

# Hierarchical PSD damage detection methods for smart sensor networks

R.K. Giles & B.F. Spencer, Jr.

*University of Illinois at Urbana-Champaign, USA*

**ABSTRACT:** Structural health monitoring (SHM) will transform the management and maintenance of civil infrastructure as available technology and methods continue to improve. Realizing the full potential of SHM requires the development of dense arrays of multi-scale sensors running robust and efficient programs. However, the scale of a densely instrumented civil infrastructure will generate vast amounts of data. Organizing the sensors in a hierarchical computing environment will store and process more data locally on each smart-sensor. This organization will greatly reduce the amount of data broadcast to a base station or multi-hopped along the network, thereby reducing its power consumption. However, because hierarchical computing only shares data locally, damage detection algorithms need to effectively perform in the presence of this constraint. This paper examines a damage detection algorithm that analyzes the changes in the structure's Power Spectral Density (PSD) in a hierarchical, distributed computing environment. The method is model-independent, requiring only output measurement. The effect of group size, sensor overlap, and frequency range are considered using numeric simulation. The results show that limitations exist for these variables to maintain the functionality of the damage detection algorithm. Nevertheless, when the hierarchical distribution properly addresses these limitations, the proposed algorithm is effective in accurately detecting damage.

*Keywords:* hierarchical computing, smart sensor, damage detection.

## 1 INTRODUCTION

Structural health monitoring (SHM) systems include networks of sensors used to monitor the condition, and thus the safety, of structures. Once built, civil infrastructure immediately begins to age and deteriorate due to its constant exposure to the elements. Systems of manual inspections exist to ensure the public safety; but, they are costly in terms of human and financial capital and their effectiveness is dependent on the skill and thoroughness of the inspector. Installing a SHM system could improve the efficiency of the inspection process, increase the accuracy of the evaluation, and thus improve the safety levels of the structure (Sohn et al. 2003, Doebling et al. 1996). The effectiveness of an installed SHM system depends on the type of sensor network installed, the computational model employed, and the damage detection algorithm implemented.

Ideally, a SHM system would consist of densely-distributed, multi-scale sensors performing various algorithms to verify and refine the identification, localization, and quantification of damage. However, the scale of civil infrastructure poses many chal-

lenges in developing the ideal SHM system. Two general categories of sensor networks exist: wired and wireless. A bridge instrumented with a wired network would require considerable lengths of cable to provide power to and communicate with the sensors. A wireless network eliminates the long cable lengths but has a limited power supply. Both wired and wireless networks have relied on a traditional computing paradigm where neither data storage nor calculation occurs at the sensor location; rather, the sensors transmit the data back to a base station for storage and computation. For small-scale deployments, this strategy is sufficient. However, a large-scale dense network would be unwieldy with such a centralized approach. An alternative is a smart sensor network where each sensor has an on-board microprocessor and a wireless communication link. A smart sensor network can provide advantages in data aggregation, computation, and transmittance, as well as providing a more fault-tolerant approach to SHM. This new approach is not without problems. A principal task is to develop effective algorithms that function within the distributed computing environment of smart sensor networks. A smart algorithm

requires that damage detection occur hierarchically where only limited, local information is available and the computation can occur on the sensors themselves (Gao et al. 2006, Nagayama et al. 2006).

This paper presents numerical simulations that explore a model-independent, hierarchical, distributed-computing, damage-detection algorithm. Experimental variations in group size, multiple damage locations, group overlap, and frequency content are considered to examine the limits of each factor on the effectiveness of the proposed algorithm.

## 2 DAMAGE DETECTION USING POWER SPECTRAL DENSITY METHODS

### 2.1 Power Spectral Density Methods

The proposed approach is an extension of the work of Beskhyroun et al. (2004, 2005) that examined the changes in the Power Spectral Density (PSD) of the structure induced by damage (Beskhyroun et al. 2005a,b 2006). For completeness, this section summarizes this work.

The theory behind the PSD damage detection algorithms holds that changes in the operational mode shapes, or changes in their curvatures, correspond to changes in the structure due to damage. Beskhyroun et al. (2005a, 2006) presented two variations on the PSD methods. Both the Absolute Difference PSD (ADPSD) Method and the Curvature Difference PSD (CDPSD) Method have the same initial calculations. The first step is to calculate the PSD,  $G_i(f)$ , via

$$G_i(f) = \frac{2}{T} E \left[ |X(f)|^2 \right] \quad (1)$$

where  $X(f)$  is the Fourier transform of the measured acceleration,  $f$  is frequency in Hertz,  $E[\cdot]$  is the expected value operator,  $T$  is the measured record length, and  $i$  is the sensor node number. Then, the methods normalize the PSD using the sum of the squares of the PSD evaluated at each frequency

$$P(f) = G(f) / \sqrt{\sum_{i=1}^n G_i(f)^2} \quad (2)$$

where  $P(f)$  is the normalized PSD and  $n$  is the number of nodes. Normalizing the PSD serves a dual purpose. First, it allows for the use of different excitation events in the analysis. Second, it equalizes the importance of each mode shape by effectively eliminating the modal contribution factors. This normalization of the modal contributions suggests that the more modes shapes included in the analysis, the more accurate the algorithm will become.

### 2.2 Absolute Difference PSD Method

After normalization by Equation 2, the ADPSD and the CDPSD have divergent calculation paths. The

ADPSD immediately determines the change in the normalized PSD,  $\Delta P_i(f)$ , using the undamaged normalized PSD,  $P_i^u(f)$ , and the damaged normalized PSD,  $P_i^d(f)$ , via

$$\Delta P_i(f) = \left| P_i^u(f) - P_i^d(f) \right| \quad (3)$$

Thresholding then eliminates all but the maximum change in the normalized PSD value for each frequency. Thus, the matrix  $\Delta P^{MAX}$ , representing the maximum change in the normalized PSDs, has a single non-zero value for each frequency but may have many non-zero values for each node, to wit

$$\Delta P^{MAX} = \begin{bmatrix} 0 & \Delta P^{MAX}(f_2) & \Lambda & 0 \\ \Delta P^{MAX}(f_1) & 0 & \Lambda & \Delta P^{MAX}(f_3) \\ M & M & O & M \\ 0 & 0 & \Lambda & 0 \end{bmatrix} \quad (4)$$

$$C^{\Delta P} = \begin{bmatrix} 0 & 1 & \Lambda & 0 \\ 1 & 0 & \Lambda & 1 \\ M & M & O & M \\ 0 & 0 & \Lambda & 0 \end{bmatrix} \quad (5)$$

where  $m$  is the frequency index. At the same time as the thresholding, a counting matrix,  $C^{\Delta P}$ , is established by replacing each non-zero value in  $\Delta P^{MAX}$  with a 1. These matrices are then summed for each node to create a vector with length equal to the group size. After computing the standard deviation for each vector, twice the respective standard deviation is subtracted from each value to form a normalized nodal damage quality vector,  $\hat{P}$ , and a normalized nodal damage quantity vector,  $\hat{C}$ , where all negative values are eliminated via

$$\hat{P}_i = \sum_{f=f_i}^{f_m} \Delta P_i^{MAX}(f) - 2\sigma_P \quad (6)$$

$$\hat{C}_i = \sum_{f=f_i}^{f_m} C_i^{\Delta P}(f) - 2\sigma_C \quad (7)$$

where  $f_m$  is the last frequency in the PSD,  $\sigma_P$  is the standard deviation of  $\Delta P^{MAX}$ , and  $\sigma_C$  is the standard deviation of  $C^{\Delta P}$ . The Accumulated Damage Index (ADI),  $D$ , is then computed by multiplying the nodal values of the  $\hat{P}$  and  $\hat{C}$  values together.

$$D_i = \hat{P}_i \hat{C}_i \quad (8)$$

### 2.3 Curvature Difference PSD Method

After the normalization in Equation 2, the CDPSD fits the normalized PSD,  $P(f)$ , to a series of cubic splines with periodic end conditions for each frequency. This step allows the second derivative of the PSD – the curvature of the operational mode shapes – to be determined. After computing the undamaged and damaged curvatures, they are used to compute  $\Delta P''(f)$ , the change in the curvature of the normalized PSD via

$$\Delta P_i'' = |P_i''^u(f) - P_i''^d(f)| \quad (9)$$

where  $P''^u$  is the undamaged normalized PSD curvature and  $P''^d$  is the damaged normalized PSD curvature. Before thresholding,  $\Delta P''(f)$  is normalized as a sample population using the mean and standard deviation of the sample, to wit

$$\tilde{P}_i(f) = \frac{\Delta P_i(f) - \Delta \bar{P}(f)}{\sigma_p(f)} \quad (10)$$

where  $\tilde{P}(f)$  is the renormalized PSD curvature difference matrix,  $\Delta \bar{P}(f)$  is the mean of nodal values for each frequency, and  $\sigma_p(f)$  is the standard deviation for each frequency.  $\tilde{P}_i(f)$  is then thresholded and counted by eliminating all values that are less than a percentage,  $\alpha$ , of the maximum  $\tilde{P}_i(f)$  value,  $\tilde{P}_{max}(f)$ . A counting matrix,  $C(f)$ , is also formed at this point as in Equation 5 according to the following criteria.

$$\text{if } |\tilde{P}_i(f)| < |\alpha \tilde{P}_{max}| \text{ then } \tilde{P}_i(f)=0 \text{ and } C_i(f)=0 \quad (11)$$

$$\text{if } |\tilde{P}_i(f)| \geq |\alpha \tilde{P}_{max}| \text{ then } \tilde{P}_i(f)=\tilde{P}_i(f) \text{ and } C_i(f)=1 \quad (12)$$

In this paper,  $\alpha$  was set to 57.5% to achieve suitable results. The  $\tilde{P}(f)$  and  $C(f)$  matrices are then summed for each nodal value to create vectors with length equal to the group size.

$$A_i = \sum_{f=f_i}^{f_m} |\tilde{P}_i(f)| \quad (13)$$

$$O_i = \sum_{f=f_i}^{f_m} C_i(f) \quad (14)$$

Finally, these values are multiplied together at each node to produce the ADI matrix,  $D$ .

$$D_i = A_i O_i \quad (15)$$

## 2.4 Proposed Algorithm

Modifications to the PSD methods are necessary to accommodate the requirements of operating in the distributed computing environment intrinsic to full-scale smart sensor networks. Rather than the data collected by every sensor being available for calculation, the sensors arrange themselves in data-clusters such that a manager sensor will execute algorithms using only the data available within the clusters. When the unaltered PSD methods are applied to the data-clusters, the significant variation in the magnitudes of the ADI make it difficult to determine whether damage or noise is responsible for the given results. Normalizing the ADIs of both the ADPSD and CDPSD by dividing each  $D_i$  from (6) and (11) by the respective sum of  $D_i$  for the group will eliminate the variations in magnitude. The normalization allows for the establishment of a uniform set of criteria to be programmed into the algorithm to detect damage even if group size varies. The uniform set of damage criteria improves the robustness

and fault tolerance of the smart sensor network by allowing truly dynamic assignment or reassignment of clusters if the system detects a sensor failure. However, the normalization process also eliminates the ability of the method to determine the magnitude of the damage.

ADPSD damage indication has one condition:

- 1  $D_{max} > 0.02$  where  $D_{max}$  is the maximum value excluding the  $D_i$  at the end nodes

CDPSD damage indication must satisfy the following two conditions:

- 1  $D_{ma} > 1.5/n$  where  $n$  is the length of the group
- 2  $D_{ma} < 1.5/n$  where  $n$  is the length of the group and  $D_{end}$  is either end node.

Note that the criteria exclude the cluster's end nodes from the calculations because the end nodes return false positives for damage. These false positives are remnants of the clustering process and its affect on the normalization constants and different virtual boundary conditions for the group because of the spline curve fitting in the CDPSD method.

Using the ADPSD and CDPSD in sequence creates a combined algorithm more robust in locating structural damage than either component. Both individual methods must yield positive damage identification results for the algorithm as a whole to return a positive damage result. Using the methods in sequence to form a single algorithm takes advantage of the unique characteristics and abilities of each individual procedure. The ADPSD requires less calculation, and therefore power consumption, because it does not require spline fitting and differentiation.

Its less intensive damage detection calculations allow the node to return to a lower power state, without performing the CDPSD, if the method detects no damage. The ADPSD is very sensitive to damage and though at times yields false positives, indicating damage is present when none exists, it does not yield false negatives that would indicate the dangerous situation where no damage is detected when damage exists. The sensitivity of the ADPSD to the presence of damage comes at the expense of its ability to locate effectively the damage location. The ADPSD often gives a singular strong damage indication on one sensor when the damage is located between two sensors that should both experience the effects of damage. Therefore, to both confirm and better locate the damage between two sensors, once the ADPSD has indicated its presence, the smart-sensor will perform the more calculation intensive CDPSD. Using the CDPSD on its own is not advisable because of its calculation intensity and its propensity for indicating damage when no damage is present in the hierarchical environment due to spline fitting residuals.

The final step in the algorithm is to combine the results of the two PSD methods into a final damage indicator and locator. To do so, the two normalized ADIs from both algorithms are average to form a

Combined ADI. Throughout the algorithm, the cluster heads perform the calculations locally without needing to communicate with sensors outside the designated group. However, once the algorithm detects damage, the cluster heads can communicate with adjoining and overlapping clusters heads to confirm the presence and location of the damage. If they confirm each other's results, they only need to transmit a message as small as a single bit indicating damage and two bytes containing the numbers of the affected nodes to the base station. Transmitting information to additional nodes and base stations only when damage is present improves the power efficiency of the network without sacrificing confidence in the reported results.

### 3 NUMERIC EXAMPLES

#### 3.1 Mathematical Model

A MATLAB finite element model of a planar truss served to analyze the proposed algorithm. As illustrated in Figure 1, the truss contains 53 frame elements arranged in fourteen bays with a centerline axis of symmetry and simple supports. Applied vertically at node 7, a band limited white noise served as excitation to simulate ambient vibration. In generating this white noise, a different seed began the random generation sequence to simulate acceleration records taken at two distinct times under different ambient excitation for the damaged and undamaged case. The simulation recorded the vertical acceleration at the 15 sensor locations shown in Figure 1. The analysis used two cutoff frequencies, 190 Hz and 500 Hz, with appropriate sampling rates and filters to prevent aliasing. The calculations used 1024 FFT points.

#### 3.2 Investigated Damage Cases

Examining the effects of cluster size, node overlap, and frequency content on the performance of the proposed algorithm required the analysis of several damage cases. Reducing the cross-sectional area of designated members in the MATLAB model served to simulate damage in the truss structure. For illustrative purposes, this paper presents only cases that simulate only a 5% reduction in cross-sectional area of members 9 and 33 as shown in Figure 1. For each damage case presented, the cluster size and node overlap between the clusters of the groupings were varied. For each case, the algorithm was performed

Table 1. Investigated Damage Cases

Case	No. of Groups	Size	Overlap
I	1	15	0
II	3	7	3
III	12	4	3

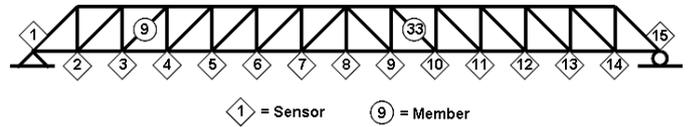


Figure 1: 14-Bay Planar Truss

using cutoff frequencies of both 190 Hz and 500 Hz to examine the inclusion of higher modes in the calculations. In the damage cases, the groups are numbered from left to right, include the indicated number of sensors, and overlap the indicated number of sequential sensors from the previous group. Table 1 lists all damage cases presented in this paper.

## 4 ANALYSIS AND RESULTS

### 4.1 Case I

Test Case I examined the functionality of the proposed algorithm in the standard non-hierarchical computing scenario. This case uses all the sensors, and therefore all the information available, in a single group. However, beyond just the proof of concept, it shows the proposed algorithm's ability to detect multiple damage sites within a group. Figures 2 and 3 show that using cutoffs of both 190 Hz and 500 Hz the method can detect the multiple damage sites on the structure and within the group. In reading these figures, the three graphs represent the output of the ADPSD, CDPSD, and Combined ADI from top to bottom respectively. The Combined ADI

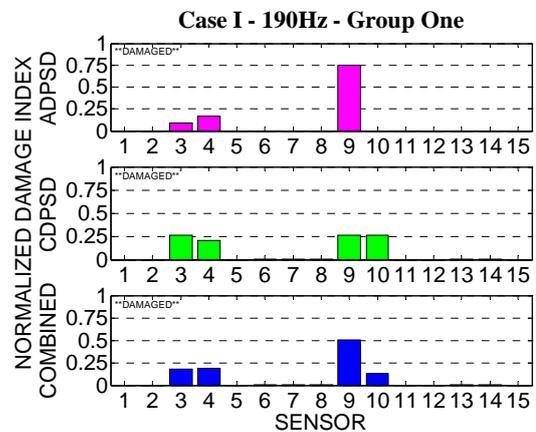


Figure 2. Case I – Group 1 – 190 Hz

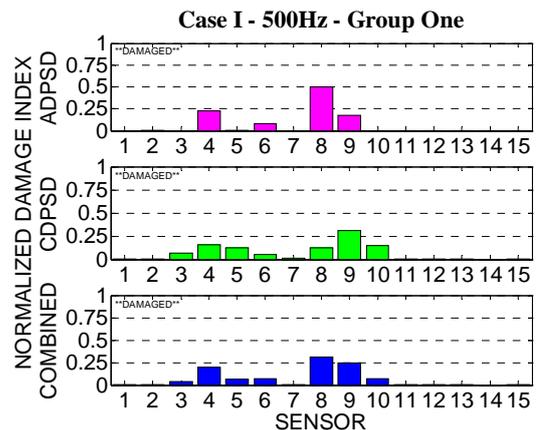


Figure 3. Case I – Group 1 – 500 Hz

is the actual result of the proposed algorithm, but the ADPSD and CDPSD are included for illustrative purposes.

Comparing the results of the algorithm at the different cutoff frequencies in Figures 2 and 3 shows the influence of the higher modes in the algorithm's effectiveness. The higher frequencies tend to have a greater influence on the algorithm because the changes in the higher operational modes are more dramatic and therefore survive the thresholding methods in greater number. However, when more frequencies are included the changes in operational mode shapes caused by the damage begin to occur less locally and the damage indicators to spread around the damage location. This result indicates that when selecting a sampling rate and cutoff frequency, attention to which modes are included is important. This issue does not make the method model dependent, as an updated and accurate model is still not necessary for its functionality. However, having a simple model to help determine a suitable cutoff frequency can improve its efficacy.

#### 4.2 Case II

This computational scenario divides the sensors into three groups of seven sensors each with three sensors overlapping the groups. This separates the damaged locations from each other and places it on the boundary of two groups. The method successfully locates the damage when the damage location is not located on the edge of the group. This is true for both cutoffs, though the diffusion of the damage indicator away from the damage location at the 500 Hz cutoff is greater than in Case I. Figure 4 showing Group 2 of the 190 Hz cutoff serves as the illustrative example of the method's success. Damage is successfully located between sensor 9 and 10.

Figure 4 also illustrates how the CDPSD serves as a check to the ADPSD. By itself, the ADPSD in Figure 4 indicates damage only at sensor 9. This result could indicate that the vertical member above sensor 9 is damaged. However, this damage scenario would still affect the neighboring sensors due to the stress distributions and asymmetry of the truss. Together, the CDPSD and ADPSD detected damage and properly located the bay in which it occurs.

However, Figure 5 shows the failings of using the CDPSD on its own in a hierarchical environment. When damage is located at the end of the group, the methods, even when used together as proposed, have difficulty detecting it. Figure 5 shows that the ADPSD indicates no damage at all because the end nodes, where the damage is located, are eliminated from the analysis. The elimination is necessary because the ends always include a large amount of indicated damage due to the virtual cut in the truss and the change in boundary condition assumptions for each group's operational mode shapes. As such,

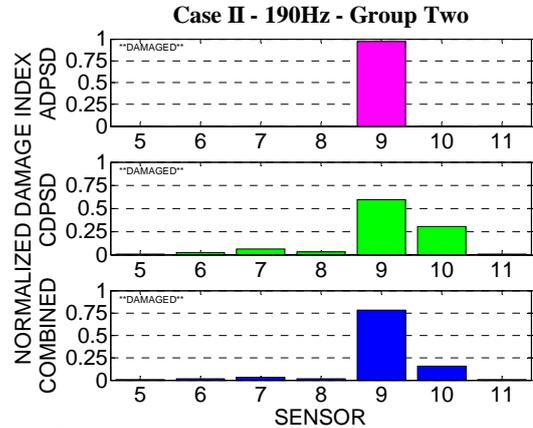


Figure 4. Case II – Group 2 – 190 Hz

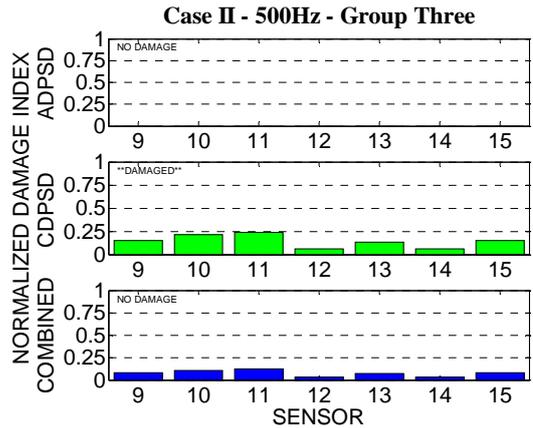


Figure 5. Case II – Group 3 – 500 Hz

these values are set to zero in the graphs for easier readability. The CDPSD still detects damage at sensors 9 and 10 but the false positives have increased to a significant level to cloud the actual damage location. The algorithm indicates no damage in Figure 5 even though damage is technically present because of this problem with the end nodes. To overcome this deficiency, the cluster assignments need to ensure they maintain an overlap of three, as a minimum. Doing so will ensure that damage will still be properly indicated in the structure as a whole.

#### 4.3 Case III

Case III illustrates what happens when the group sizes are too small for proper normalization of the PSD and subsequent spline fitting to determine the derivatives. In the ideal case, where damage is located between the two center nodes of the four node group, the proposed algorithm properly indicates the damage. However, when damage is located at the edges, either the method indicates no damage as shown in Figure 5 of Case II or it indicates damage in the wrong bay as illustrated in Figure 6. As stated above, this can be rectified by having a minimum overlap. However, the small group size also produced a large number of false positives in bays in which no damage was present in any form. Figure 7 shows an example of this worrisome situation. This group is far from either damage location, yet it falsely indicates damage between sensors 13 and 14.

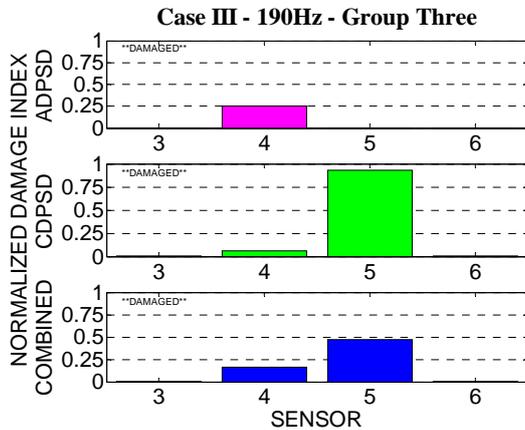


Figure 6. Case III – Group 3 – 190 Hz

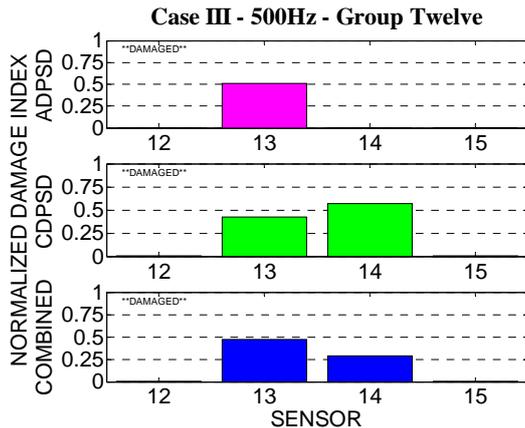


Figure 7. Case III – Group 12 – 500 Hz

The false positive results from the minimal differences in the small group being amplified through the normalization processes. Increasing the group size ensures that the normalization does not amplify the small changes due to different excitation records when damage is not present in the group. Having a minimum group size of five eliminates the presence of the type of false positive shown in Figure 7. The proposed algorithm does not report any false positives, even when no damage is present anywhere, as long as the group size is above the minimum of five.

## 5 CONCLUSIONS

The methods introduced by Beskhyroun et al. (2004, 2005) have been extended to improve their performance and to facilitate implementation in the distributed computing environment intrinsic to a smart sensor network. Through numerical examples, the modified approach has been shown effective in damage detection and location when limitations of node overlap, cluster size, cutoff frequency are properly observed. The minimum overlap of three sensors and minimum group size of five are exactly that – minimums. In full-scale civil infrastructure, the envisioned smart sensor network would consist of thousands of sensors deployed along its length and height. A network of only five sensors would not provide sufficient information to make an educated assessment of the structure’s functionality. How-

ever, any software that controls the clustering of the nodes must observe these minimums in order to maintain the accuracy and efficiency of the entire network. Being aware of the limitations of the damage detection method used in the hierarchical environment greatly improves the fault tolerance of the smart sensor network.

While consisting of thousands of nodes, smart sensor networks will also be multi-scale. This implies that the networks will use various damage detection algorithms using different input data to evaluate the structure’s status. The proposed PSD method, though lacking information on the severity of damage, serves as a computationally and energy efficient method to detect and locate the damage. Implementing other methods using additional data, increased data sharing, or more intense calculations could then supplement the initial damage report. Further analytical and physical experiments are underway to expand the results presented heretofore.

## 6 ACKNOWLEDGEMENTS

The authors gratefully acknowledge National Science Foundation’s support under grants CMS 03-01140 and CMS 06-00433, Dr. S.C. Liu, Program Manager. The first author also acknowledges support by NSF’s Graduate Research Fellowship Program.

## REFERENCES

- Beskhyroun S, Oshima T, Mikami S. 2004. Structural Health Monitoring of Bridges Based on Vibration Measurements. *International Workshop on Modern Science and Technology*, Kitami, Japan, September 2-3, 2004
- Beskhyroun S, Oshima T, Mikami S, Yamazaki T. 2005. A Numerical Analysis of Structural Damage Detection Using Changes in the Curvature of Power Spectral De *Journal of Structural Engineering*. Japan. 51A
- Beskhyroun S, Oshima T, Mikami S, Tsubota Y. 2006. Structural Damage Identification Algorithm Based on Changes in Power Spectral Density. *Journal of Applied Mechanics*. Japan. 8
- Doebbling SW, Farrar CR, Prime MB, Shevitz DW. 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*.
- Gao Y, Spencer BF Jr, Ruiz-Sandoval M. 2006. Distributed Computing Strategy for Structural Health Monitoring. *Structural Control and Health Monitoring*. 13:488-507
- Nagayama T, Rice JA, Spencer BF, Jr. 2006. Efficacy of Intel’s Imote2 Wireless Sensor Platform for Structural Health Monitoring Applications. *Proceedings of the Asia-Pacific Workshop on Structural Health Monitoring*.
- Sohn H, Farrar CR, Hermez FM, Shunk DD, Stinematates DW, Nadler BR. 2003. A Review of Structural Health Monitoring Literature: 1996-2001. *Los Alamos National Laboratory Report LA-13976-MS*.