Enabling Generalized Coverage-Preserving Scheduling in Wireless Sensor Networks

Peng Guo, Xuefeng Liu, Shaojie Tang, Jiannong Cao, *Fellow, IEEE*

Abstract—Wireless sensor networks (WSNs) are generally utilized to monitor, in an area, certain events or targets that users are interested. To extend system lifetime, a widely used technique is 'Energy-Efficient Coverage-Preserving Scheduling (EECPS)', in which sensor nodes are divided into multiple cover sets and activated in turn to complete the designated task. Although many EECPS schemes have been proposed, an intrinsic assumption in existing EECPS schemes is that each node has a fixed coverage area. This assumption, however, does not hold for some 'domain-specific' applications of WSNs such as structural health monitoring (SHM) and volcano tomography. In these applications, to complete the designated task always requires low level collaboration of multiple sensors. The coverage area for an individual sensor node cannot be defined explicitly since a single sensor is not able to fulfill the function alone, even it is close to the event or target to be monitored. In this paper, using an example of SHM, we extend the EECPS to broader applications of WSNs. We re-define the 'coverage' and propose two methods to partition deployed sensor nodes into qualified cover sets such that system lifetime can be maximized. The performance of the methods is demonstrated through both extensive simulations and real experiments.

Index Terms—Wireless sensor networks, coverage preserving scheduling, structural health monitoring.

1 Introduction

Wireless sensor networks (WSNs) are widely used to detect some events or targets in a given area [1]. To extend system lifetime, a widely used technique is to activate only part of the deployed sensor nodes each time and put the others into sleep. Since active sensors should still be able to cover the whole monitoring area, this technique is generally called as 'energy-efficient coverage-preserving scheduling (EECPS)'.

The problem of EECPS in WSNs has been studied extensively in the last decade. Many EECPS schemes have been proposed and they are generally based on some concepts like 'coverage' and 'cover set', etc. Based on the coverage areas of deployed sensor nodes, the EECPS schemes can either divide the nodes into mutually exclusive or overlapping cover sets as in [2] [3] [4], or determine in a distributed way a node's activity according to whether its coverage area has already been covered by its active neighbors [5] [6].

From the above discussion, we can see one basic assumption of using EECPS is that *each sensor node has a specific coverage area*. The coverage area of a sensor node is intrinsically its sensing area for which the node can reliably detect any event/target occurring in it. This assumption is valid for most traditional applications of WSNs such as forest fire monitoring and intruder detection, etc. However, we found that it does not hold for some newly emerged domain-specific applications of WSNs. Typical examples are structural health monitoring (SHM) and volcano tomography.

Case 1: WSN-based SHM systems

Let us first take SHM as an example. Fig. 1 illustrates how the status of a bridge can be monitored by a typical WSN-based SHM system. An array of wireless sensors are deployed on different locations on the bridge. Every certain period of time, these sensors collect the structure's responses for a while and transmit the data to a sink node where a SHM algorithm is implemented. The SHM algorithm usually performs the following steps: Firstly, data collected from part or all of the deployed nodes are put into a matrix; then from this data matrix, some vibration features of the structure are extracted, generally through various of matrix computations such as singular value decomposition [7]. The damage information is then identified by examining the changes in these vibration features [8].

Accurately identifying vibration features, however, does not require data from all the deployed sensor nodes. Data from some properly selected sensor nodes can also obtain accurate feature vectors. Therefore, in each round of monitoring period, we can select part of the deployed sensor nodes that are able to estimate features accurately to monitor structural condition. This matches the objective of the EECPS very well.

However, we can see from Fig. 1 that, SHM utilizes a different way to detect events (i.e. structural damage) from traditional applications of WSNs. To detect events in SHM, features are identified from matrices containing data from multiple sensor nodes. An individual node, no matter how close it is to damage location, is not able to

- *Peng Guo is with Huazhong University of Science and Technology, China. E-mail: guopeng@hust.edu.cn*
- *Shaojie Tang is with Jindal School of Management, University of Texas at Dallas, U.S.A. E-mail: shaojie.tang@utdallas.edu.*
- *Xuefeng Liu and Jiannong Cao are with Department of Computing, The Hong Kong Polytechnic University. E-mail: csxfliu,csjcao@comp.polyu.edu.hk*

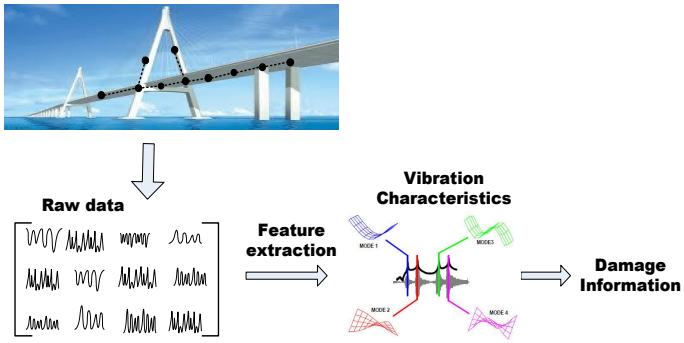


Fig. 1: The procedures of event detection in SHM. The procedures of SHM involves complicated computation on a data matrix including measurement from multiple sensor nodes.

detect the damage by itself. *This indicates that we cannot define a specific coverage area for each sensor node.* Someone may argue that as an analogy for coverage area, we can define a 'contribution factor' for each sensor node which reflects the contribution of the node in feature extraction. This is however not possible since the contribution of a certain node is different if it is involved in different sensor sets. Since we cannot define a specific coverage area or the equivalent contribution factor for individual sensor nodes, existing EECPS cannot be applied in SHM.

Case 2: Volcano Seismic Tomography using WSNs

The objective of volcano seismic tomography is to reveal deep structure beneath volcanic areas. Earthquakes generate seismic waves that propagating through the earth and accumulating the information on the inner structure of the Earth. Seismic tomography method, shown in Fig. 2, is to decipher this information and to provide the 3D distributions of velocities of seismic waves. The velocity information can give clue about the temperature, rheology and composition of deep rocks and fluid inside a volcano.

WSNs have been recently utilized for Volcano Seismic Tomography [9]. In a WSN-based system, a large-scale sensor network of low-cost geophysical stations sense seismic signals and implement seismic tomography in a collaborative manner. Establishing a 3D tomographic model can be formulated as a large, sparse matrix inversion: $\mathbf{A} * \mathbf{s} = \mathbf{t}$, where $\mathbf{s}$ is the velocity model and is the parameter vectors to be estimated; $\mathbf{A}$ and $\mathbf{t}$ are the matrices containing direct and undirect measurements from deployed sensor nodes.

We can see that in volcano seismic tomography, deployed sensor nodes must collaborate with each other to identify the 3D tomographic model. Individual nodes do not have the capability to obtain even part of parameters $\mathbf{s}$. In addition, in the least square estimation, the contribution of individual nodes cannot be evaluated separately. This again indicates that *there is no specific coverage area, or similar 'contribution factor' that can be defined for each sensor node*. Without this information, existing EECPS methods cannot be applied.

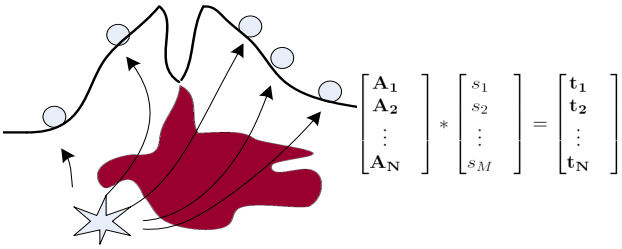


Fig. 2: Volcano seismic tomography. The problem of seismic tomography can be formulated as a large, sparse matrix inversion in which the matrix containing measurement from multiple deployed sensor nodes.

To summarize, different from most traditional applications of WSNs where the sensing region of a sensor is modeled as a circle (in 2D space) [3], a sphere (in a 3D space), or a convex function [10], there do exist some WSN applications in which individual nodes do not have specific coverage area. It is therefore necessary to re-define the coverage such that EECPS technique can be applied in these applications.

In this paper, taking SHM as an example, we illustrate how the EECPS can be extended to these WSN applications. We first define a generalized coverage model. Different from conventional coverage model which is a geographic area defined for individual sensors, this generalized coverage model only needs a function to determine whether a set of sensor nodes is able to complete the required task. Based on this generalized coverage model, we then propose two methods to partition the deployed nodes into qualified cover sets that can work in turn to maximize system lifetime. The performance of the proposed methods is demonstrated through both extensive simulations and real experiments.

Our contribution in this paper is as follows:

- We found that in some domain-specific WSN applications, individual nodes do not have specific coverage area and as a result, the EECPS cannot be directly applied.
- Taking an example of SHM, we re-define the 'coverage' and based on this generalized coverage model, two methods are proposed to partition the sensor nodes into qualified 'cover sets' such that each cover set is able to complete the designated task.
- The proposed methods can be generalized to other application areas besides SHM such as volcano tomography in which coverage area of individual nodes cannot be defined explicitly.

2 Preliminaries

In this section, we first introduce how structural damage is identified in SHM. Two important vibrational features, namely, natural frequency and mode shape, are introduced. Based on these two features, a multi-tired damage identification strategy is described.Table 1

$f^k, \mathbf{\Psi^k}$	The k^{th} modal parameter of a structure
$G = (V, E)$	a WSN deployed on a structure
m	the number of nodes in V
N	the amount of data sampled at each sensor node in one round of damage detection
e_S, e_R, e_T	Energy consumed for sampling/rece./trans. one data
$cost(v_i, S_j)$	energy consumed by node v_i in one round of damage detection using sensor set S_j
er_i	the initial energy stored in node v_i
$cond(S_j)$	the condition number of sensor set S_j
r_j	number of rounds assigned to S_j
S_{v_i}	a sensor set includes v_i and all its one-hop neighbors in $G = (V, E)$
$\mathbb{S}_{v_i}$	a group of sensor sets satisfying constraints in Eq. 15
C_V	a collection of C_{v_i}s for all v_i in $G(V, E)$
M	the total number of sensor sets in C_V
$\bar{C}_V$	$\{S_{v_1}, S_{v_2}, \cdots, S_{v_m}\}$, the group of all the sensor sets S_{v_i} established using nodes in V
$\bar{C}_V^{str}$	a group of sensor sets by mapping a binary string `str` to $\bar{C}_V$

TABLE 1: Important mathematical notations

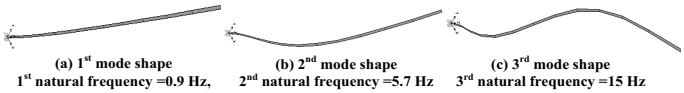


Fig. 3: The first 3 modal parameters of a cantilever beam.

summarizes important mathematical notations used in this paper.

2.1 Vibration features utilized in SHM

Every structure has tendency to oscillate with much larger amplitude at some frequencies than others. These frequencies are called **natural frequencies**. When a structure is vibrating under one natural frequency, the corresponding vibration pattern it exhibits is called the **mode shape** of this natural frequency. For a structure with n-degrees of freedom (DOFs), its natural frequency set and mode shapes are denoted as:

$$\mathbf{f} = [f^1, f^2, ..f^n]' \tag{1}$$

$$\mathbf{\Phi} = [\mathbf{\Psi}^1, \mathbf{\Psi}^2, \cdots, \mathbf{\Psi}^n] = \begin{bmatrix} \phi_1^1 & \phi_1^2 & \cdots & \phi_1^n \\ \phi_2^1 & \phi_2^2 & \cdots & \phi_2^n \\ \vdots & \vdots & \ddots & \vdots \\ \phi_n^1 & \phi_n^2 & \cdots & \phi_n^n \end{bmatrix} \tag{2}$$

where f^k $(k = 1..., n)$ is the k^{th} natural frequency, $\mathbf{\Psi}^k (k = 1, ..., n)$ is the mode shape corresponding to f^k. $\phi_i^k (i = 1, 2, ..., n)$ is the value of $\mathbf{\Psi}^k$ at the i^{th} DOF. For convenience, f^k and $\mathbf{\Psi}^k$ are also called **modal parameters** corresponding to the k^{th} **mode** of a structure. As an example, Fig. 3 shows the first few natural frequencies and mode shapes of a typical cantilevered beam.

Modal parameters are internal properties of a structure and by examining the changes in the modal parameters, damage on a structure can be identified [11]. However, natural frequencies $\mathbf{f}$ and mode shapes $\mathbf{\Phi}$ play different roles in damage identification. It can be seen from Eq. 1 that $\mathbf{f}$ do not contain any spatial information. This means that by examining the changes of natural frequencies, only the information about the existence of damage can be obtained. On the other hand, mode shape $\mathbf{\Phi}$ in Eq. 2 has an element corresponding to each DOF and thus contains spatial information. Mode shapes and their derivatives have been proven to be very effective features to locate structural damage [11].

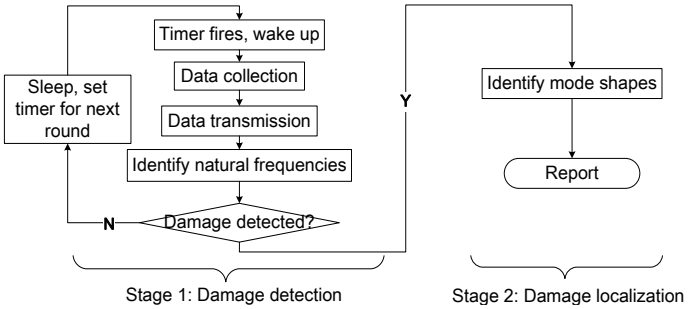


Fig. 4: The multi-tired strategy in SHM

2.2 A multi-tired strategy in SHM

In this paper, a multi-tired strategy shown in Fig. 4 is adopted to detect and localize structural damage. In this approach, every period of time, deployed nodes collect a number of data and transmit them to the sink where the natural frequencies $\mathbf{f}$ are identified to detect the existence of damage. Once damage is detected, mode shapes $\mathbf{\Phi}$ are identified to give damage locations.

We can see that in this multi-tired strategy, the first stage (i.e. natural frequency-based damage detection) is of the most importance since most of the time, the system will be iteratively running at this stage. Therefore, how to identify natural frequencies in an energy-efficient manner becomes critical for a WSN.

In this paper, we utilize the classic eigen-system realization algorithm (ERA) to identify both natural frequencies and mode shapes [12]. The details of the ERA, along with the associated energy cost, will be described in the next section. Interestingly, we will show that the *identification of natural frequencies does not requires data from all the sensor nodes*. Some properly selected sensor nodes are able to give accurate estimation of natural frequencies. Therefore, we can divide the deployed sensor nodes into a number of sets with each set can provide accurate natural frequencies. Once structural damage is identified, all the deployed sensor nodes are initiated to identify mode shapes to find out the damage location.

2.3 The quality model and energy consumption model using the ERA

In this section, we briefly introduce how natural frequencies are identified in a WSN and the associated cost. We then give the quality model which is used to evaluate whether a given set of sensors is able to generate accurate estimation of natural frequencies.

In every monitoring period, each sensor node, including the sink node, will collect N samples. Then the shortest path tree will be established in the network and along which each node will transmit its data to the sink node. Having collected the data from all the sensor nodes, the sink node implements the ERA as follows. First, data from all the sensor nodes are formed into two Hankel matrices. Then a series of matrix computations including singular value decomposition (SVD) and eigen decomposition will be applied on the matrices to produce the estimate of natural frequencies [12]. Note that to detect structural damage, only the first few frequencies needs to be identified. The number of frequencies to be identified, denoted as p_{mod}, is determined by civil analysts. Using a large p_{mod} is more likely to reveal minor structural damage occurred on a structure.

Energy consumption model

We estimate the energy consumption of a WSN in one monitoring period. Assume the WSN is denoted as $G = (V, E)$ where V being the sensor nodes and E being the wireless links among V. For a sensor node $v_i \in V$, its total energy consumption in one round of damage detection, denoted as $cost(v_i, G)$, can be mainly decomposed into the following three parts:

$$cost(v_i, G) = Er_s(v_i) + Er_c(v_i, G) + Er_a(v_i, G) \quad (3)$$

where $Er_s(v_i)$, $Er_c(v_i, G)$ and $Er_a(v_i, G)$ are the energy consumed in data sampling, wireless communication and computation, respectively. Note that the last two terms are dependent on whole network G.

For each sensor node in V, the sampling cost $Er_s(v_i)$ is the same:

$$Er_s(v_i) = N \cdot e_S \quad (4)$$

where e_S is the energy for sampling one data.

However, $Er_c(v_i, G)$ and $Er_a(v_i, G)$ will be different for different nodes in V. Assume in the network, the energy consumption for transmitting and receiving one data unit are denoted as e_R and e_T, respectively. If v_i is the sink node, which needs to collect the data from all the other nodes, the wireless communication cost $Er_c(v_i, G)$ will be:

$$Er_c(v_i, G) = (|V| - 1)N \cdot e_R \quad (5)$$

where $|V|$ is the number of nodes in V.

While for a non-sink node v_i, we let the shortest path from a node in V to the sink node be $p_a = h_0 h_1, \cdots, h_k$. Define $p_a[i] = h_i$ as the i^{th} hop sensor on path p_a. $Er_c(v_i, G)$ can then be computed as:

$$Er_c(v_i, G) = \sum_{\forall p_a, \exists j>0, p_a[j]=v_i} N \cdot e_R + \sum_{\forall p_a, \exists j, p_a[j]=v_i} N \cdot e_T \quad (6)$$

In terms of computation cost $Er_a(v_i, G)$, for a sink node, its $Er_a(v_i, G)$ is formulated as:

$$Er_a(v_i, G) = e_{ERA}(|V|) \quad (7)$$

where e_{ERA} is the energy used in the sink when it carries out the ERA for natural frequency identification. e_{ERA} is dependent on $|V|$ and it is generally not a linear function of $|V|$ since the ERA contains many computations such as the SVD whose complexity dramatically increases with $|V|$.

To summarize, the energy consumption in one monitoring period for a sink node and a common node are denoted as

$$cost(v_i, G) = N \cdot e_S + (|V| - 1)N \cdot e_R + e_{ERA}(|V|) \quad (8)$$

and

$$cost(v_i, G) = N \cdot e_S + \sum_{\substack{\forall p_a, \exists j>0, \\ p_a[j]=v_i}} N \cdot e_R + \sum_{\substack{\forall p_a \exists j, \\ p_a[j]=v_i}} N \cdot e_T \quad (9)$$

respectively.

Quality model

Traditionally, to identify natural frequencies, data from all the deployed sensor nodes are used in the ERA [12]. However, in the ERA, data from some properly selected sensor nodes can also obtain accurate natural frequencies. Briefly speaking, the accuracy of identified natural frequencies from a sensor set S is dependent by measurement noise, the finite element model (FEM) of the structure, and the number as well as the locations of nodes in S. Furthermore, according to [13], under the assumptions that (1) all the sensor nodes in S have the same measurement noise, and (2) $|S| \geq p_{mod}$ where $|S|$ is the number of sensor nodes in S, the accuracy of identified natural frequencies is determined by the condition number of S:

$$cond(S) = \|\mathbf{\Phi}_S\| \cdot \left\|\mathbf{\Phi}_S^{-1}\right\| \quad (10)$$

where $\|.\|$ is the Euclidean norm, and $\mathbf{\Phi}_S$ is the structure's mode shape matrix $\mathbf{\Phi}$ in Eq. 2 retaining only the rows corresponding to the DOFs of S. *The larger the $cond(S)$, the less accurate of identified natural frequencies from S.* $\mathbf{\Phi}_S$ can be obtained by the FEM of the structure. In this paper, we assume that the measurement noise is the same for all the sensor nodes and p_{mod} is a value pre-determined by civil analysts. Under this assumption, a sensor set $S \subseteq V$ is able to produce accurate natural frequencies if the following two conditions are satisfied:

$$\begin{aligned} &\text{Condition 1:} \quad |S| \geq p_{mod} \\ &\text{Condition 2:} \quad cond(S) \leq \gamma \end{aligned} \quad (11)$$

where γ is the threshold for $cond(S)$. Some examples of how to choose γ will be given in Section 5.

3 Definition and Problem Formulation

Following the same idea of the EECPS, if we can find a number of sensor sets in V which satisfy the requirements in Eq. 11 and let them work successively, system lifetime can be extended. In addition, with the help of the energy consumption model shown in Eq. 8

and Eq. 9, it is possible to find out the optimal sets to maximize the overall system lifetime.

The discussions above imply that although we cannot define a 'coverage area' for an individual sensor node in SHM, Eq.11 does provide information whether a set of sensor nodes is able to provide accurate vibration features to monitor the structural condition. *This, in some respects, is analogous to the concept of 'coverage'.* In the following sections, we first re-define the coverage in SHM and then give problem formulation in which the deployed sensor nodes are divided into optimal 'cover sets' such that system lifetime can be maximized.

3.1 Re-define the Coverage in SHM: SHM coverage

In this section, we give two definitions of coverage in SHM.

Definition 1 (α-SHM coverage)**.** *A sensor set S is called to be able to α-SHM cover a structure iff using S, the vibration features can be identified with no less than a pre-defined accuracy α.*

The argument for this definition is as follows. Detecting events in SHM relies on identified vibration features (which is natural frequencies $\mathbf{f}$ in this paper). Given a certain level of damage to be detected and measurement noise, we can always find a threshold α such that as long as the accuracy of identified vibration features is higher than α, the pre-defined damage can be detected. An example of finding α given a certain level of damage to be detected will be given in Section 5.2. Obviously, detecting small damage under high measurement noise usually needs a large α.

α-SHM coverage can be transformed to a more convenient form. Considering the accuracy of identified vibration features is directly associated with $cond(S)$ shown in Eq.10, we utilize γ, the threshold for $cond(S)$, to define SHM coverage as follows:

Definition 2 (γ-SHM coverage)**.** *A sensor set S is able to γ-SHM cover a structure iff $|S| \geq p_{mod}$ and $cond(S) \leq \gamma$.*

In these two definitions, we use coverage models that are different from traditional one. Traditional coverage model is a geographic area defined for individual sensors, while these two coverage models do not define the coverage area of individual nodes, but rely on some functions to specify whether a set of sensor nodes can monitor the whole area with a certain capability (specified by α or γ). In the remaining of the paper, only γ-SHM coverage is utilized, and for simplicity, we simple use SHM coverage for γ-SHM coverage.

Obviously, γ is dependent on α. Generally speaking, given α, the corresponding γ can be obtained using the *trail-and-error* method as follows. Initially, we set γ to be a relatively large value and find out, under the given noise level, the estimation error of natural frequencies from the SHM-cover sets. Then we can keep decreasing γ until the estimation error meets the requirement of α.

For a SHM cover set $S \subseteq V$, the energy consumption of the sensor nodes in S when identifying natural frequencies can be easily obtained as in Eq. 8 and Eq. 9 by replacing G with G_S, which is the graph of S. Furthermore, in this paper, considering relaying large amount of sampled data through multiple hops can be energy consuming, we simply require that each SHM cover set S is a single-hop cluster. This means that there is a cluster head (CH) in G_S which can directly communicate with all its members. A CH needs to collect data from its members and implement the ERA to identify natural frequencies. Therefore, CHs in SHM cover sets now take the function of the sink node in the original WSN. Under this constraint, the cost of a CH and a cluster member in G_S is

$$cost(v_i, G_S) = N \cdot e_S + (|S| - 1)N \cdot e_R + e_{ERA}(|S|) \quad (12)$$

and

$$cost(v_i, G_S) = N \cdot e_S + N \cdot e_T, \quad (13)$$

, respectively.

3.2 Problem formulation

Having defined SHM coverage, the objective becomes *to divide the deployed sensor nodes into multiple SHM cover sets and when they are activated in turn, the total number of rounds they work is maximized.* We give the formal problem definition below:

Maximum SHM Cover Set Problem (MSCSP):
Given:

- a WSN $G = (V, E)$.
- p_{mod}, γ, and $\mathbf{\Phi}_S$ which determines whether a set $S \subseteq V$ is a SHM cover set.
- $cost(v_i, G_S)$, which is the energy consumption of each node v_i when a sensor set $S \subseteq V$ is chosen to implement one round of damage detection. If $v_i \in S$, $cost(v_i, G_S)$ is taking in the form of Eq. 12 or Eq. 13 according to the role it plays in S. If $v_i \notin S$, $cost(v_i, G_S) = 0$.
- er_i, the remaining energy of each sensor node v_i.

The problem is to find a family of sensor sets $S_1, \cdots, S_p$ ($S_j \subseteq V, j = 1, \cdots, p$) with the corresponding number of rounds $r_1, \cdots, r_p$ allocated for these sets such that

- $r_1 + r_2 + \cdots + r_p$ is maximized, while

subject to the following constraints:

- $\forall j = 1, \cdots, p$, $|S_j| \geq p_{mod}$ and $cond(S_j) \leq \gamma$.
- $\forall j = 1, \cdots, p$, $\exists v \in S_j$, such that $\forall u \in S_j (u \neq v)$ there is an edge between v and u.
- $\forall v_i \in V$, $\sum_{j=1}^{p} cost(v_i, G_{S_j}) r_j \leq er_i$.

Remarks:

- Inspired by [4], instead of dividing V into disjoint sets, we allow every node to be part of more than one set, and allow the sets to work for different number of rounds.
- The first constraint guarantees each set is a SHM cover set. The second constraint is to ensure only

single-hop SHM cover sets are generated. The last constraint guarantees that the total energy consumed for each node v_i across all SHM cover sets is no larger than er_i.

We prove that the MSCSP is NP-hard by proving that the decision version of the problem is NP complete which is defined as: ***given a threshold*** k ***, does there exist a collection of sensor sets*** $C = \{S_1, S_2, \cdots, S_p\}$ ***and the corresponding*** $r_1, \cdots r_p$***, which satisfy all the constraints above and*** $r_1 + r_2 + \cdots + r_p$ ***is equal or larger than*** k***?***

Proof: It is easy to prove this problem is NP. We show it is NP-complete by reducing the set packing problem [14] to it. The set packing problem is defined as:

Given: A universe $U = \{\bar{s}_1, \bar{s}_2, \cdots, \bar{s}_m\}$, a collection of subsets: $\bar{C} = \{\bar{S}_1, \bar{S}_2, \cdots, \bar{S}_p\}$ with $\bar{S}_j \subseteq U$ for $j = 1, \cdots, p$, and a number k.

Find: if there exist k subsets in $\bar{C}$ which are pairwise disjoint (in other words, no two of them intersect).

To reduce the set packing problem to the MSCSP, we construct a network $G = (V, E)$, mode shape matrix $\mathbf{\Phi}$, the number of natural frequencies to be identified p_{mod}, cost function $cost(v_i, S)$, threshold γ, and remaining energy er_i from the inputs of set packing problem in the following way:

1. For each $\bar{S}_j \in \bar{C}$, establish a local network $\bar{G}_j = (\bar{V}_j, \bar{E}_j)$ with a hub-spoke architecture. $\bar{V}_j$ includes one CH, denoted as $\bar{v}_0^j$ and m cluster members $\{\bar{v}_1^j, \bar{v}_2^j, \cdots, \bar{v}_m^j\}$. Each $\bar{v}_i^j, i = 1, \cdots, m$ corresponds to an element $\bar{s}_i \in U$. The mode shape vector corresponding to the CH $\bar{v}_0^j$ is a 1-by-m row vector with only the first element to be '1': $[1, 0, \cdots, 0]$, and the mode shape vector for the cluster member $\bar{v}_i^j$ is $[0, \cdots, 1, 0, \cdots]$, with the i^{th} element to be '1'. This setting applies for all the $\bar{S}_j \in \bar{C}$. It can be seen that after this stage, a total of p equivalent local star networks are generated.

2. For each $\bar{s}_k \in \bar{S}_i \cap \bar{S}_j$ ($\bar{S}_i, \bar{S}_j \in \bar{C}$ and $\bar{S}_i \cap \bar{S}_j \neq \emptyset$), the corresponding sensor nodes $\bar{v}_k^j$ and $\bar{v}_k^i$ in the local networks $\bar{G}_i$ and $\bar{G}_j$ will be merged. After this stage, we obtained a WSN $G = (V, E)$ which includes m single-hop clusters. Each cluster has exactly $m+1$ sensor nodes and some clusters may overlap with each other.

3. By adjusting e_T and e_R in Eq. 12 and Eq. 13, we make that given a single-hop network $\bar{G}_j = (\bar{V}_j, \bar{E}_j)$ with $|\bar{V}_j| = m+1$, the $cost(v, \bar{G}_j)$ ($v \in \bar{V}_j$) is the same no matter whether v is a CH or not. Also, the remaining energy of each node in $\bar{G}_j$ is set to be $er = cost(v, \bar{G}_j)$. This setting imposes that every sensor node can only make one round of damage detection and hence the SHM cover sets established using this approach are disjoint. $p_{mod} = m$ and the threshold γ is set to be 1.

With this transformation, it can be easily proved that

1. Assume, without loss of generality, $\{\bar{S}_1, \bar{S}_2, \cdots \bar{S}_k\}$ is a solution to the set packing problem, then the local networks $\{\bar{G}_1, \bar{G}_2, \cdots \bar{G}_k\}$ are SHM cover sets which satisfy the constraints in the MSCSP, and the number of rounds of damage detection assigned are $r_1 = r_2 = \cdots = r_k = 1$. Therefore, the total number of rounds of damage detection is k.

2. Assume a $G = (V, E)$ is constructed from the set packing problem and we have SHM cover sets $\{\bar{G}_1, \bar{G}_2, \cdots\}$ with $\sum_{j=1} r_j = k$ to the MSCSP problem, then the subsets $\{\bar{S}_1, \bar{S}_2, ...\bar{S}_k\}$ is a solution to the set packing problem, where $\bar{S}_j$ ($j = 1, \cdots k$) is the subset from which the local network $\bar{G}_j$ is established. The detailed proof is omitted for brevity.

□

4 Proposed Methods

In this section, we describe two methods to solve the MSCSP, one is based on bounded multi-dimensional knapsack problem (BMKP) and the other is based on the genetic algorithm.

4.1 A BMKP based algorithm for the MSCSP

In this section, we propose a method based on the BMKP. This method is largely divided into two stages. First, a set of candidate single-hop SHM sets are enumerated. Then the problem is reduced to the bounded multi-dimensional knapsack problem.

Given $G = (V, E)$ and a node $v_i \in V$, we find out a sensor set S_{v_i} which includes v_i and all its one-hop neighbors. Furthermore, based on S_{v_i}, a group of sensor sets, denoted as $\mathbb{S}_{v_i}$, is established:

$$\mathbb{S}_{v_i} = \{S_{v_i}^1, S_{v_i}^2, ...\} \tag{14}$$

$S_{v_i}^k$ ($k = 1, ...$) are constructed from nodes in S_{v_i} and must satisfy the following constraints:

$$\begin{aligned} &1: \forall S_{v_i}^k \in \mathbb{S}_{v_i}, v_i \in S_{v_i}^k \text{ and } S_{v_i}^k \subseteq S_{v_i} \\ &2: \forall S_{v_i}^k \in \mathbb{S}_{v_i}, cond(S_{v_i}^k) \leq \gamma \text{ and } |S_{v_i}^k| \geq p_{mod} \\ &3: \forall S_{v_i}^{k_1}, S_{v_i}^{k_2} \in \mathbb{S}_{v_i} (k_1 \neq k_2), S_{v_i}^{k_1} \not\subseteq S_{v_i}^{k_2} \end{aligned} \tag{15}$$

Briefly speaking, the first constraint requires that for each $S_{v_i}^k \in \mathbb{S}_{v_i}$, it must contain v_i and the rest nodes in $S_{v_i}^k$ are chosen from v_i's one-hop neighbors. v_i will be functioning as the CH in $S_{v_i}^k$ if $S_{v_i}^k$ is selected to detect damage. The second constraint is to ensure that each $S_{v_i}^k$ is a SHM cover set. The third constraint is to remove some redundant candidates when solving the knapsack problem. For two sets $S_{v_i}^{k_1}$, $S_{v_i}^{k_2}$, both satisfying the first two constraints above and $S_{v_i}^{k_1} \subseteq S_{v_i}^{k_2}$, it can be easily proved that when a solution of the MSCSP contains $S_{v_i}^{k_2}$, a better or at least equally good solution can be obtained by replacing $S_{v_i}^{k_2}$ with $S_{v_i}^{k_1}$.

For each node $v_i \in V$, we wish to enumerate all possible sensor sets that satisfy the constraints above. Theoretically, the number of possible sets in $\mathbb{S}_{v_i}$ could grow exponentially with respect to $|S_{v_i}|$. However, different from many applications of WSNs where thousands or tens of thousands cheap wireless motes can be deployed, the number of sensor nodes deployed on a civil structure are generally less than one hundred considering the cost of attached sensors and the application requirement. $|S_{v_i}|$ is thus further limited. Therefore, we can assume that $\mathbb{S}_{v_i}$ for any sensor node

$v_i \in V$ can be enumerated in relatively short time. For a WSN $G = (V, E)$ where $|V| = m$, a collection of sensor sets which satisfy the constraints above is denoted as:

$$C_V = \{\mathbb{S}_{v_1}, \mathbb{S}_{v_2}, \cdots \mathbb{S}_{v_m}\} = \{S_1, S_2, \cdots, S_M\} \tag{16}$$

where S_i is the i^{th} set in C_V and M is the total number of sets in C_V. Note that when determining the sensor sets for each $\mathbb{S}_{v_i}$, the energy constraint in the MSCSP is not considered.

After all the possible SHM cover sets have been enumerated, the MSCSP can be written in the form of the BMKP as follows:

$$\begin{aligned} \text{max.} \ & \sum_{j=1}^{M} r_j, \ r_j = \{0, \cdots l_j\} \\ \text{s.t.} \ & \sum_{j=1}^{M} cost(v_i, G_{S_j}) r_j \leq er_i, i = 1, \cdots, m \end{aligned} \tag{17}$$

where $l_j = \min\limits_{v_i \in S_j} \lfloor \frac{er_i}{cost(v_i, G_{S_j})} \rfloor$. It can be seen that when the MSCSP is expressed in the form of BMKP, each sensor set $S_j \in C_V$ represents a certain type of item in the BMKP. All sensor sets in C_V have the same benefit. We wish to maximize the total profit of the selected sensor sets (i.e. $\sum_{j=1}^{M} r_j$) under the constraint that the energy of each sensor node across all the selected sensor sets will not exceed its remaining energy (i.e. $\sum_{j=1}^{M} cost(v_i, S_j) r_j \leq er_i$). For a SHM cover set S_j, the maximum number of time it can be selected is bounded by the node in S_j whose energy will be depleted first when implementing the ERA (i.e. node v_i which satisfies $\min\limits_{v_i \in S_j} \lfloor \frac{er_i}{cost(v_i, G_{S_j})} \rfloor$, where $\lfloor \cdot \rfloor$ is the floor function which maps a real number to the largest previous integer).

After this transformation, we are able to employ existing algorithms originally designed for the BMKP to tackle our problem. In particular, we first transform the BMKP into 01MKP based on [15], and then adopt the cross-entropy optimization [16] to solve the 01MKP.

4.2 The GA Method for the MSCSP

In this section, we propose a Genetic Algorithm (GA) to solve the MSCSP. We will show later that in some conditions, the GA method can achieve satisfactory results comparable with the BMKP-based method but using less computation time.

A basic GA includes: [17]:

Generate an initial population;
repeat
  Select parents from the population to produce offspring;
  Evaluate fitness of the children and replace some or all of the population by the offspring with good fitness;
until a satisfactory solution has been found.

The first step in designing a GA for the MSCSP is to devise a suitable representation scheme to represent solution of MSCSP. As before, we first find out S_{v_i} for each v_i which includes v_i and all its one-hop neighbors in $G = (V, E)$. Then we have a group of sensor sets:

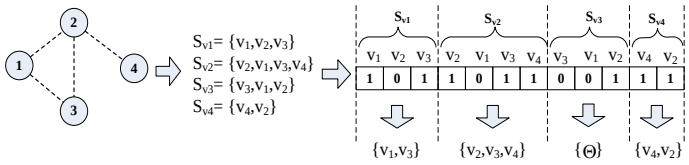


Fig. 5: GA example. Left: the topology of a WSN; Middle: The S_{v_i} of each node; Right: the encoding of $S_{v_1} \sim S_{v_4}$

$$\bar{C}_V = \{S_{v_1}, S_{v_2}, \cdots, S_{v_m}\} \tag{18}$$

It should be noted that $\bar{C}_V$ only includes m sensor sets and is different from C_V defined in Eq. 16. All the sensor sets in $\bar{C}_V$ are aligned together as an array and we use a $\sum_{i=1}^{m} n_i$-bit binary string `str` to encode it, where m is the number of nodes in V and n_i is the number of nodes in S_{v_i}. In this representation, a value of 1 or 0 at the j^{th} bit of `str` implies that whether the j^{th} node in $\bar{C}_V$ is selected or not. Therefore, from each S_{v_i}, at most one sensor set is generated by selecting the nodes in S_{v_i} whose values in `str` are encoded as $'1'$. Furthermore, we require that in S_{v_i}, if v_i is not selected, no sensor set will be generated from S_{v_i}, even if some other nodes in S_{v_i} are selected according to the `str`. After we apply a `str` on $\bar{C}_V$, we obtain a collection of sensor sets:

$$\bar{C}_V^{str} = \{S_1^{str}, S_2^{str}, \cdots, S_p^{str}\} \tag{19}$$

where $p \leq m$ since at most m sets can be obtained. The sensor sets in $\bar{C}_V^{str}$ will be a candidate solution for the MSCSP.

We use an example shown in Fig. 5 to demonstrate the procedures above. According to the network topology shown in the left figure of Fig. 5, four S_{v_i}s are obtained. When they are aligned as an array, we have:

$$\bar{C}_V = \underbrace{\{v_1, v_2, v_3\}}_{S_{v_1}} \underbrace{\{v_2, v_1, v_3, v_4\}}_{S_{v_2}} \underbrace{\{v_3, v_1, v_2\}}_{S_{v_3}} \underbrace{\{v_4, v_2\}}_{S_{v_4}} \tag{20}$$

Assume a binary string `str` on the $\bar{C}_V$ is $'101101100111'$. According to this `str`, three sets $\{v_1, v_3\}$, $\{v_2, v_3, v_4\}$, and $\{v_4, v_2\}$ are respectively generated from S_{v_1}, S_{v_2}, and S_{v_4} and are shown in the right figure of Fig. 5. No sensor set came from S_{v_3} since its CH v_3 is 0 in `str`. Therefore, we have:

$$\bar{C}_V^{str} = \{\{v_1, v_3\}, \{v_2, v_3, v_4\}, \{v_4, v_2\}\} \tag{21}$$

So far, the encoding method described above only indicates which sensor sets will be chosen to work but does not provide the number of rounds assigned to them. To solve this problem, we modify the $\bar{C}_V$ in Eq. 18 and allow the repeated S_{v_i}:

$$\bar{C}_V = \{ \underbrace{S_{v_1}, \cdots, S_{v_1}}_{\text{repeated } rep_1 \text{ times}}, \cdots, \underbrace{S_{v_m}, \cdots, S_{v_m}}_{rep_m \text{ times}} \} \tag{22}$$

where rep_i $(i = 1, \cdots, m)$ is the number of S_{v_i} repeated in $\bar{C}_V$. Theoretically, rep_i should be no less than the maximum number of rounds that any sensor sets generated from S_{v_i} (by applying str on $\bar{C}_V$) can work to implement the ERA:

$$rep_i \geq \max_{\forall \bar{S}_{v_i} \subseteq S_{v_i}} \min_{\forall v_k \in \bar{S}_{v_i}} \lfloor \frac{er_{v_k}}{cost(v_k, G_{\bar{S}_{v_i}})} \rfloor \tag{23}$$

where er_{v_k} is the remaining energy of v_k, $\bar{S}_{v_i}$ is a subset of S_{v_i} but still using v_i as the CH. Given $\bar{S}_{v_i}$, $\min_{\forall v_k \in \bar{S}_{v_i}} \lfloor \frac{er_{v_k}}{cost(v_k, G_{\bar{S}_{v_i}})} \rfloor$ is the number of rounds that $\bar{S}_{v_i}$ is able to work. Since v_i is always included in S_{v_i} and the $cost(v_i, G_{\bar{S}_{v_i}})$ is a non-decreasing function with $|\bar{S}_{v_i}|$, a safe choice of rep_i is that

$$rep_i = \lfloor \frac{er_{v_i}}{cost(v_i, \bar{S}_{v_i} = v_i)} \rfloor \tag{24}$$

Note that $\bar{S}_{v_i}$ in Eq. 24 only includes v_i, and obviously, $\lfloor \frac{er_{v_i}}{cost(v_i, v_i)} \rfloor \geq \max_{\forall \bar{S}_{v_i} \subseteq S_{v_i}} \min_{\forall v_k \in \bar{S}_{v_i}} \lfloor \frac{er_{v_k}}{cost(v_k, G_{\bar{S}_{v_i}})} \rfloor$.

When encoding $\bar{C}_V$ in Eq. 22, smaller rep_i is always more favorable from computational point of view. The rep_i shown in Eq. 24 can be further decreased by considering the fact that $\bar{S}_{v_i}$ should be a SHM cover set (we will show soon that by introducing an repair operator, any str that generates a non-SHM cover set will be fixed). Obviously, by considering the constraint, the $\bar{S}_{v_i}$ in Eq. 24 should include more nodes and the corresponding rep_i will be further decreased. Here, we use a greedy method to find out the $\bar{S}_{v_i}$ in Eq. 24. Initially, $\bar{S}_{v_i} = v_i$. Then the v_i's neighbors are added one by one, each time the one which is able to minimize $cond(\bar{S}_{v_i})$ is added. The procedure iterates until $\bar{S}_{v_i}$ is a SHM cover set. The corresponding rep_i is then used in Eq. 22.

We note that the encoding method described above might represent an infeasible solution to the MSCSP. A solution is infeasible when one or both of the following conditions occur:

$$\begin{aligned} &1\text{: } \exists S_k^{str} \in \bar{C}_V^{str}, |S_k^{str}| < p_{mod} \text{ or } cond(S_k^{str}) > \gamma \\ &2\text{: } \exists v_i \in V, \sum_{j=1}^{p} cost(v_i, S_j^{str}) > er_i \end{aligned} \tag{25}$$

There are a number of standard ways of dealing with infeasible solutions in GAs. For example, to apply a penalty function to penalize the fitness of any infeasible solution [18], to separate the evaluation of fitness and infeasibility [19], or to design a repair operator [20]. In this paper, the last approach is adopted and we designed a repair operator to convert an infeasible solution into a feasible MSCSP solution.

The repair operator proposed here is very simple. This operator includes two stages and in each stage, one condition shown in Eq. 25 is considered. Given

- $\bar{C}_V = \{S_{v_1}, \cdots, S_{v_1}, S_{v_2}, \cdots, S_{v_2}, S_{v_m}, \cdots, S_{v_m}\}$,
- a binary string str, and
- the corresponding group of sensor sets $\bar{C}_V^{str} = \{S_1^{str}, S_2^{str}, \cdots\ S_q^{str}\}$,

the repair operator first evaluates, one by one, all sensor sets in $\bar{C}_V^{str}$ to see whether they can SHM-cover the structure. For any sensor set S_i^{str} which fails, the operator finds out the sensor set in $\bar{C}_V$, denoted as S_{v_l}, from which the S_i^{str} is extracted. Then it randomly selects a $'0'$ from the segment of str corresponding to S_{v_l} and sets it to $'1'$. This procedure above corresponds to adding one neighbor of v_i which was not selected into S_i^{str}. S_i^{str} is then updated and evaluated. This procedure re-iterates until one of the two conditions is valid:

$$\begin{aligned} &1\text{: the updated } |S_i^{str}| \geq p_{mod} \text{ and } cond(S_i^{str}) \leq \gamma \\ &2\text{: } S_i^{str} = S_{v_l} \end{aligned} \tag{26}$$

The occurrence of the second condition in Eq. 26 indicates that even when all the one-hop neighbors of v_l have been used, the sensor set S_i^{str} still cannot SHM-cover the structure. If this is the case, the segment of str corresponding to S_{v_l} will be cleared and S_i^{str} will be deleted from $\bar{C}_V^{str}$.

After the repair operator goes through all the sensor sets in $\bar{C}_V^{str}$, the remaining sensor sets in $\bar{C}_V^{str}$ are able to meet the SHM-coverage requirements. The repair operator then seeks the most energy consuming node across the sensor sets in $\bar{C}_V^{str}$ and see whether its energy consumption exceeds the energy threshold. If it is true, the operator will delete, from $\bar{C}_V^{str}$, one sensor set in which the energy consumption of this node is highest in $\bar{C}_V^{str}$. This procedure re-iterates until the energy consumption constraint is satisfied for all the sensor nodes.

The detail procedure of this two-stage repair operator is illustrated in Algorithm 1. Algorithm 1 is guaranteed to produce a feasible solution for the MSCSP, irrespective of the initial binary string str.

We now need to define the fitness function for str. Given a str and the corresponding $\bar{C}_V^{str}$, the fitness function of the str is the number of sensor sets in $\bar{C}_V^{str}$. Note that since we allow repeated SHM cover sets in $\bar{C}_V^{str}$, the number of sets in $\bar{C}_V^{str}$ directly corresponds to the number of rounds of damage detection can implemented. The larger the number of sensor sets in $\bar{C}_V^{str}$, the higher the fitness of the corresponding str.

Having determined the repair operator and the fitness function, now we can start the evolution process. We first generate, at random, a population of strs (i.e. the genes), and then feed this population into the repair operator. From the output of the repair operator, each updated gene is evaluated by the fitness function. The the fitter genes will be used for mating and crossover to create the next generation of genes. A gene having the maximum fitness value among a population is called elite and carried through unchanged to the next generation. The iteration stops if there is no improvement in the maximum fitness value for ten consecutive generations.

Algorithm 1 The designed repair operator for MSCSP

Input: $\bar{C}_V$, str, and $\bar{C}_V^{str} = \{S_1^{str}, S_2^{str}, \cdots, S_q^{str}\}$
1: **for** i = 1 to q **do**
2: Find out the sensor set in $\bar{C}_V$ from which S_i^{str} is extracted. This set is denoted as S_{v_l}
3: **while** ($|S_i^{str}| < p_{mod}$ or $cond(S_i^{str}) > \gamma$) and $S_I^{str} \subset S_{v_l}$ **do**
4: Randomly select a $'0'$ from the segment of str corresponding to S_{v_l} and set it to $'1'$
5: Update S_i^{str}, str and $\bar{C}_V^{str}$
6: **end while**
7: **if** $|S_i^{str}| < p_{mod}$ or $cond(S_i^{str}) > \gamma$ **then**
8: Delete S_i^{str} from $\bar{C}_V^{str}$
9: Clear the segment of str corresponding to S_{v_l}
10: **end if**
11: **end for**
12: Find out the node with the highest energy consumption across all sets in $\bar{C}_V^{str}$. This node is denoted as v_{max}.
13: **while** $\sum_{S_i^{str} \in \bar{C}_V^{str}} cost(v_{max}, G_{S_i^{str}}) > er_{v_{max}}$ **do**
14: Delete the sensor set in which the energy consumption of v_{max} is highest among all the sensor set in $\bar{C}_V^{str}$
15: Update $\bar{C}_V^{str}$ and str
16: Find out the v_{max} in the updated $\bar{C}_V^{str}$.
17: **end while**
Output: The updated str and $\bar{C}_V^{str}$

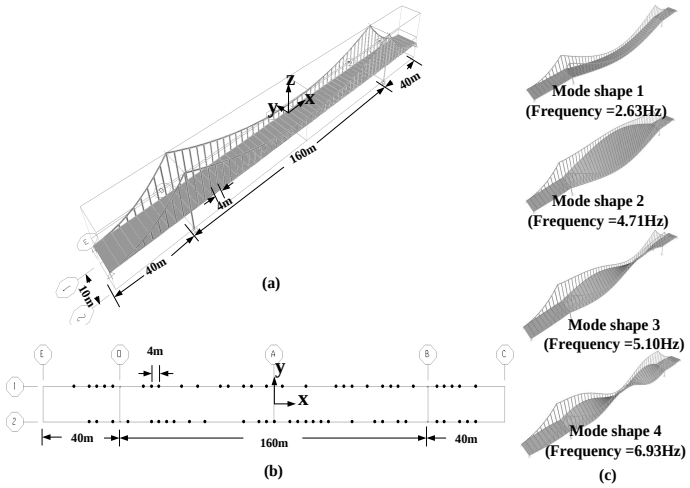


Fig. 6: The simulated bridge (a) The dimensions of the suspension bridge (3D), (b) the sensor locations (X-Y plane) (c) The first four theoretical modes of the bridge

5 Performance Evaluation

5.1 Results on a simulated bridge

To test the effectiveness of the proposed methods, a simulated suspension bridge shown in Fig. 6(a) is generated by SAP2000 [21].

A total of 60 wireless sensor nodes are generated to measure the vibration at the transverse direction (z direction in Fig.6 (a)) of the deck of the bridge. The locations of these nodes are shown in Fig. 6 (b). These 60 locations are selected using some domain knowledge from civil engineering [22]. We set the communication range of each node to be $C_r = 15m$.

The theoretical first 4 modes of the structure are illustrated in Fig. 6 (c). The mode shapes will be used to calculate the condition number of selected sensor sets. In addition, the natural frequencies shown in Fig. 6 (c) are utilized as the ground truth for validating the accuracy of estimated frequencies using selected SHM cover sets. The parameters used in the simulation are listed in Table 2. Some of the parameters, such as e_S, e_R, e_T and e_{ERA}, are obtained by some real tests on our own designed SenetSHM motes that are described in detail in [23]. In this simulation, it is assumed that all the sensor nodes use the same battery with $er = 700$ mAh (the voltage of the battery is 3.3v). To make it comparable, we also utilize *mAh* as an alternative unit for *J* (joule) when we calculate the energy consumption of sensor nodes. We require that the first 4 natural frequencies need to be identified (i.e. $p_{mod} = 4$) and the condition number of each SHM cover set should be less than $\gamma = 1000$ so that the identified natural frequencies can be accurate enough to detect a certain level of damage on this bridge. Choosing $\gamma = 1000$ is determined via the trail-and-error method described in Section 3.1 and its validity will be shown at the end of this section.

TABLE 2: Parameters used in the simulation.

N	er (mAh)	e_S (mAh)	e_R (mAh)	e_T (mAh)	e_{ERA} (mAh)
20480	700	1.1e-4	5e-4	5e-4	$0.0417(0.4\|V\|^2 +1.2\|V\| - 3.6)$

Fig. 7 illustrates the results of using the BMKP-based method proposed in Section 4.1. From Fig. 7 (a), it can be seen that 19 sensor sets are generated, and the number of working rounds assigned to each sensor set ranges from 1 to 20. The total system lifetime (i.e. the number of rounds of damage detection) becomes 190. Also, although not shown in the figure, the number of sensor nodes in each sensor set ranges from 4 to 5. From Fig. 7 (b) and (c), we can see that all the sensor sets are SHM cover sets ($cond(S) \leq 1000$), and the energy consumed of each node is below 700 mAh. We can also see from Fig. 7 (c) that some nodes are not selected by any SHM cover sets and hence seem to be 'redundant'. Note that these nodes are only 'redundant' when estimating natural frequencies but will still be useful when estimating mode shapes after damage is detected.

The redundant nodes are partially caused by the sparsity of the network. Intuitively, in a dense network where each node has a large number of neighbors, the number of redundant nodes should be smaller since each node has a higher probability to be involved in one or more SHM cover sets. Correspondingly, the total number of rounds that the system can work should be larger. To validate this, we fix the parameters shown in Table 2 but change the network density by adjusting the communication range of each node C_r. As an illustration, Fig. 8 shows the results when C_r = 19m. In this condition, 246 SHM cover sets are generated and the system lifetime is 393. Also can be seen from Fig. 8 (c) is that

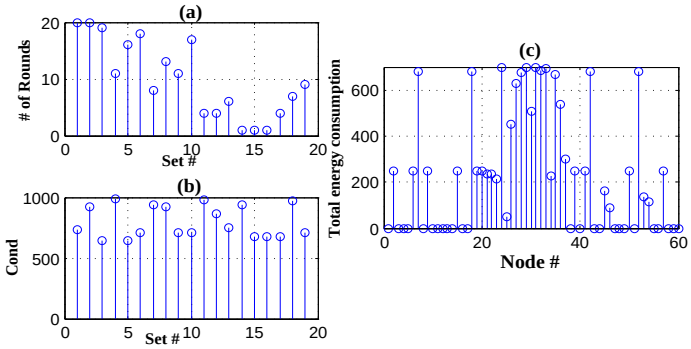


Fig. 7: Results from the BMKP-based method when C_r = 15m (a) The SHM cover sets and the number of rounds assigned (b) The condition number of each SHM cover set (c) The energy consumed of each sensor node across all SHM cover sets.

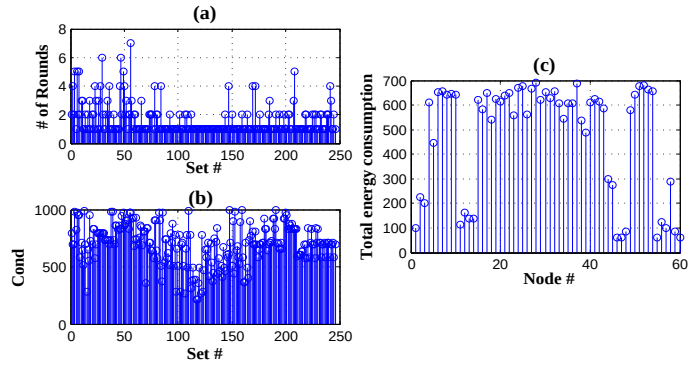


Fig. 8: Results from the BMKP-based method when C_r = 19m

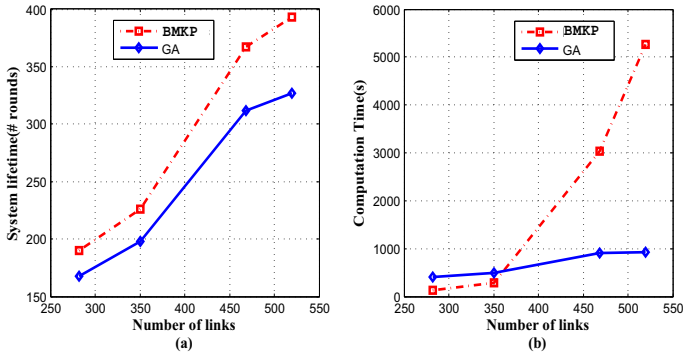


Fig. 9: Compare the results from the BMKP-based method and the GA method in terms of (a) lifetime, and (b) the computation time

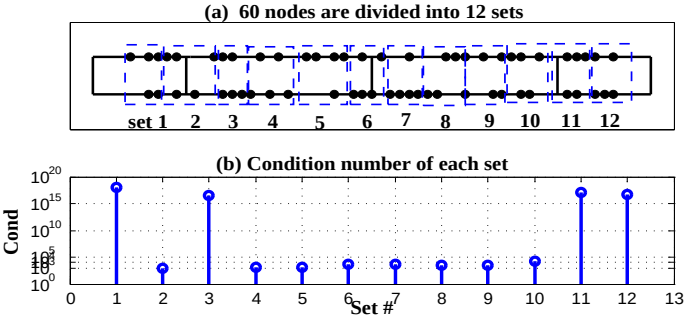


Fig. 10: (a) The deployed 60 sensors are divided into 12 disjoint sensor sets (b) the condition number of each sensor set.

there are no un-selected sensor nodes. A dense WSN obviously can increase the utility of individual nodes when they form into sensor sets to detect structural damage.

We wish to find out system lifetime under different network densities. First, we notice that compared to communication range C_r , the total number of links in the network may be a better measure to present network density since it is more robust to various sensor deployments. Fig. 9 (a) illustrates system lifetime when the number of links in the network is increasing from $142 \rightarrow 175 \rightarrow 238 \rightarrow 260$ (the corresponding C_r are 12, 14, 16 and 19m, respectively).

For comparison, Fig. 9 (a) also shows the system lifetime of using the GA method. It can be seen that compared with the BMKP-based method, the system lifetime of the GA method is slightly lower. However, the advantage of the GA over the BMKP-based method lies in the computation time. For the BMKP-based method, the computation time increases significantly with the increase of network density. This is illustrated in Fig. 9 (b), where it shows the computation time on a laptop computer (Lenovo Thinkpad W500, with 2.80 GHz CPU and 4G RAM).

The reason can be attributed to the fact that in the worst case, the number of candidate sensor sets in $\mathbb{S}_{v_i}$ of Eq. 14 is with the level of $O(n!)$ where n is the number of nodes in S_{v_i}. Consequently, when using the cross-entropy optimization to solve the transformed Knapsack problem, the computational complexity increases dramatically with the increase of the network density. However for the GA method, the length of the binary string `str`, which highly affects the complexity of the GA algorithm, is only $O(m^2)$ where m is the number of the nodes in the network. Hence the advantage of the GA over the BMKP-based method in terms of computation time is more obvious in a dense WSN with large number of nodes. This is also demonstrated in Fig. 9 (b), where the computation time of the two methods are compared in different network density. Fig. 9 (b) shows that the computation time of the GA becomes lower than the BMKP-based method when the number of links exceeds 350, and this gap becomes more obvious with the increase of network density.

In the following of this section, we discuss the importance of controlling the condition number of each SHM cover set. For comparison, we divide the deployed 60 sensors quite arbitrarily into 12 disjoint sensor sets (see Fig. 10 (a)). Each set contains exact 5 nodes, which is the maximum number of nodes in the SHM cover sets identified in Fig. 7. The corresponding condition number of each set is shown in Fig. 10 (b). We can see that the condition number of an arbitrarily selected sensor set can easily go wildly large. As an example, only set #2 has a condition number below 1000 and the condition numbers #1,#3,#11 even exceed 10^{15}.

Then we demonstrate how the condition number will

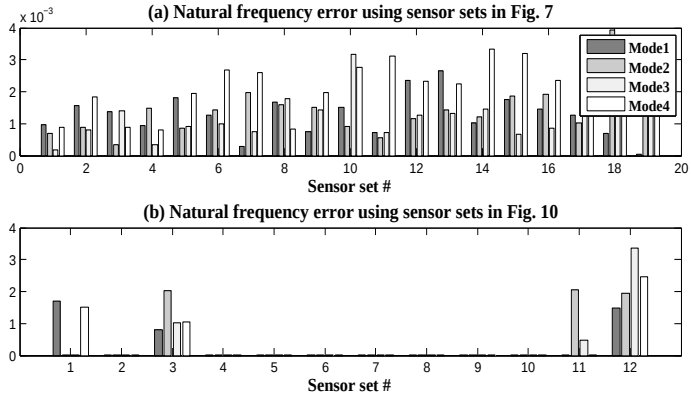


Fig. 11: The identified natural frequency error using the sensor sets in (a) Fig. 7 and in (b) Fig. 10

affect the accuracy of identified natural frequencies. We apply an impulse force along the z-direction in the middle span of the bridge and collect the impulse responses of the deployed 60 sensor nodes. The response time series are sampled at 200 Hz. Noise is added to the sensor data at each sample as a zero-mean Gaussian sequence with variance σ^2. In this simulation, σ^2 is chosen such that the ratio of the σ to the root-mean-square sensor output averaged over all the 60 sensors is 15%.

The error of the i^{th} identified natural frequencies, denoted as Δf^i, is calculated as

$$\Delta f^i = \left| \hat{f}^i - f^i \right| / f^i \tag{27}$$

where $\hat{f}^i$ is the identified i^{th} natural frequency and the f^i is the true one.

Fig. 11 (a) illustrates the identified natural frequency error using the SHM cover sets obtained from the BMKP-based method. The natural frequency error of all the SHM cover sets in all of the four modes are below 0.5%. It can be seen that if we set $\gamma = 1000$ for all the SHM cover sets, then even with the relatively high noise-to-signal ratio (15% in this case), the natural frequencies of the bridge are very accurately identified (with error $< 0.5\%$). In other words, damage which causes the deviation of natural frequencies larger than 0.5% is possibly to be detected by these SHM cover sets.

For comparison, the natural frequency error using the sensor sets in Fig. 10 are shown in Fig. 11 (b). It can be seen that the natural frequencies identified from sensor sets without controlling condition number are far from accurate. The maximum identification error can even reach over 300%. With such a large error, it is not possible to use the estimated natural frequencies for damage detection.

5.2 Experiment on a lab structure

The effectiveness of the proposed approaches is tested on a lab structure. To address the generally high requirements of SHM application, we designed a particular type of wireless sensor nodes called SenetSHM mote [23]. A SenetSHM mote mainly includes an Imote2 and a specially designed sensor board (see Fig.12 (a)). Imote2 is chosen as the central unit because it has a good balance between low power consumption and rich resources. The sensor board contains: (1) an on-board 3-Axis accelerometer and an interface for connecting external sensors, (2) a 2G-Byte μSD that is utilized for storing measured data, and (3) a radio triggered-wakeup unit that can wake up the mote when it receives wakeup messages from others. In particular, the radio-triggered wakeup unit provides a convenient way to implement the EECPS based on the obtained SHM cover sets.

The structure under test is shown in Fig. 12 (b). It has 12 floors, and 12 SenetSHM motes are deployed on different floors to monitor the structure's horizontal accelerations under the excitation from a hammer. Since it is a relatively small lab structure, the deployed SenetSHM motes can form a complete network. However, to test the effectiveness of the proposed methods in a multi-hop wireless network, we let each mote only directly communicate with four nearest neighbors. As an example, the neighbors of the SenetSHM mote deployed on the 12^{th} floor are shown in Fig. 12 (c). We use a gateway node which is connected to a laptop computer to collect data wirelessly. The SenetSHM motes run modified TinyOS and are configured to sample the accelerometers at frequency of 1024 Hz for 20 seconds.

At the same time, vibration data are also recorded by a wired system for reference. Using data sampled from the wired-system, we can estimate the first four mode shapes and natural frequencies based on FEM updating techniques [24]. The mode shapes will be used to calculate the condition number, while the natural frequencies will be utilized as ground truth to test the accuracy of using SHM cover sets.

Damage on this lab structure is generated by releasing a support ring on the third floor (see Fig. 12 (b)). To determine the required accuracy for the identified natural frequencies to detect such damage, we compare the natural frequencies before and after damage using data from the wired system. Table 3 lists the first 4 natural frequencies before and after damage. It can be seen that this damage generates $2\% \sim 6\%$ change on these natural frequencies. Therefore, we require that identified natural frequencies error should be at least lower than 2%.

Then we implement the trail-and-error method mentioned in Section 3 to obtain γ that can achieve the accuracy above. We found that for this test structure, γ should be smaller than 200. Under this requirement, we utilize the proposed methods to partition the deployed 12 SenetSHM motes into different SHM cover sets. Fig. 13 (a) and (b) illustrate the SHM cover sets and the corresponding number of rounds assigned using the BMKP method and the GA method, respectively. The expected system lifetime calculated from these two

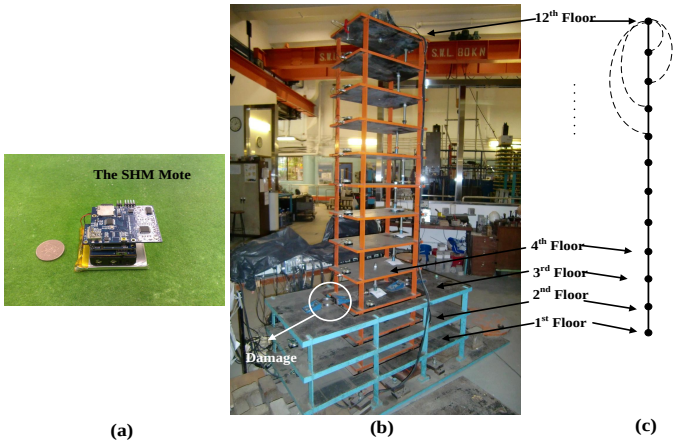


Fig. 12: The SHM mote and test structure (a) the SHM mote (b) test structure (c) the network topology of the mote on the 12^{th} floor.

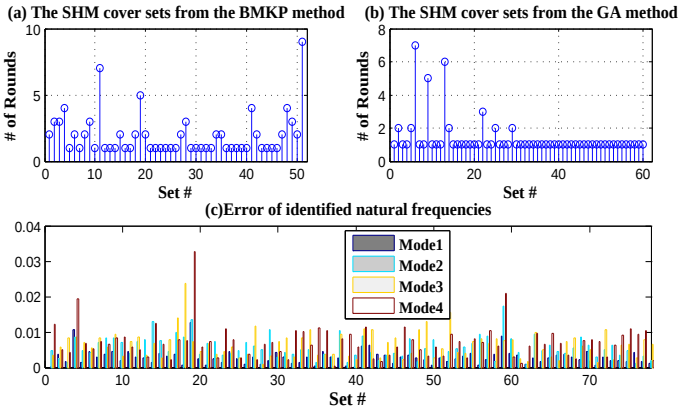


Fig. 13: Test results on the lab structure (a) The number of rounds assigned to each SHM cover set using the BMKP-based method, and (b) using the GA method. (c) The error of identified natural frequencies in each round of damage detection.

TABLE 3: The natural frequencies before/after damage and the differences.

	1^{st}	2^{nd}	3^{rd}	4^{th}
$\mathbf{f}$ (Healthy)	64.7	161.2	224.5	312.1
$\mathbf{f}$ (Damaged)	63.2	155.3	218.1	292.5
Δf	2.32%	3.66%	2.81%	6.28%

methods are 100 and 82, respectively.

As an example, we utilize the results from the BMKP-based method to implement the EECPS on this small WSN. Initially, all the deployed SenetSHM motes are put into sleep mode. The gateway first wakes up all the deployed motes wirelessly and then broadcasts a message containing the IDs of the motes in the first SHM cover set. Motes that find their IDs in the message will notify the gateway accordingly. If the gateway receives responses from all the motes in the cover set, it sends a beacon to trigger a round of damage detection. Otherwise, the gateway will directly jump to the next SHM cover sets. After a set of natural frequencies $\mathbf{f}$ is identified from a SHM cover set, all nodes in the set will go back to sleep and the gateway starts to initiate the next round of damage detection. Note that to test the performance of the EECPS, we did not implement damage localization as in Fig. 4, even it can be carried out straightforwardly.

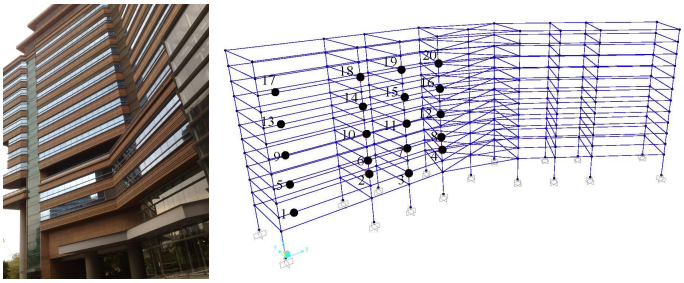
To find out the actual system lifetime that can be achieved, we simply iterates the procedures above until no SHM cover sets are available. In this experiment, the voltage of each mote is monitored via an onboard voltage sensor. According to the discharge characteristics of lithium battery, a SenetSHM mote will shut itself down if its voltage falls down below 3.3v. Obviously, once a mote is shut down, any cover sets that containing this mote will not be implemented for damage detection.

In this test, we found that the actual number of rounds that the system can work is 79, which is smaller than the expected 100. The reason for the gap can be mainly attributed to the error of the parameters utilized in the BMKP method. In addition, the energy consumption model utilized in BMKP method does not include the energy consumed for control messages, retransmissions, idle listening and other overhead. Nevertheless, considering the number of rounds that the system can work is less than 7 if all the deployed SenetSHM motes are activated (which can be easily calculated by applying parameters in Table 2 to Eq. 8), using the EECPS can significantly improve the system lifetime.

In terms of accuracy, Fig. 13 (c) shows the error of identified natural frequencies in each round of damage detecting. It shows that using $\gamma = 200$, the error of identified natural frequencies is less than 2% in most rounds of damage detection, while a few large values (up to 6%) can be attributed to the abrupt noise in the measurements.

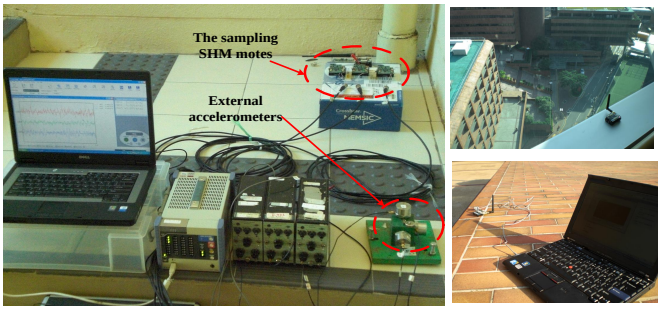
5.3 Experiment on a large civil infrastructure

We have tested our proposed scheme on a large civil infrastructure. We deployed a number of SenetSHM motes in the LSK Building to measure its vibration under ambient environment (see Fig. 14 (a)). The numbering of the measured locations is shown in Fig 14 (b). The system setup on measurement location 17 is illustrated in Fig. 15 (a). At each measurement location, we use three high-sensitive external sensor named KD1300 to record the vibration (see Fig.15 (a)) in three orthogonal directions. Signals recorded at each KD1300 are amplified and then fed into a SHM mote. For convenience, we call the SHM motes connecting to the KD1300 the sampling motes. In this experiment, wireless communication cannot be directly established among the sampling motes at different locations since the locations shown in Fig. 14 (b) belong to different rooms. To solve this problem, we deploy a particular SenetSHM mote acting as local data collector deployed near the window of each location (see the top figure of Fig. 15 (b)). Vibrational data sampled at the three sampling motes are



(a) (b)

Fig. 14: The LSK building and measurement locations (a) The LSK building (b) 20 measurement locations



(a) (b)

Fig. 15: Experiment setup (a) Sampling motes and sensors at a certain location(b)Top: a collector node deployed near the window; Bottom: a gateway node connected to a laptop.

first transmitted to this collector node. These 20 collector nodes can be regarded as independent wireless sensor nodes in this WSN. In this paper, we assume the collector nodes already have the local vibration data and only consider the wireless communication among them.

During the test, we first find the network topology of the collector nodes by running the collection tree protocol (CTP) [25] on the collector nodes. The network topology of these 20 collector nodes is illustrated in Fig. 16 (a). The topology information is transmitted to a gateway node that is connected to a laptop computer (see the bottom figure of Fig. 15 (b)). In addition, considering each SenetSHM mote has a large onboard μSD card, we collect the data from all the collector nodes after the experiment is finished. These data are utilized to obtained mode shapes and natural frequencies. As before, the former will be utilized when calculating the condition number and latter will serve as the reference when estimating error of SHM cover sets.

Since we cannot generate damage on the structure, we simply set the threshold $\gamma = 200$. Based on the network topology, mode shapes and γ, we implement the BMKP-based method at the laptop computer and it generates a total of 52 SHM cover sets. The number of rounds assigned to each set is illustrated in Fig. 16 (b). The expected number of rounds that these SHM cover sets can work is thus 150. The information of SHM cover sets is broadcast to all the collector nodes. Each collector

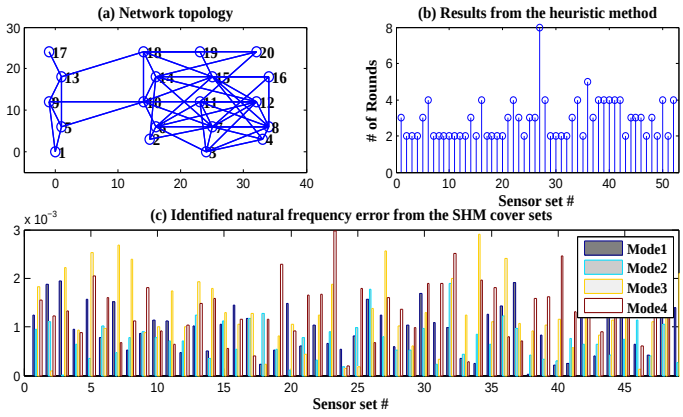


Fig. 16: (a) The topology of the collector nodes (b) The SHM cover sets obtained using the BMKP method and (c) the corresponding natural frequency error

node is then aware of the SHM cover sets it belongs and its status (CH or cluster member) in these sets.

Then we implement the EECPS on this WSN. The basic procedure is similar to that described in Section 5.2. The only difference is that in each round of damage detection, all the sampling motes corresponding to the collector nodes in a cover set will wake up and synchronize using the modified flooding time synchronization protocol (FTSP) [26].

The actual number of working rounds that the system can work is 49. Obviously, the difference between the expected and actual system lifetime is mainly caused by the fact that the energy consumed when the collectors were receiving data from sampling motes is not taken into consideration. Fig. 16 (c) shows the error of the identified natural frequencies in each SHM cover sets. It can be seen that all the SHM cover sets are able to identify the natural frequencies accurately.

6 Related Works

According to how the full coverage is achieved, existing EECPS protocols can be largely categorized into centralized and distributed. In [2] and [3], centralized protocols based on the set cover problem are proposed which allocate sensor nodes into the maximum number of mutually exclusive sets, where each set completely covers the area. A improved version was proposed in [4], where sensors are allowed to participate in multiple sets.

When designing distributed protocols, the basic idea is that each active sensor node first exchanges information with the active neighbors to see whether or not its sensing region has already been covered by the neighbors and then activates or goes to sleep accordingly. In [5], a distributed coverage configuration protocol (CCP) is proposed which can configure a sensor network to any coverage degree. In [6], a distributed algorithm is proposed that ensures complete coverage using the concept of 'sponsored area'. Similar approach can be found in [27], where an Optimal Geographical Density

Control (OGDC) algorithm is proposed to achieve energy efficient monitoring by controlling the density of the active nodes.

However, both centralized and distributed methods mentioned above would fail in some applications of WSNs where detection events or targets requires collaboration from multiple sensor nodes. Since individual sensor node is not able to fulfill the task alone, a definite coverage area for individual sensor node does not exist.

7 Discussion and Conclusion

In most existing WSN applications, a task can be fulfilled by 'sensor nodes comparing their measurements to a threshold'. While in some domain-specific applications, to accomplish a task relies on low-level collaboration of multiple sensors, which is usually characterized by matrix computations. Different from existing applications, we cannot define coverage area or analogous contribution factor for individual nodes. Thus existing works which partition sensor nodes into multiple cover sets will fail.

In this paper, using an example of SHM, we extend the EECPS to more 'domain-specific' applications of WSNs. In particular, we re-define the 'coverage' and propose two methods to partition the deployed sensor nodes into qualified cover sets such that system lifetime can be maximized.

Besides SHM, volcano tomography shown in Fig. 2 is another example in which data from deployed sensor nodes must collaborate to solve a large matrix inversion problem. The way how coverage is re-defined, how problem is formulated and the methods proposed in this paper can give good guidance of selecting the optimal subsets of deployed nodes to obtain accurate tomography results in an energy-efficient manner. In addition, we believe that with the advancement of MEMS technology, WSNs will penetrate into more and more domain specific applications which feature collaborative and complicated signal processing algorithms.

References

[1] A. Mainwaring and et al., "Wireless sensor networks for habitat monitoring," in *Proceedings of the 1st ACM international workshop on Wireless sensor networks and applications*. ACM, 2002, pp. 88–97.

[2] S. Slijepcevic and M. Potkonjak, "Power efficient organization of wireless sensor networks," in *Communications, 2001. ICC 2001. IEEE International Conference on*, vol. 2. IEEE, 2001, pp. 472–476.

[3] S. Gao and et al., "Sensor scheduling for k-coverage in wireless sensor networks," *Mobile Ad-hoc and Sensor Networks*, pp. 268–280, 2006.

[4] M. Cardei, M. Thai, Y. Li, and W. Wu, "Energy-efficient target coverage in wireless sensor networks," in *INFOCOM 2005. Proceedings IEEE*, vol. 3. IEEE, 2005, pp. 1976–1984.

[5] X. Wang and et al., "Integrated coverage and connectivity configuration in wireless sensor networks," in *Proceedings of the 1st international conference on ENSS*. ACM, 2003, pp. 28–39.

[6] D. Tian and N. Georganas, "A coverage-preserving node scheduling scheme for large wireless sensor networks," in *Proceedings of the 1st ACM workshop on WSNs and applications*, 2002, pp. 32–41.

[7] P. Van Overschee, "Subspace identification for linear systems: Theory, implementation," 1996.

[8] S. Doebling, C. Farrar, M. Prime, and D. Shevitz, "Damage identification and health monitoring of structural and mechanical systems from changes in their vibration characteristics: a literature review," Los Alamos National Lab., NM (United States), Tech. Rep., 1996.

[9] L. Shi, W.-Z. Song, and et al., "Imaging seismic tomography in sensor network," in *10th AnnualIEEE SECON*. IEEE, 2013, pp. 327–335.

[10] J. Hwang, T. He, and Y. Kim, "Exploring in-situ sensing irregularity in wireless sensor networks," *IEEE Transactions on Parallel and Distributed Systems*, pp. 547–561, 2009.

[11] C. Farrar and K. Worden, "An introduction to SHM," *Philosophical Transactions of the Royal Society*, vol. 365, no. 1851, p. 303, 2007.

[12] J. Juang and R. Pappa, "An eigensystem realization algorithm for modal parameter identification and model reduction," *Journal of Guidance*, vol. 8, no. 5, pp. 620–627, 1985.

[13] L. Yao, W. Sethares, and D. Kammer, "Sensor placement for on-orbit modal identification via a genetic algorithm," *AIAA journal*, vol. 31, no. 10, pp. 1922–1928, 1993.

[14] M. Garey and D. Johnson, *Computers and Intractability: A Guide to the Theory of NP-completeness*. WH Freeman & Co., 1979.

[15] S. Martello and P. Toth, *Knapsack problems: algorithms and computer implementations*. John Wiley & Sons, Inc., 1990.

[16] Z. Botev and D. Kroese, "An efficient algorithm for rare-event probability estimation, combinatorial optimization, and counting," *Methodology and Computing in Applied Probability*, vol. 10, pp. 471–505, 2008.

[17] D. E. Goldberg and J. H. Holland, "Genetic algorithms and machine learning," *Machine learning*, vol. 3, no. 2, pp. 95–99, 1988.

[18] A. Smith and D. Tate, "Genetic optimization using a penalty function," in *Proceedings of the 5th International Conference on Genetic Algorithms*. Morgan Kaufmann Publishers Inc., 1993, pp. 499–505.

[19] P. Chu and J. Beasley, "Constraint handling in genetic algorithms: the set partitioning problem," *Journal of Heuristics*, vol. 4, no. 4, pp. 323–357, 1998.

[20] Chu and Beasley, "A genetic algorithm for the generalised assignment problem," *Computers & Operations Research*, vol. 24, no. 1, pp. 17–23, 1997.

[21] E. Wilson and A. Habibullah, "Sap2000łstructural analysis users manual," *Computers and Structures, Inc*, 1998.

[22] D. C. Kammer, "Sensor placement for on-orbit modal identification and correlation of large space structures," *Journal of Guidance, Control, and Dynamics*, vol. 14, no. 2, pp. 251–259, 1991.

[23] X. Liu, J. Cao, and S. Tang, "Enabling fast and reliable network-wide event-triggered wakeup in wsns," in *IEEE 34th Real-Time Systems Symposium (RTSS)*. IEEE, 2013, pp. 278–287.

[24] M. Friswell and J. E. Mottershead, *Finite element model updating in structural dynamics*. Springer, 1995, vol. 38.

[25] K. S. P. R. Fonseca, O. Gnawali and A.Woo, " TEP 123: Collection Tree Protocol, http://www.tinyos.net/tinyos-2.x/doc/."

[26] M. Maróti, B. Kusy, G. Simon, and Á. Lédeczi, "The flooding time synchronization protocol," in *Sensys 2004*. ACM, 2004, pp. 39–49.

[27] H. Zhang and J. Hou, "Maintaining sensing coverage and connectivity in large sensor networks," *Urbana*, vol. 51, p. 61801.