrix have recently been shown to be quite promising. In particular, Bernal (2002) proposed a flexibility-based damage localization method, the damage locating vector (DLV) method, which provides the foundation for the current work. The basic DLV method has been experimentally verified in Gao et al. (2004) and extended for continuous online monitoring employing ambient vibration in Gao and Spencer (2005). The DLV method as employed herein is briefly reviewed in this section.

3.1 *Basic concept*

The DLV method is based on determination of a special set of load vectors, the so-called damage locating vectors (DLVs). The DLVs have the property that when they are applied to the structure as static forces at the sensor locations, no stress is produced in the damaged elements. This unique characteristic can be employed to localize structural damage.

Assuming nominally linear structural behavior, the flexibility matrices at sensor locations are constructed from measured data before and after damage and denoted as $\mathbf{F}_u$ and $\mathbf{F}_d$, respectively. Then, all of the linear-independent load vectors are collected which satisfy the following relationship

$$\mathbf{F}_d\mathbf{L} = \mathbf{F}_u\mathbf{L} \quad \text{or} \quad \mathbf{F}_\Delta\mathbf{L} = (\mathbf{F}_d - \mathbf{F}_u)\mathbf{L} = 0 \tag{1}$$

This equation implies that the load vectors $\mathbf{L}$ produce the same displacements at the sensor locations before and after damage. From the definition, the DLVs are seen to also satisfy Equation (1); that is, because the DLVs produce no stress in the damaged structural elements, the damage of those elements does not affect the displacements at the sensor locations. Therefore, the DLVs are indeed the vectors in $\mathbf{L}$.

To calculate $\mathbf{L}$, the singular value decomposition (SVD) of the matrix $\mathbf{F}_\Delta$ is employed

$$\mathbf{F}_\Delta = \mathbf{U}\mathbf{S}\mathbf{V}^T = \begin{bmatrix}\mathbf{U}_1 & \mathbf{U}_0\end{bmatrix}\begin{bmatrix}\mathbf{S}_1 & 0\\ 0 & 0\end{bmatrix}\begin{bmatrix}\mathbf{V}_1 & \mathbf{V}_0\end{bmatrix}^T \tag{2}$$

From Equation (2), one obtains $\mathbf{F}_\Delta\mathbf{V}_0 = 0$ and $\mathbf{L} = \mathbf{V}_0$, i.e., DLVs can be obtained from the SVD of the difference matrix $\mathbf{F}_\Delta$.

Each of the DLVs is then applied to an undamaged analytical model of the structure. The stress in each structural element is calculated and a normalized cumulative stress is obtained. The normalized cumulative stress for the *j*th element is defined as

$$\bar{\sigma}_j = \frac{1}{\max_k(\sigma_k)}\sum_{i=1}^{n}\mathrm{abs}\left(\frac{\sigma_{ij}}{\max(\sigma_{ik})}\right) \tag{3}$$

where σ_{ij} = stress in the *j*th element induced by the *i*th DLV; σ_j = cumulative stress in the *j*th element. If an element has zero normalized cumulative stress $\bar{\sigma}_j$, then this element is a possible candidate of damage. In practice, the normalized cumulative stresses induced by the DLVs in the damaged elements may not be exactly zero due to noise and uncertainties.

3.2 *Construction of flexibility matrix*

As shown in the previous section, the flexibility matrices before and after damage need to be constructed to implement the DLV method. Depending on whether or not the input excitations are measured, two approaches have been proposed to construct flexibility matrix from the measured data.

When the input is measured and there is at least one co-located sensor and actuator pair, Bernal & Gunes (2004) have shown how the experimental data can be used to obtain flexibility matrix for structures with general viscous damping. For ambient vibration case, a mass perturbation method suggested by Bernal (2004) can be used to construct the flexibility matrix for a linear structure with classical damping.

3.3 *Extending DLV method for continuous online SHM*

The DLV method is not easily employed for continuous online damage diagnosis. For both forced and ambient vibration cases, a certain degree of interruption of structural operation will be needed to construct the flexibility matrix from measurements. The method based on forced vibration requires employing an external exciter, e.g. an impactor or a rotating imbalance vibrator, to shake the structure. The method based on ambient vibration requires adding mass to the structure to conduct dynamic testing. These methods can be used to construct the undamaged flexibility matrix. However, on-calculation of the damaged flexibility matrix by either of these methods is intractable.

Gao & Spencer (2005) proposed an algorithm that extends the DLV method for continuous online monitoring. The essence of the approach is to construct an approximate flexibility matrix for the damaged structure utilizing the undamaged normalization constants with the mode shapes before and after damage being normalized to have a unit magnitude.

The first step of the proposed algorithm is to compute the undamaged normalization constants and then the undamaged flexibility matrix from the measured data. This step is termed algorithm initialization. Once the initialization is completed, the

second step of the proposed approach is to construct an approximate flexibility matrix for the potentially damaged structure employing ambient vibration and then to apply the DLV method to detect damage in the structure. Herein, this step is referred as algorithm operation. The flow charts for the algorithm initialization and operation are shown in Figures 1 and 2.

For more detailed information regarding the DLV method, the interested reader is directed to Bernal (2002), Gao et al. (2004), and Gao & Spencer (2005).

4 DAMAGE DETECTION USING LIMITED SENSOR INFORMATION

In this section, damage detection using a limited number of sensors localized in damaged regions is investigated. The central question to address is whether local information from the sensors in the vicinity of the damaged region is adequate to locate damage. A numerical example employing a 14-bay planar truss (Fig. 3) is considered in this section. A Matlab finite element model consisting of 53 bars and 28 nodes was developed. Each structural node number has a circle around it.

For the truss shown in Figure 3, sensors on every three consecutive lower chord nodes and the corresponding upper chord nodes are grouped together to monitor the elements connected to these nodes. These groups are formed from left to right in the truss structure. To allow for greater fault tolerance, every two groups have some overlap. For example, group 1 includes nodes [2 3 4 5 6 7], and group 2 includes structural nodes [4 5 6 7 8 9], etc. Therefore, for each group of localized sensors there are a total of 12 sensors which are installed in the *x*- and *y*-direction at six structural nodes. And there are a total of 11 different localized sensor groups.

In this initial example, system identification is not employed; rather, the exact modal parameters (i.e., mode shapes and frequencies) are used to constructed truncated flexibility matrix which is then employed in the DLV method. The flexibility matrix at the sensor locations is constructed using the first four modes. The normalized cumulative stress of all the structural members is computed for each localized sensor group and shown in Figure 4. Results of groups 1 to 11 are shown in the order from top to the bottom. In each plot, a circle is used to mark elements in which both nodes are members of the group, and a cross signifies the elements in which only one of its node is a member of a group. Only the condition of elements in which both nodes are in a given group are evaluated here for damage by the localized sensor group.

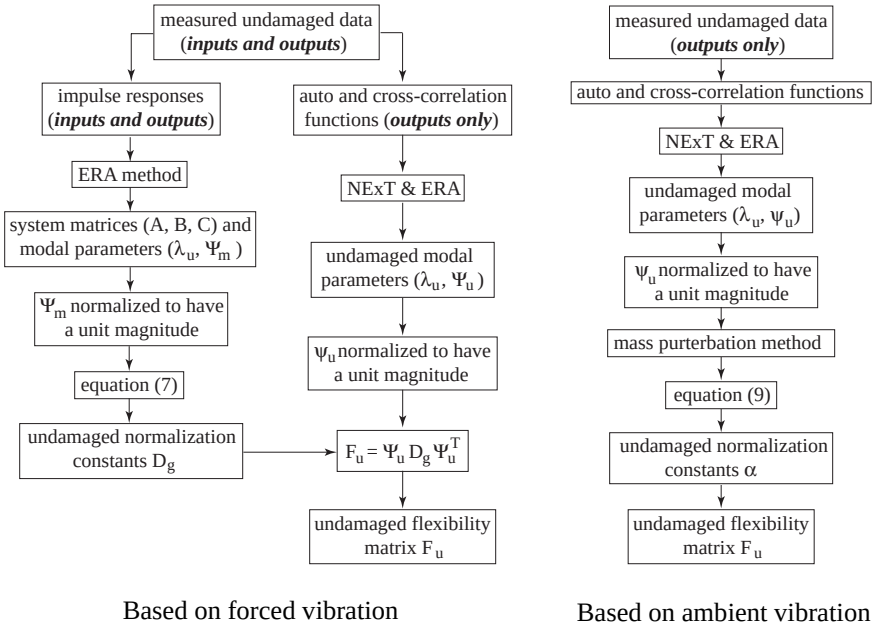


Figure 1. Algorithm initialization.

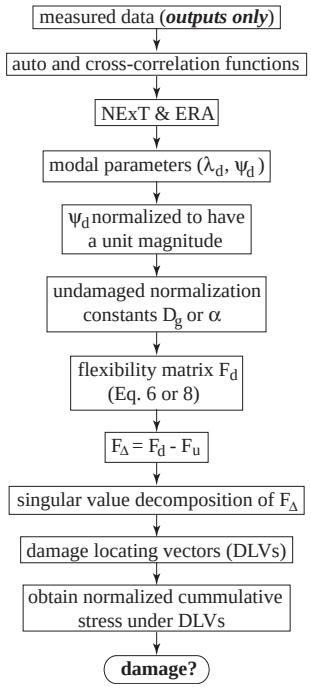


Figure 2. Algorithm operation

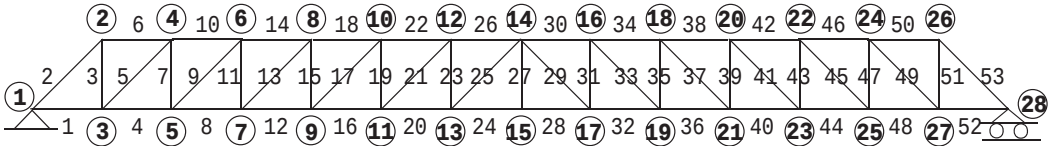


Figure 3. Schematic of the 14-bay laboratory-scale planar truss.

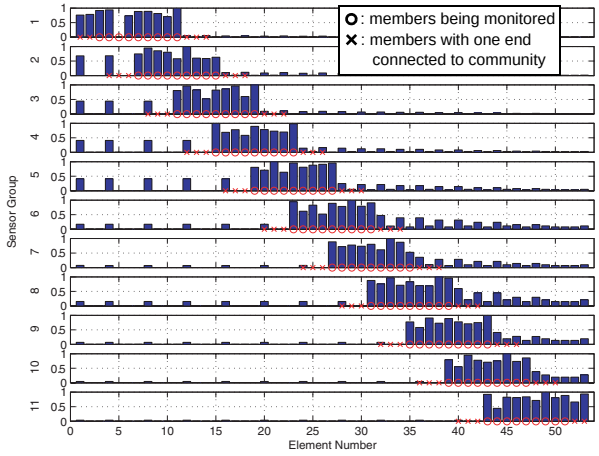


Figure 4. Normalized cumulative stress when element 5 has a 20% stiffness reduction.

Results in group 1 shows that element 5 has a very small normalized cumulative stress compared with other elements in the group, which indicates that element 5 is a possibly damaged element. The results from all other groups indicate that there is no other damaged element in this truss. In this case, damaged element 5 can located based on information of a small group of sensors localized in the damaged region. This result implies that the information away from the damage location might not be necessary for damage detection. This concept can be employed for development of the distributed computing strategy (DCS) detailed in the next section.

5 DISTRIBUTED COMPUTING STRATEGY

5.1 *Hierarchical organization*

The conceptual hierarchical organization of the DCS approach is shown in Figure 5. In contrast to traditional SHM algorithms which require all the measured information to be transferred to a central station, the measured information is aggregated locally by a selected sensor within the sensor group, termed the manager sensor, and only limited information is sent back to the central station to provide the condition of the structure. Small numbers of smart sensors are grouped to form different communities. For clarity, this figure shows each sensor as being included in only one community; however, in the proposed approach, each sensor can participate in

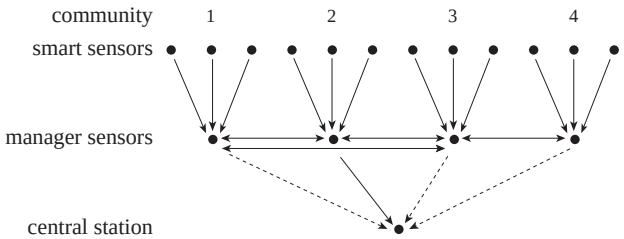


Figure 5. Sketch of hierarchical organization.

multiple communities. For each community, the manager sensor collects measured responses and implements the damage detection algorithm for this community. Adjacent manager sensors need to interact with each other to exchange information. Referring again to Figure 5, manager sensors in communities 1, 2, and 3 interact with each other while community 4 only interacts with community 3.

After the measured information is aggregated locally, the manager sensor determines what information needs to be sent back to the central station for decision-making purposes. In the proposed approach, each of the communities in which damage has not occurred only transmits an “ok” signal to the central station, which is reflected by the dotted line connection in Figure 5. The communities in which damage has occurred need to send information about the damaged elements, which is reflected by the solid line connection in Figure 5. In this way, only limited information needs to be transferred between sensors throughout the entire structure. This approach will significantly reduce the communication traffic and the associated power demands in the sensor network.

5.2 *Strategy implementation*

The planar truss structure shown in Figure 3 is employed to illustrate the details of the implementation of the proposed SHM strategy.

5.3 *Community development*

First, the sensor communities are formed. A single community includes a set of adjacent structural nodes, sensors on these nodes, and members. These structural members have both ends connected to the structural nodes in the same community. Figure 6 shows an example of how communities can be formed.

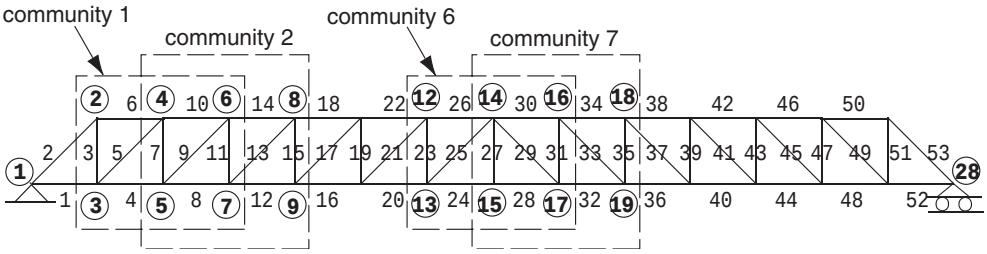


Figure 6. Forming community for damage detection.

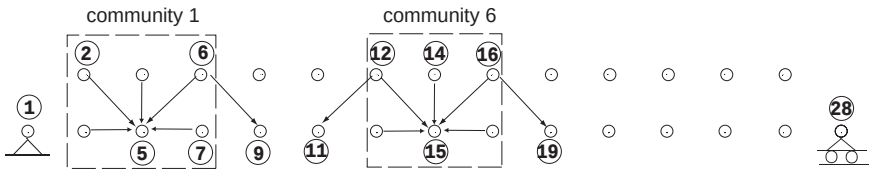


Figure 7. Data collection.

Different communities are developed from left to right in the truss structure. To facilitate efficient communication among the smart sensors, the structural nodes within the same community should be close to each other. As an example, community 6 in Fig. 6 includes nodes [12 13 14 15 16 17] and elements [23 24 25 26 27 28 29 30 31].

Only the elements which have both ends connected to community nodes are monitored by the manager sensor for this community. There is an exception for those communities close to the supports. If a structural member has one end connected to the support and the other end connected to a community node, then this structural member can also monitored by this community. For example, in Figure 6, elements 1 and 2 only have one end connected to the community nodes, but they both have the other end connected to a support; therefore these two elements will be investigated by the smart sensors in community 1.

To allow fault tolerance, adjacent communities are recommended to have some overlaps so that each structural member is monitored by more than a single community.

5.4 *Computing undamaged normalization constants for communities*

Once the community is formed, the modal normalization constants for the undamaged structure have to be computed employing either forced or ambient vibration. For this purpose, one sensor from each community, termed the reference sensor, will be required to send the recording data to the central station. These reference sensors are selected such that their mode shape magnitudes will not be zero for the modes of interest. The modal parameters associated with these reference sensors, denoted as $\hat{\Psi}_j$ or $\hat{\lambda}_j$ and $\hat{\psi}_j$ respectively, are identified using the Eigensystem Realization Algorithm (ERA) (Juang & Pappa 1985) method or the Natural Excitation Technique (NExT) (James et al. 1993) in conjunction with the ERA method. The undamaged normalization constants can then be computed and denoted as $\hat{d}_j$ or $\hat{\alpha}_j$.

However, the normalization constants $\hat{d}_j$ or $\hat{\alpha}_j$ can't be employed directly by each community for damage detection, because the scalar for the *j*th mode shape $\hat{\Psi}_j$ or $\hat{\psi}_j$ and the *j*th mode shape Ψ_j^i or ψ_j^i in *i*th community can be different. The *j*th undamaged normalization constant for the *i*th community d_j^i or α_j^i can be obtained as

$$d_j^i = \hat{d}_j \cdot \left(\frac{\hat{\Psi}_j(i)}{\Psi_j^i(k)} \right)^2 \quad \text{and} \quad \alpha_j^i = \hat{\alpha}_j \cdot \frac{\hat{\psi}_j(i)}{\psi_j^i(k)} \tag{4}$$

where j = *j*th mode; i = *i*th community; and k = reference sensor location in the *i*th community. The undamaged normalization constants for the *i*th community are $\mathbf{D}_g^i = [d_1^i, d_2^i, \ldots d_j^i]$ or $\alpha^i = [\alpha_1^i, \alpha_2^i, \ldots \alpha_j^i]$.

5.5 *Data aggregation*

To minimize the communication traffic in the sensor network, measured data need to be transferred to the manager sensor for data processing.

Clocks of smart sensors in the same community are first synchronized with each other. The measurements are then collected, which will need to be transferred to the manager sensor for computation. To facilitate communication, the manager sensor should be centrally located to the other sensors in the community. For example, in Figure 7, the sensors at structural node 15 is selected as the manager sensor for community 6. As can be seen from the figure, some sensors may need to transfer information to more than one manager sensors. This situation occurs when the smart sensors participate in different communities.

After the data have been transferred to the manager sensor, computation can be conducted using the onboard microprocessor to locate the damage within the community. The extended DLV method presented in this section is incorporated in this SHM strategy to localize the damage in each community.

5.6 *Decision making*

Actions need to be taken after the data aggregation is done for a community. If there is no damage detected in a community, the manager sensor does not initiate interaction with other manager sensors; rather, it simply sends an "ok" signal back to the central station. If there is damage identified in a community, the manager sensor needs to send queries to its counterpart in adjacent communities. There are three possibilities after sending the queries:

- The damage candidate in community *i* does not participate in adjacent communities. The manager sensor in community *i* sends the damage information back to the central station.
- The damage candidate in community *i* participates in other communities and identified as the poten-

tially damaged location in all of these communities. This damage candidate is then confirmed and reported to the central station by these communities.

- The damage candidate identified by community *i* participates in other communities, but not all of the communities identify it as the potentially damaged element. These communities then need to retake data, and re-conduct data aggregation and decision making.

A flow chart for the decision making is shown in Figure 8.

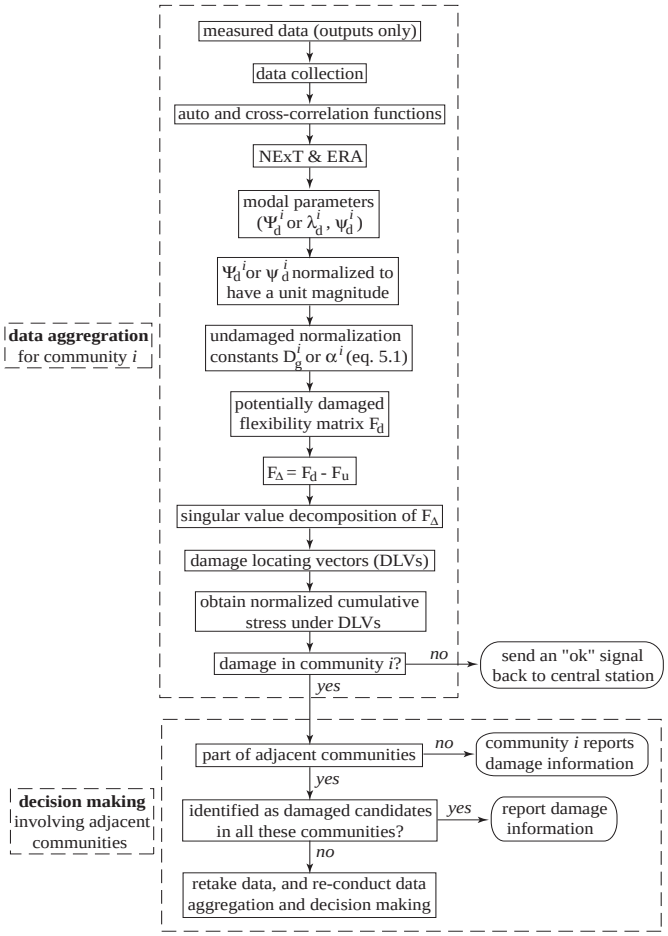


Figure 8. Data aggregation and decision making.

A flow chart for the decision making is shown in Figure 8.

To better illustrate how the proposed decision making algorithm works, a simple example is shown in Figure 9. The manager sensors in each community first conduct damage detection locally. Those communities having no damage send "ok" signal back to the central station. Communities 3, 4, and 8 interact with the surrounding communities because damage is detected in these communities. As a result, communities 3 and 4 confirm that element 16 is the damage location and send this damage information back to the central station. Communities 7 and 8 are required to retake data and re-conduct damage detection as inconsistent information has been obtained regarding element 33. Results regarding element 33 need to be reported to the central station after re-conducting damage detection.

6 EXPERIMENTAL VALIDATION

6.1 *Experimental setup*

The DCS strategy is experimentally verified using the three-dimensional truss structure shown in Figure 10. This structure was tested at the Smart Structures Technology Laboratory (SSTL) of the University of Illinois at Urbana-Champaign (*http://cee.uiuc.edu/sstl/*). The length of each bay of the truss is 0.4 m on each side.

	damage	action		action		action		
		send	receive	send	receive	send	receive	
G1	No	**S "ok" B**	-	SLEEP				→
G2	No	**S "ok" B**	R ?16 G3	S "0" G3	-	SLEEP		→
G3	**Yes**	S ?16 G2	-	-	R "0" G2	**S "D16" B**	-	SLEEP
		S ?16 G4	R ?16 G4	S "1" G4	R "1" G4			
G4	**Yes**	S ?16 G3	R ?16 G3	S "1" G3	R "1" G3	**S "D16" B**	-	SLEEP
		S ?16 G5	-	-	R "0" G5			
G5	No	**S "ok" B**	R ?16 G4	S "0" G4	-	SLEEP		→
G6	No	**S "ok" B**	-	SLEEP				→
G7	No	**S "ok" B**	R ?33 G8	S "-1" G8	-	-	R "C" G8	**P "C"**
G8	**Yes**	S ?33 G7	-	-	R "-1" G7	S "C" G7	-	**P "C"**
		S ?33 G9	-	-	R "0" G9	**S "I33 7-8" B**		
G9	No	**S "ok" B**	R ?33 G8	S "0" G8	-	SLEEP		→
G10	No	**S "ok" B**	-	SLEEP				→
G11	No	**S "ok" B**	-	SLEEP				→

Notes:

G1	Manager sensor in community 1.
S	Send.
R	Receive.
B	Base station.
P	Perform.
SLEEP	Sleep mode.
"-"	Idle or listening mode.
?	Query.
"0"	No. It is not a member of our group.
"1"	Yes. It is a member of our group and is damaged.
"-1"	Yes. It is a member of our group but is NOT damaged.
"D"	Damage information.
"ok"	No damage found in this group.
"C"	Cycle: cycle of retaking data and re-conducting damage evaluation.
"I"	Inconsistent.

Examples

S ?16 G2	Send query about element 16 TO group 2.
S "1" G2	Send signal "1" TO group 2.
S "ok" B	Send ok signal TO base station.
S "D16" B	Send damage information about element 16 TO base station.
S "I33 7-8" B	Send signal to base station indicating that inconsistent information regarding element 33 has been obtained by groups 7 and 8.
S "C" G3	Send "Cycle" signal TO group 3 advising it to retake data and re-conduct damage evaluation.
R ?16 G2	Receive query about element 16 FROM group2.
R "1" G2	Receive signal "1" FROM group 2.
R "C" G4	Receive "Cycle" signal FROM group 4.
P "C"	Perform another cycle of retaking data and re-conducting damage evaluation.

Figure 9. Decision making example (element 16 is consistently identified as having damage in communities 3 & 4; and inconsistent information has been obtained by communities 7 & 8 regarding element 33)

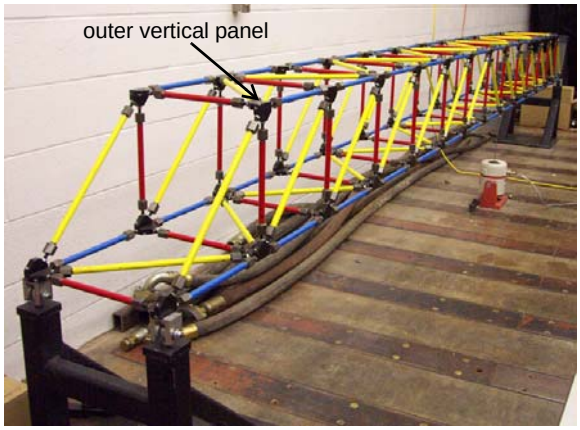


Figure 10. 5.6 m long truss structure

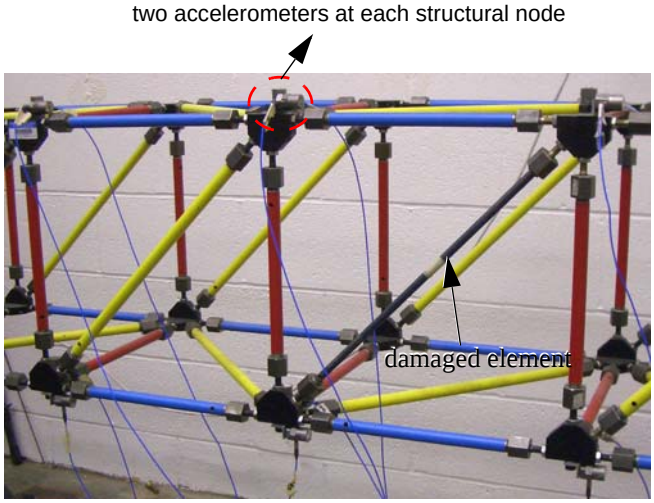


Figure 11. Example of a single sensor community for the outer vertical panel of the truss.

The truss sits on two rigid supports. One end of the truss is pinned to the support, and the other is roller-supported. The pinned end can rotate freely with all three translations restricted. The roller end can move in the longitudinal direction.

The truss members are steel tubes with an inner diameter of 1.09 cm and an outer diameter of 1.55 cm. The joints of the elements are specially designed so that the truss member can be easily removed or replaced to simulate damage without dissembling the entire structure.

A Ling Dynamic Systems permanent magnetic V408 shaker is attached to the bottom nodes of this panel to excite the structure vertically. A load cell with a sensitivity of 20 lb/volt is installed between the shaker and the bottom of the joint to monitor and measure the input to the structure.

Ideally, all the structural members should be monitored for damage. For this paper, only elements in the outer vertical panel of the truss will be monitored employing the proposed DCS approach. A sketch of this panel is depicted in Figure 6 with each node number having a circle around it.

Formation of sensor communities is shown in Figure 6. There are a total of 12 accelerometers in a sensor community with one accelerometer in the vertical direction and the other in longitudinal direction at each structural node. These localized sensors are used to monitor the condition of nine local structural members which have both ends connected to the structural nodes in the same community. Again, there are a total of 11 sensor communities in this experiment.

One of the communities with a single damaged member is shown in Figure 11. Because of limited experimental equipment, these 12 accelerometers shown are moved along the truss structure to simulate a dense array of sensors.

6.2 *Algorithm initialization*

To detect the damage in this truss structure employing the proposed DCS approach, the undamaged flexibility matrix at the sensor communities needs to be constructed. Construction of the flexibility matrix based on both forced and ambient vibration employing experimental data is investigated.

For the case of initialization based on forced vibration, dynamic testing is conducted when seven accelerometers are installed at nodes [7, 11, 13, 17, 19, 21, 23] to measure the acceleration in the vertical direction and the shaker is attached to the bottom of node 17. Therefore, there is one co-located sensor and actuator pair which is at node 17. Modal normalization constants associated with these mode shapes can then be obtained from experimental data.

For the case of algorithm initialization based on ambient vibration, accelerometers were installed at the vertical direction of nodes [7, 9, 11, 13, 15, 17, 19, 21, 23, 25] and at the longitudinal and transverse direction of node 11. A lumped mass of 1.124 kg, which is about 48% of the original nodal mass, was attached to the bottom of node 11 to simulate mass perturbation. The modal normalization constants can be obtained based on the mass perturbation method.

After the modal normalization constants associated with the mode shapes at reference sensor locations were determined, a series of dynamic testing was conducted with 12 sensors installed in each community to obtain the mode shapes in these communities under ambient vibration. The associated modal normalization constants in each of these communities can then be obtained from Equation (4) and the undamaged flexibility matrix at each community can be constructed.

Based on the contribution of the frequency modes to the measurements, different communities might need to utilize different frequency modes to construct the flexibility matrix. Figure 12 shows a total of 12 cross spectral density functions for the communities 4 and 8, respectively. These cross spectral density functions are computed from the measurements of the 12 accelerometers in one community, with the vertical

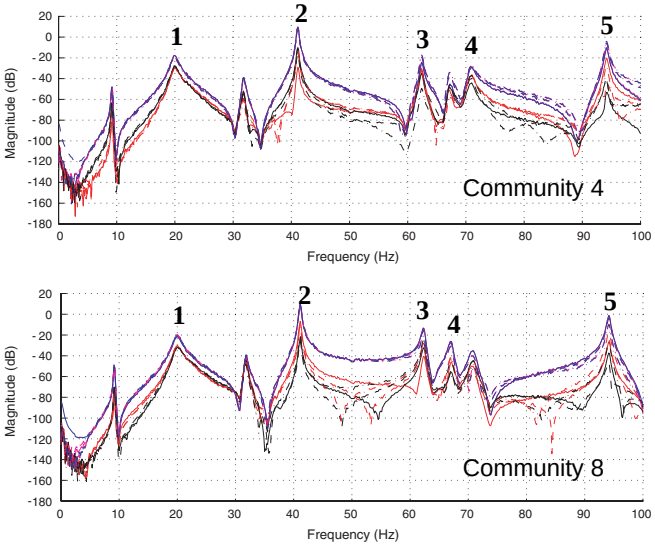


Figure 12. Cross spectral density functions.

acceleration at node 9 being selected as the reference output for community 4 and the vertical acceleration at node 21 being selected as the reference output for community 8. The first 5 dominant modes are identified for both communities 4 and 8. As shown in Fig. 12, the fourth dominant mode is different for these two communities.

In this experimental validation, flexibility matrices in communities 1 through 5 are constructed using the same dominant modes as community 4, and communities 6 through 11 use the same dominant modes as community 8. Therefore, the flexibility matrix in each community is constructed using the first five dominant modes identified from the measurements.

6.3 *Damage detection and decision making*

Once the algorithm initialization is finished, the sensors can start collecting data from the structure to monitor its condition.

A series of damage detection tests have been conducted using the 14-bay three-dimensional truss structure located at SSTL of UIUC. Structural damage is simulated by replacing the original elements with ones having a 52.7% stiffness reduction, which results in a slightly different axial stiffness reduction in different members (see Table 1).

Three excitation conditions (see Table 2) have been investigated. As shown in the table, excitation condition 1 has a same excitation location, magnitude, and bandwidth before and after damage. Under this excitation condition, the shaker is attached to the bottom of node 17 to excite the structure vertically. A band-limited white noise with a 0.1 RMS value of magnitude and a bandwidth of 100 Hz is sent from the Siglab spectral analyzer to drive the shaker.

To better simulate the real ambient vibration condition in which the excitation contents can be changed, performance of the proposed approach under excitation conditions 2 and 3 is also studied. Under condition 2, the magnitude of the band-limited white noise excitation is changed from a RMS value of 0.1 to 0.08, and the bandwidth is changed from 100 Hz to 200 Hz before and after damage. For condition 3, not only are the excitation location and magnitude different, the shaker is also relocated from node 17 to node 23 before and after damage.

Results from three damage cases (see Table 3) are presented in the remainder of the paper. The process of retaking data when there is inconsistent damage information is not investigated in cases 1. Instead it will be studied in cases 2 and 3.

Table 1: Structural damage.

	Longitudinal element	Vertical element	Diagonal element
Cross section reduction	52.7%	52.7%	52.7%
Equivalent axial stiffness reduction	44%	43.5%	46.9%

Table 2: Various excitation conditions before and after damage.

	Location	Magnitude	Bandwidth
Condition 1	same	same	same
Condition 2	same	different	different
Condition 3	different	different	different

Table 3: Damage cases.

Cases	Excitation condition	Damaged elements
Case 1	condition 1	vertical element 7 and longitudinal element 36
Case 2	condition 2	longitudinal element 8
Case 3	condition 3	diagonal element 9

A cut-off value of 0.3 for the normalized cumulative stress is used and provides good performance for a wide range of damage cases by using a total of five DLVs associated with the smallest singular values. This cut-off value will be used to select the damaged elements from the detection results.

Case 1

The detection results regarding damage in vertical element 7 and longitudinal element 36 are shown in Figure 13. The damaged elements have been correctly identified for both cases of initialization based on forced and ambient vibration.

For the case of initialization based on ambient vibration, community 2 identifies element 9 as possibly having damage because its normalized cumulative stress is smaller than the threshold value of 0.3.

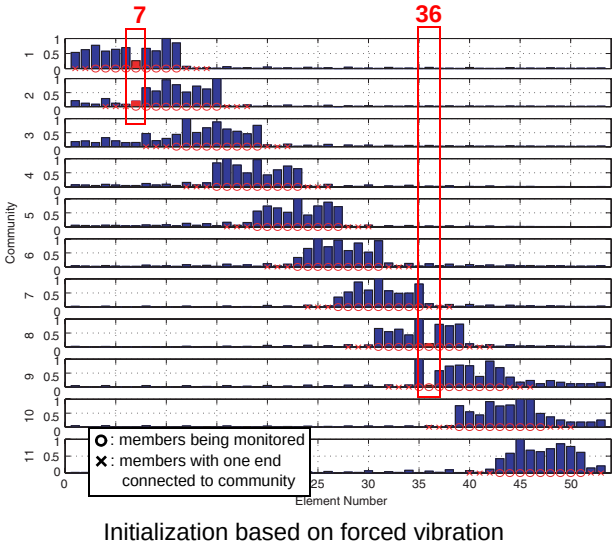


Figure 13. Normalized cumulative stress when elements 7 and 36 are damaged.

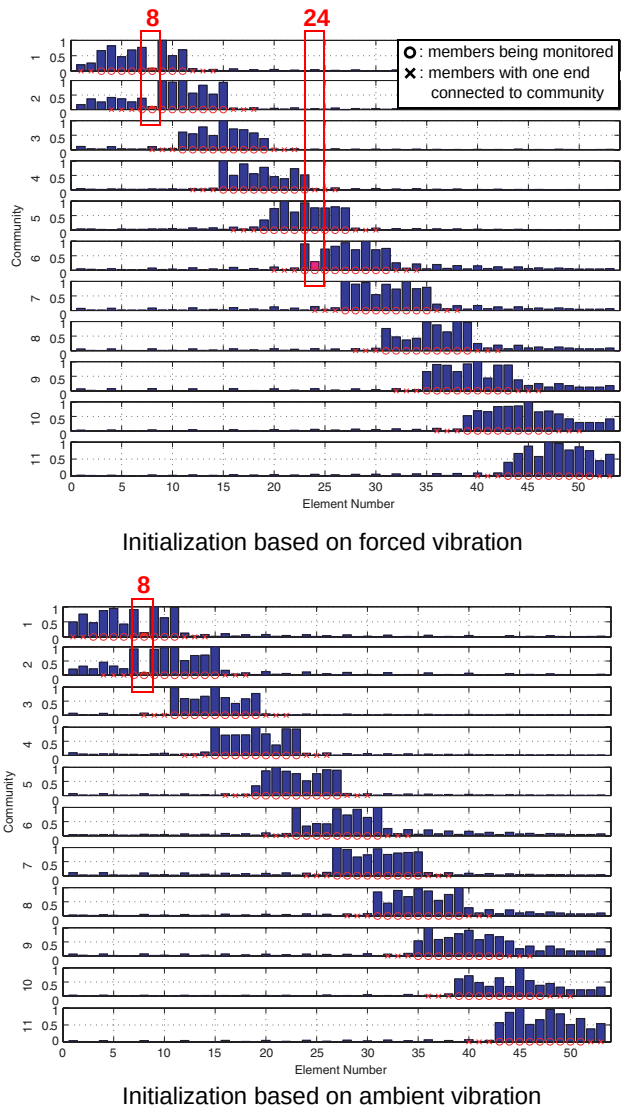


Figure 14. Normalized cumulative stress when element 8 is damaged.

However, community 1 reports this element as undamaged because its stress is larger than the threshold. Inconsistent information regarding the condition of element 9 is obtained in community 1 and 2, therefore these communities need to retake data and re-conduct damage detection. Similarly, communities 9 and 10 need to retake data and re-conduct damage detection regarding element 40. As mentioned above, the process of re-conducting damage detection is not investigated in this case.

Case 2

Damage detection results when longitudinal element 8 is damaged are shown in Figure 14. Element 8 is successfully identified as the damage candidate in both cases as its computed normalized cumulative stress is smaller than the threshold.

For the case of initialization based on forced vibration, community 6 reports that element 24 is damaged as it has a normalized cumulative stress smaller than the threshold value of 0.3. However, community 5 identify element 24 as having no damage as its stress exceeds the threshold. Inconsistent information is obtained by communities 5 and 6 regarding the condition of element 24, therefore these two communities need to retake data and re-conduct damage detection to determine its condition. Figure 15 shows the results after re-conducting damage detection in these two communities, element 24 is determined as undamaged because both communities report that it has a normalized cumulative stress larger than the threshold.

Therefore, only communities 1 and 2 report damage information to the central station, while other communities send “ok” signal back to the central station.

Case 3

Figure 16 shows the damage detection results when element 9 is damaged. In both cases, element 9 is consistently identified as the damage candidate and there is no other damage. Therefore, only communities 1 and 2 report the damage information and other communities only need to send a “ok” signal back to the central station.

These experimental results demonstrate that the proposed DCS approach not only works robustly

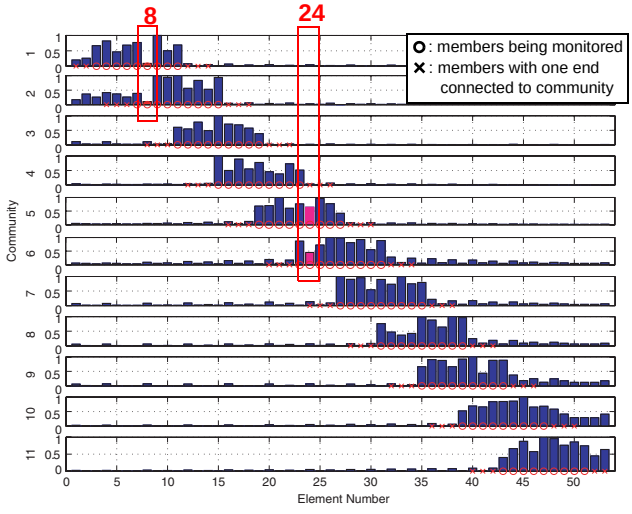
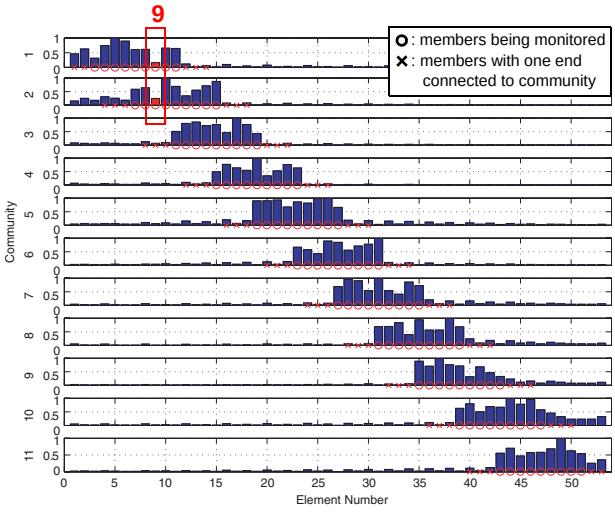


Figure 15. Normalized cumulative stress when element 8 is damaged (retaking data regarding element 24).



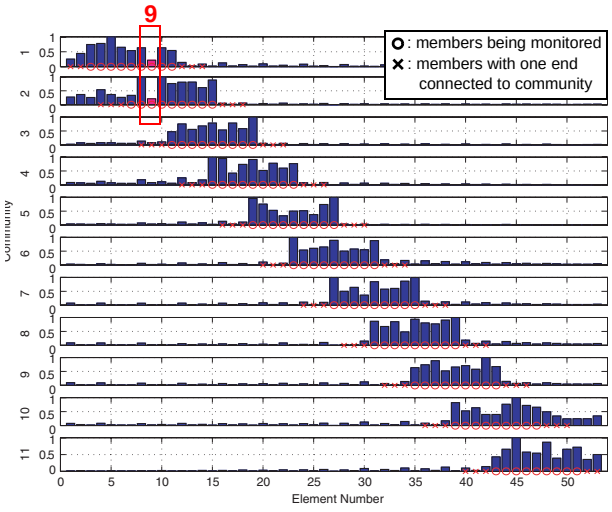


Figure 16. Normalized cumulative stress when element 9 is damaged.

when excitation remains the same before and after damage, it also performs well when excitation contents have been changed. Therefore, the proposed

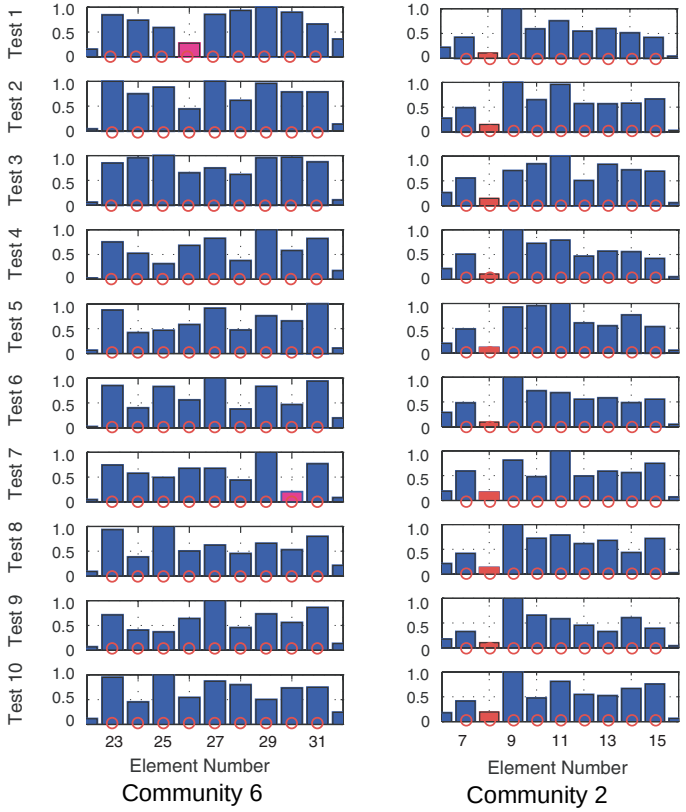


Figure 17. Normalized cumulative stress when element 8 is damaged.

approach is promising for potential application to practice with a dense array of smart sensors.

6.4 *Discussions*

In this experiment, inconsistent damage information is obtained in various damage cases. Questions arise how frequently this situation happens, as well as how reliable the proposed algorithms can detect structural damage. This section provides some insights on these questions.

To investigate the situation of inconsistent damage information, case 2 in which element 8 is damaged is further considered. As shown in Figure 14, for the case of initialization from forced vibration, inconsistent information regarding element 24 is obtained by community 6. For this specific community, a total of ten sequential tests of retaking data for the damaged structure and re-conducting damage detection have been conducted for varying amplitudes and bandwidths. The damage detection results from community 6 are shown in Figure 17. As can be seen, among these ten tests, only tests 1 and 7 reports possible damage information for a different location. Although these results are problem dependent, they indicate that the situation of inconsistent damage information is rather random and does not occur in the majority of the tests.

To provide insight into how reliable the proposed DCS approach can detect the damage, case 2 is considered again. Ten sequential tests are conducted for

community 2 with varying excitation conditions. The damage detection results are displayed in Figure 17, which shows element 8 is identified as the damage candidate. These results indicate the proposed DCS approach can consistently identify the damage location, in this case, element 8.

7 CONCLUSIONS

In this paper, a new distributed computing strategy for SHM is proposed that is suitable for implementation on a network of densely distributed smart sensors. The proposed DCS approach differs from the traditional SHM algorithms because it does not rely on central acquisition and data processing. In the proposed approach, adjacent smart sensors are grouped together to form sensor comminutes to monitor the local elements. Extraneous information is discarded before damage information is sent to the central station, therefore only limited information needs to be transferred wirelessly. Numerical simulations with noise included in measurements have been conducted with band-limited excitation. To better assess the performance of the proposed DCS approach, the magnitude and location of the excitation were changed before and after damage. Both single and multiple damage scenarios were studied, with structural damage being consistently identified using locally measured information. These results indicate that the proposed DCS approach is promising for continuous damage monitoring with a densely distributed smart sensor network. Implementation of the DCS on the Berkeley Mote platform is currently underway.

ACKNOWLEDGEMENTS

The authors gratefully acknowledge the support of the research by the National Science Foundation, under grant CMS 03-01140 (Dr. S.C. Liu, Program Director).

REFERENCES

Bernal, D. 2002. Load vectors for damage Localization. *J. of Engrg. Mech.*128(1): 7–14.

Bernal, D. 2004. Modal scaling from known mass perturbations. *J. of Eng. Mech.* 130(9): 1083–1088.

Bernal, D. & Gunes, B. 2004. Flexibility based approach for damage characterization: benchmark application. *J. of Engrg. Mech.* 130(1): 61–70.

Cha, P.D. & Tuck-Lee, J.P. 2000. Updating structural system parameters using frequency response data. *J. of Engrg. Mech.*126(12): 1240–1246.

Gao, Y., Spencer Jr., B.F. and Bernal D. 2004. Experimental verification of the damage locating vector method. *J. of Eng. Mech.*, accepted.

Gao, Y. & Spencer Jr., B.F. 2005. Online damage diagnosis of civil infrastructure employing a flexibility-based approach. *J. of Smart Materials and Structures*, accepted.

Hill, J. & Culler, D. 2002. Mica: a wireless platform for deeply embedded networks. *IEEE Micro* 22(6): 12–24.

Hollar, S. 2000. "COTS Dust". Master of Science Thesis, University of California at Berkeley.

James, G.H. III, Carne, T.G. and Lauffer, J.P. 1993. The natural excitation technique (NExT) for modal parameter extraction from operating wind turbines. SAND92-1666, UC-261, Sandia National Laboratories.

Juang, J.N. & Pappa, R.S. 1985. An eigensystem realization algorithm for modal parameter identification and model reduction. *J. of Guidance Control and Dynamics*, 8: 620–627.

Kiremidjian, A.S., Straser, E.G., Meng, T.H., Law, K. and Soon, H. 1997. Structural damage monitoring for civil structures. *International Workshop – Structural Health Monitoring*, 371–382.

Kirkegaard, P., Nielsen, S. and Hansen, H. 1995. Identification of nonlinear structures using recurrent neural networks. *Proc. of the 13th International Modal Analysis Conf.*, 1128–1134.

Kling, R. 2003. Intel Research Mote. Intel Corporation Research, Santa Clara, California.

Kudva, J., Munir, N. and Tan, P. 1991. Damage detection in smart structures using neural networks and finite element analysis. *Proc. of ADPA/AIAA/ASME/SPIE Conf. on Active Materials and Adaptive Structures*, 559–562.

Liu, R.C., Zhou, L., Chen, X. and Mau, S.T. 2001. Wireless sensors for structural monitoring. *Strong Motion Instrumentation for Civil Engineering Structures*, 253–266.

Lynch, J.P., Law, K.H., Kiremidjian, A.S., Kenny, T.W., Carryer, E. and Partridge, A. 2001. The design of a wireless sensing unit for structural health monitoring. *3rd International Workshop on Structural Health Monitoring*, Stanford, CA, 1041–1050.

Pandey, A.K. & Biswas, M. 1994. Damage detection in structures using changes in flexibility. *J. of Sound and Vibration*, 169(1): 3–17.

Polastre, J., Szewczyk, R. and Culler, R. 2005. Telos: enabling ultra-low power wireless research. *Proc. of the Fourth Int. Conf. on Infor. Proc, in Sensor Networks: Special track on Platform Tools and Design Methods for Network Embedded Sensors (IPSN/SPOTS)*, April 25–27.

Pothisiri, T. & Hjelmstad, K.D. 2003. Structural damage detection and assessment from modal response. *J. of Engrg. Mech.*, 129(2): 135–145.

Spencer Jr., B.F., Ruiz-Sandoval, M.E. and Kurata, N. 2004. Smart sensing technology: opportunities and challenges. *J. of Structural Control and Health Monitoring*, 11: 349–368.

Straser, E.G. & Kiremidjian, A.S. 1996. A modular visual approach to damage monitoring for civil structures. *Proc. of SPIE v2719, Smart Structures and Materials*, 112–122.

Vandiver, J.K. 1995. Detection of structural failure on fixed platforms by measurement of dynamic response. *Proc. of the 7th Annual Offshore Technology Conf.*, 243–252.

Yao, G.C., Chang, K.C. and Lee, G.C. 1992. Damage diagnosis of steel frames using vibrational signature analysis. *J. of Engrg. of Mech.*, 118(9): 1946–1961.