Connectivity-Aware Network Maintenance and Repair via Relays Deployment

Ahmed S. Ibrahim, *Student Member, IEEE,* Karim G. Seddik, *Student Member, IEEE,* and K. J. Ray Liu, *Fellow, IEEE*

***Abstract*—In this paper we address the network maintenance problem, in which we aim to maximize the lifetime of a sensor network by adding a set of relays to it. The network lifetime is defined as the time until the network becomes disconnected. The Fiedler value, which is the algebraic connectivity of a graph, is used as an indicator of the network health. The network maintenance problem is formulated as a semi-definite programming (SDP) optimization problem that can be solved efficiently in polynomial time. First, we propose a network maintenance algorithm that obtains the SDP-based locations for a given set of relays. Second we propose a routing algorithm, namely, Weighted Minimum Power Routing (WMPR) algorithm, that significantly increases the network lifetime due to the efficient utilization of the deployed relays. Third, we propose an adaptive network maintenance algorithm that relocates the deployed relays based on the network health indicator. Further, we study the effect of two different transmission scenarios, with and without interference, on the network maintenance algorithm. Finally, we consider the network repair problem, in which we find the minimum number of relays along with their SDP-based locations to reconnect a disconnected network. We propose an iterative network repair algorithm that utilizes the network maintenance algorithm.**

***Index Terms*—Connectivity, power control, routing, semi-definite programming, sensor networks.**

I. Introduction

RECENTLY, there have been much interest in wireless sensor networks due to its various application areas such as battlefield surveillance systems, target tracking, and industry monitoring systems [1]. A sensor network consists of a large number of sensor nodes, which are deployed in a particular area to measure certain phenomenon such as temperature and pressure. These sensors send their measured data to a central processing unit (information sink), which collects the data and develops a decision accordingly. Often sensors have limited energy supply. Hence efficient utilization of the sensors' limited energy, and consequently extending the network lifetime, is one of the design challenges in wireless sensor networks.

The network lifetime is defined in this paper as the time until the network becomes disconnected. The network is considered connected if there is a path, possibly a multi-hop one, from each sensor to the central processing unit. In various applications, sensors are deployed randomly in the field and there is no much control over the specific location of each sensor. In the scenario where relays are available, it could be possible to deploy relays in some particular locations to enhance the network performance and extend its lifetime. An example is that low-altitude unmanned air vehicle (UAV) can perform as a relay that can be deployed in particular locations. Throughout this work, we assume that the deployed relays have the same capability as that of the sensors. Particularly, the relays forward the received data without any processing operations.

Deploying a set of relays in a wireless sensor network is one of the main approaches to extend the network lifetime. More precisely, relays can forward the sensors' data and hence they contribute to reducing the transmission power required by many sensors per transmission, which can extend the lifetime of these sensors. However, the problem of finding the optimum locations of these relays is shown to be NP-hard [2]. Therefore, there is a need to find a heuristic algorithm that can find good locations for the available set of relays in polynomial time. This problem is referred to in the literature as *network maintenance* problem.

In wireless sensor networks and after deploying the sensors for a while, some sensors may lose their available energy, which affects each sensor's ability to send its own data as well as forward the other sensors' data. This affects the network connectivity and may result in the network being disconnected. In this case, there is a need to determine the minimum number of relays along with their optimum locations that are needed to reconnect this network. Similar to the network maintenance problem, this problem is NP-complete [3] and there is a need for a heuristic algorithm to solve this problem in polynomial time. This problem is referred to as *network repair* problem.

In this paper, we address the network maintenance and network repair problems in wireless sensor networks. We propose various *cross-layer* algorithms for relay deployment and data routing, which are jointly designed across the physical and network layers. First, we propose an efficient network maintenance algorithm that finds heuristic locations for an available set of relays to extend the network lifetime. The network connectivity and consequently the network lifetime are quantified via the Fiedler value, which is the algebraic connectivity of the network graph. The Fiedler value is equal to the second smallest eigenvalue of the Laplacian matrix representing the network graph. The proposed network maintenance algorithm aims at formulating the network lifetime problem as a semi-definite programming (SDP) optimization problem that can be solved in polynomial time.

Manuscript received January 11, 2008; revised June 13, 2008; accepted August 17, 2008. The associate editor coordinating the review of this paper and approving it for publication was X. Zhang.

A. S. Ibrahim and K. J. R. Liu are with the Department of Electrical and Computer Engineering, University of Maryland, College Park, MD, 20742 USA (e-mail: {asalah, kjrliu}@umd.edu).

K. G. Seddik is with the Electrical Engineering Department, Alexandria University, Egypt (e-mail: kseddik@gmail.com).

Digital Object Identifier 10.1109/T-WC.2009.080045

Building upon the proposed network maintenance algorithm, we propose a routing algorithm, namely, Weighted Minimum Power Routing (WMPR) algorithm, that can extend the network lifetime whenever the deployed relays have higher initial energy than that of the existing sensors. The WMPR assigns weights to the sensors that are different from that of the relays. It tends to use the relays more often and hence balance the network load among the existing sensors and relays, which results in longer network lifetime. Furthermore, we propose an adaptive network maintenance algorithm that increases the network lifetime by relocating the relays depending on the network status. We consider the Fiedler value of the remaining network as a good *network health indicator*. Finally, we propose an iterative network repair algorithm which finds the minimum number of relays along with their locations needed to reconnect a disconnected network.

The proposed network maintenance algorithms are applied in two different transmission scenarios depending on the employed medium access control protocol. First, we consider a zero-interference scenario where each node is assigned an orthogonal channel and hence there is no interference among the nodes. Second, we consider an interference-based scenario where a set of nodes is allowed to send simultaneously and hence causing interference to each other. We show that the transmission power required by each sensor per transmitted packet is higher in the interference-based scenario compared to that in the zero-interference scenario. Therefore in a limited-energy network setup, where network lifetime is of big concern, a zero-interference transmission scenario should be favorably considered to extend the network lifetime.

The remainder of this paper is organized as follows. In the next section, we summarize some related work. In Section III, we describe the system model and present a brief revision on the algebraic connectivity of a graph. We formulate the network maintenance problem and describe the proposed algorithm in Section IV. We build upon that algorithm and propose different strategies to increase the network lifetime in Section V. In Section VI, we address the network repair problem and describe the proposed algorithm. In Section VII, we present some simulation results that show the significance of our proposed algorithms. Finally, Section VIII concludes the paper.

II. Related Work

In this section, we briefly review some of the existing network maintenance and network repair strategies in wireless sensor networks. Most of the previous works consider the time until the first sensor dies, i.e., runs out of energy, as the network lifetime. In sensor networks, sensors are usually deployed with large numbers and each area is often covered by more than one sensor. Therefore, there is a strong correlation in the sensors' information and that the death of one sensor may not affect the performance of the others sending their measurements to the central unit. Thus, we consider the time until the network becomes disconnected as the network lifetime [4], [5].

Recently, there have been numerous network maintenance algorithms [2], [6]–[8]. In [2], the problem of provisioning additional energy on the existing sensors along with deploying additional relays in two-tier wireless sensor networks was considered. It was shown that the problem of joint design of energy provisioning and relay node placement can be formulated as a mixed-integer nonlinear programming problem, which is NP-hard in general. A relay deployment algorithm that maximizes the minimum sensor lifetime by exploiting the cooperative diversity was proposed in [6]. In [8], a joint design of relay deployment and transmission power control was considered to maximize the network lifetime. In that work, there is no solution to deploy the relays in particular locations, instead the probability distribution of the relays' location is quantified. More precisely, the relay density is higher near the central unit.

There have been recent works that considered the connectivity in wireless sensor networks [9]-[12]. In [9], the problem of adding relays to improve the connectivity of multi-hop wireless networks was addressed. A set of designated points are given and the available relays must be deployed in a smaller set of these designated points. The set of relay locations, are determined based on testing all the designated points and choosing the combination, which results in higher connectivity measure. Obviously, this scheme is very complex as the network size increases. In [11], three random deployment strategies, namely, connectivity-oriented, lifetime-oriented, and hybrid-oriented, were proposed. However, there is no explicit optimization problem for maximizing the network lifetime in that work. A mathematical approach to positioning and flying an unmanned air vehicle (UAV) over a wireless ad hoc network in order to optimize the network's connectivity for better Quality of Service (QoS) and coverage was proposed in [12].

Several works have considered the network repair problem, in which the objective is to find the minimum number of relays needed to have a connected graph. This is the same problem as the Steiner minimum tree with minimum number of Steiner points and bounded edge length problem defined in [13], which is NP-hard. Several approximate algorithms have been proposed to solve it in [3], [14]–[16]. For instance, in [16] the proposed algorithm first computes the minimum spanning tree (MST) of the given graph, then it adds relays on the MST edges, which are not existing in the original graph. The connectivity improvement using Delaunay Triangulation [3] constructs a Delaunay Triangulation in the disconnected network and deploy nodes in certain triangles according to several criteria. The network repair problem has been generalized to k-connectivity, both in the sense of edge and vertex connectivity, in [17].

Finally, we point out some of the unique aspects of our work compared to the existing works summarized above. First, the topology model is based on some of the physical layer parameters. More precisely, the graph edges are constructed based on the desired bit error rate, maximum transmission power of the sensors, noise variance, and Rayleigh fading channel model parameters. This helps in proposing cross-layer design of relay deployment and data routing schemes. Second the Fiedler value, which is a good measure of the connectivity, is considered as the network health indicator. Third, the main relay deployment algorithm is less complex than the previously proposed algorithms, because it is based on a SDP formulation, which can be solved in polynomial time.

III. System Model

In this section, first we describe the wireless sensor network model. Second, we derive the required transmission power to achieve a particular Quality of Service (QoS), which is the bit error rate in this work. Finally, we briefly review some concepts related to the spectral graph theory.

A wireless sensor network can be modeled as an undirected weighted simple finite graph $G(V,E)$, where $V = \{v_1, v_2, \cdots, v_n\}$ is the set of all nodes (sensors) and E is the set of all edges (links). An undirected graph implies that all the links in the network are bidirectional, hence, if node v_i can reach node v_j then the opposite is also true. A simple graph means that there is no self loop in each node and there are no multiple edges connecting two nodes. Finally, a finite graph implies that the cardinality of the sets V and E is finite. Let n and m denote the number of nodes and edges in the graph, respectively, i.e., $|V| = n$ and $|E| = m$, where $|\,.\,|$ denotes the cardinality of the given set.

Without loss of generality, we assume that binary phase shift keying (BPSK) modulation scheme is employed for the transmission between any two nodes. BPSK is primarily chosen since the data rate in most of the sensor network applications is relatively low, and the BPSK modulation is an intuitive choice for such applications. We point out that the proposed algorithms can be easily applied with other modulation types as well. Let $d_{i,j}$ denote the distance between two nodes $\{v_i, v_j\} \in V$ and let α denote the path loss exponent. The channel between each two nodes $\{v_i, v_j\} \in V$, denoted by $h_{i,j}$, is modeled as a complex Gaussian random variable with zero-mean and variance equal to $d_{i,j}^{-\alpha}$, i.e., $h_{i,j} \sim CN(0, d_{i,j}^{-\alpha})$. The channel gain $|h_{i,j}|$ follows a Rayleigh fading model [[18], Ch.14]. Furthermore, the channel gain squared $|h_{i,j}|^2$ is an exponential random variable with mean $d_{i,j}^{-\alpha}$, i.e., $p(|h_{i,j}|^2) = d_{i,j}^{\alpha} \exp(-|h_{i,j}|^2 \, d_{i,j}^{\alpha})$ is the probability density function (pdf) of $|h_{i,j}|^2$. The noise in each transmission is modeled as a Gaussian random variable with zero-mean and variance N_0.

Without loss of generality, we assume the zero-interference transmission scenario[1], in which sensors transmit their data over orthogonal channels whether in time or frequency domain. For instance, we consider the Time Division Multiple Access (TDMA) scenario. The transmission from node v_i to v_j can be modeled as

$$y_j = \sqrt{P_i}\, h_{i,j}\, x_i + n_j \ , \tag{1}$$

where x_i is the transmitted symbol with unit energy, i.e., $|x_i|^2 = 1$. In (1), P_i is the transmitted power, y_j is the received symbol, and n_j is the added noise term.

The probability of bit error, or bit error rate (BER), can be calculated as [[18], Ch14]

$$\varepsilon = \frac{1}{2}\Big(1 - \sqrt{\frac{\gamma_{i,j}}{1+\gamma_{i,j}}}\Big) \ , \tag{2}$$

where $\gamma_{i,j} = \frac{P_i\, d_{i,j}^{-\alpha}}{N_0}$ denotes the average signal-to-noise ratio (SNR). The transmission power of node v_i, required to achieve a desired average BER of ε^o over link (v_i, v_j), can be calculated from (2) as

$$P_i^o = d_{i,j}^{\alpha}\, N_0\, \frac{(1-2\,\varepsilon^o)^2}{1-(1-2\,\varepsilon^o)^2} \ , \tag{3}$$

which is the required transmission power for the zero-interference transmission scenario.

We assume that each node $v_i \in V$ can transmit with power $0 \leq P_i \leq P_{max}$, where P_{max} denotes the maximum transmission power of each node. Also, we assume that the noise variance and the desired BER are constant for all the transmissions in the network. Therefore, an undirected weighted edge (v_i, v_j) exists if $P_i^o \leq P_{max}$, where P_i^o is calculated as in (3). Furthermore, the weight of an edge l connecting v_i and v_j, denoted by $w_{i,j}$ or w_l, is a function of the transmitted power P_i^o that depends on the considered routing scheme, as will be discussed in Section V-A.

For an edge l, $1 \leq l \leq m$, connecting nodes $\{v_i, v_j\} \in V$, define the edge vector $\mathbf{a_l} \in \mathbf{R^n}$, where the i-th and j-th elements are given by $a_{l,i} = 1$ and $a_{l,j} = -1$, respectively, and the rest is zero. The *incidence* matrix $\mathbf{A} \in \mathbf{R^{n\times m}}$ of the graph G is the matrix with l-th column given by $\mathbf{a_l}$. The weight vector $\mathbf{w} \in \mathbf{R^m}$ is defined as $\mathbf{w} = [w_1, w_2, ..., w_m]^T$, where T denotes transpose.

The *Laplacian* matrix $\mathbf{L} \in \mathbf{R}^{n\times n}$ is defined as

$$\mathbf{L} = \mathbf{A}\,\mathrm{diag}(\mathbf{w})\,\mathbf{A}^T = \sum_{l=1}^{m} w_l\, \mathbf{a_l}\, \mathbf{a_l}^T \ , \tag{4}$$

where $\mathrm{diag}(\mathbf{w}) \in \mathbf{R}^{m\times m}$ is the diagonal matrix formed from $\mathbf{w}$. The diagonal entry $L_{i,i} = \sum_{j\in N(i)} w_{i,j}$, where $N(i)$ is the set of neighboring nodes of node v_i that have a direct edge with node v_i. $L_{i,j} = -w_{i,j}$ if $(v_i, v_j) \in E$, otherwise $L_{i,j} = 0$. Since all the weights are nonnegative, the Laplacian matrix is positive semi-definite, which is expressed as $\mathbf{L} \succeq 0$. In addition, the smallest eigenvalue is zero, i.e., $\lambda_1(\mathbf{L}) = 0$. The second smallest eigenvalue of $\mathbf{L}$, $\lambda_2(\mathbf{L})$, is the algebraic connectivity of the graph G [19]–[22]. It is called *Fiedler value* and it measures how connected the graph is because of following main reasons. First, $\lambda_2(\mathbf{L}) > 0$ if and only if G is connected and the multiplicity of the zero-eigenvalue is equal to the number of the connected sub-graphs. Second, $\lambda_2(\mathbf{L})$ is monotone increasing in the edge set, i.e.,

$$\begin{aligned} &\text{if} \quad G_1 = (V, E_1)\ , G_2 = (V, E_2)\ , E_1 \subseteq E_2 \\ &\text{then} \quad \lambda_2(\mathbf{L_1}) \leq \lambda_2(\mathbf{L_2})\ , \end{aligned} \tag{5}$$

where $\mathbf{L_q}$ denotes the Laplacian matrix of the graph G_q for $q = 1, 2$.

As we mentioned previously, the smallest eigenvalue of the Laplacian matrix is $\lambda_1(\mathbf{L}) = 0$. In addition, its corresponding eigenvector is the all-ones vector $\mathbf{1} \in \mathbf{R}^n$, as the sum of the elements in each row (column) is zero. Let $\mathbf{y} \in \mathbf{R}^n$ be the eigenvector corresponding to $\lambda_2(\mathbf{L})$, which has unity norm $||\mathbf{y}||_2 = 1$ and is orthogonal to the all-ones vector, i.e., $\mathbf{1}^T\,\mathbf{y} = 0$. Since, $\mathbf{L}\,\mathbf{y} = \lambda_2\,\mathbf{y}$, hence $\mathbf{y}^T\,\mathbf{L}\,\mathbf{y} = \lambda_2\,\mathbf{y}^T\,\mathbf{y} = \lambda_2$. Therefore, the Fiedler value can be expressed as the smallest

[1] The transmission scenario that takes into consideration the interference effect is a simple extension of the zero-interference scenario, and it will be addressed in Section VII-A

eigenvalue that satisfy these conditions, i.e.,

$$\lambda_2(\mathbf{L}) = \inf_{\mathbf{y}} \{\mathbf{y}^T \mathbf{L}\, \mathbf{y} \ , \ ||\mathbf{y}||_2 = 1, \ \mathbf{1}^T \mathbf{y} = 0\} \ . \qquad (6)$$

In this work, the network lifetime is defined as the time until the network becomes disconnected, which happens when there is no communication path from any existing sensor to the central unit [4], [5]. Consequently, the network dies (becomes disconnected) if there is no communication path between any two living nodes including the central unit. Therefore, there is a direct relation between keeping the network connected as long as possible and maximizing the network lifetime, as was shown in [4], [5]. As discussed before, the Fiedler value defines the algebraic connectivity of the graph and it is a good measure of how connected the graph is. Intuitively the higher the Fiedler value is, the more edges that exist between the nodes, the longer the network can live without being disconnected, and thus the higher the network lifetime is. Based on that, we consider the Fiedler value as a quantitative measure of the network lifetime. In Section VII, we will validate this direct relation between the Fiedler value and the network lifetime.

IV. Network Maintenance

The network maintenance problem can be stated as follows. Given a wireless network deployed in a $g \times g$ square area and represented by the graph $G_b = (V_b, E_b)$, as well as a set of K relays, what are the optimum locations for placing relays in order to maximize the Fiedler value of the resulting network? Intuitively, adding a relay to the network may result in connecting two sensors or more, which were not connected together. Because this relay can be within the transmission range of these sensors, hence it can forward data from one sensor to the other. Therefore, adding a relay may result in adding an edge or more to the original graph.

Let $E_c(K)$ denote the set of edges resulting from adding a candidate set of K relays. Thus, the network maintenance problem can be formulated as

$$\max_{E_c(K)} \ \lambda_2\Big(\mathbf{L}\big(E_b \cup E_c(K)\big)\Big) \ . \qquad (7)$$

Since each relay can be deployed anywhere in the network, the location of each relay is considered as a continuous variable, which belongs to the interval $([0, g], [0, g])$. It has been shown that this problem is NP-hard in [2]. In the following subsection, we explain our proposed heuristic algorithm to solve this problem.

A. SDP-based Network Maintenance Algorithm

Our proposed algorithm to solve the network maintenance problem in (7) can be described as follows. First, we divide the $g \times g$ network area into n_c equal square regions, each with width h. Thus, $n_c = (\frac{g}{h})^2$. We represent each region by a relay deployed in its center. Thus, the problem can be viewed as having a set of n_c candidate relays, hence the subscript c, and we want to choose the optimum K relays among these n_c relays. This optimization problem can be formulated as

$$\begin{aligned} &\max \ \lambda_2\big(\mathbf{L}(\mathbf{x})\big) \\ &\ \text{s.t.} \ \ \mathbf{1}^T \mathbf{x} = K, \ \mathbf{x} \in \{0,1\}^{n_c} \ , \end{aligned} \qquad (8)$$

where

$$\mathbf{L}(\mathbf{x}) = \mathbf{L}_b + \sum_{l=1}^{n_c} x_l \, \mathbf{A}_l \, \mathrm{diag}(\mathbf{w}_l) \, \mathbf{A}_l^T \ , \qquad (9)$$

and $\mathbf{1} \in \mathbf{R}^{n_c}$ is the all-ones vector.

We note that the optimization vector in (8) is the vector $\mathbf{x} \in \mathbf{R}^{n_c}$. The i-th element of $\mathbf{x}$, denoted by x_i, is either 1 or 0, which corresponds to whether this relay should be chosen or not, respectively. In (9), $\mathbf{L}_b$ is the Laplacian matrix of the base graph. In addition, $\mathbf{A}_l$ and $\mathbf{w}_l$ are the incidence matrix and weight vector resulting from adding relay l to the original graph. Assuming that adding relay l results in I_l edges between the original n nodes in the base network, then the matrix $\mathbf{A}_l$ can be formed as $\mathbf{A}_l = [\mathbf{a}_l^1, \mathbf{a}_l^2, \cdots, \mathbf{a}_l^{I_l}]$, where $\mathbf{a}_l^z \in \mathbf{R}^n$, $z = 1, 2. \cdots, I_l$, represents an edge between two original nodes. Similarly, $\mathbf{W}_l = [\mathbf{w}_l^1, \mathbf{w}_l^2, \cdots, \mathbf{w}_l^{I_l}]$. We point out that the effect of adding relays appears only in the edge set E, and not in the node set V. The weight of a constructed edge equals the summation of the weights of the edges connecting the relay with the two sensors. Finally, the constraint $\mathbf{1}^T \mathbf{x} = K$ in (8) indicates that the number of chosen relays is K.

The exhaustive search scheme to solve (8) is done by computing $\lambda_2(\mathbf{L})$ for different $\binom{n_c}{K}$ Laplacian matrices, which requires huge amount of computation for large n_c. Therefore, we need an efficient and quick way to solve (8). The optimization problem (8) can be thought of as a general version of the one considered in [20]. By relaxing the Boolean constraint $\mathbf{x} \in \{0,1\}^{n_c}$ to be a linear constraint $\mathbf{x} \in [0,1]^{n_c}$, we can represent the optimization problem in (8) as

$$\begin{aligned} &\max \ \lambda_2\big(\mathbf{L}(\mathbf{x})\big) \\ &\ \text{s.t.} \ \ \mathbf{1}^T \mathbf{x} = K, 0 \le \mathbf{x} \le 1 \ . \end{aligned} \qquad (10)$$

We note that the optimal value of the relaxed problem in (10) is an upper bound for the optimal value of the original problem (8), as it has a larger feasible set.

Similar to (6), the Fiedler value of $\mathbf{L}(\mathbf{x})$ can be expressed as

$$\lambda_2\big(\mathbf{L}(\mathbf{x})\big) = \inf_{\mathbf{y}} \{\mathbf{y}^T \mathbf{L}(\mathbf{x})\, \mathbf{y} \ , \ ||\mathbf{y}||_2 = 1, \ \mathbf{1}^T \mathbf{y} = 0\} \ . \quad (11)$$

It can be shown that $\lambda_2\big(\mathbf{L}(\mathbf{x})\big)$ in (11) is the point-wise infimum of a family of linear functions of $\mathbf{x}$. Hence, it is a concave function in $\mathbf{x}$. In addition, the relaxed constraints are linear in $\mathbf{x}$. Therefore, the optimization problem in (10) is a *convex optimization* problem [23]. Furthermore, the convex optimization problem in (10) is equivalent to the following semi-definite programming (SDP) optimization problem [20], [22]

$$\begin{aligned} &\max \ s \\ &\ \text{s.t.} \ \ s(\mathbf{I} - \frac{1}{n}\mathbf{1}\,\mathbf{1}^T) \preceq \mathbf{L}(\mathbf{x}), \ \mathbf{1}^T \mathbf{x} = K, \ 0 \le \mathbf{x} \le 1 \ , \end{aligned} \qquad (12)$$

where $\mathbf{I} \in \mathbf{R}^{n \times n}$ is the identity matrix and $\mathbf{B} \preceq \mathbf{A}$ denotes that $\mathbf{A} - \mathbf{B}$ is a positive semi-definite matrix.

By solving the SDP optimization problem in (12) efficiently using any SDP standard solver such as the SDPA-M software package [24], the optimization variable $\mathbf{x}$ is obtained. Then, we use a heuristic algorithm to obtain a near-optimal Boolean solution from the SDP solution. In this paper, we consider a simple heuristic, which is to set the largest K elements in the vector $\mathbf{x}$ to 1 and the rest to 0. The obtained Boolean vector is the solution of the original problem in (8). This described procedure will be repeated a few times, and each repetition is referred to as a level. As indicated earlier, each location x_k, $k = 1, 2, \cdots, K$, represents a square region of width h. Choosing $x_k = 1$ implies that the k-th region is more significant, in terms of the connectivity of the whole network, than other ones that have not been chosen.

In order to improve the current solution, we repeat the same procedure by dividing each k-th region into n_c smaller areas and representing each area by a relay at its center. Then, we find the heuristic location in these n_c regions to have the relay deployed there. This problem is the same as the one in (12) by setting $K = 1$ relay. The same procedure is repeated for each region k, $1 \leq k \leq K$, obtained in the first level. The proposed network-maintenance algorithm applies a finite number of levels until there is no more improvement in the resulting Fiedler value. Table I summarizes the implementation of our proposed network-maintenance algorithm.

We also discuss the complexity issue of the proposed network maintenance algorithm. The interior point algorithms for solving SDP optimization problems are shown to be polynomial in time [24]. Thus, the network maintenance algorithm which applies a small number of iterations, each requires solving SDP optimization problem, has a polynomial complexity in time. Finally, we point out that our network maintenance algorithm is also suitable for the kind of applications, where there is a possible locations for the relays to be deployed [9]. In this section, we have proposed a SDP-based network maintenance algorithm that deploys a finite number of relays to maximize the Fiedler value of the resulting graph and consequently the network lifetime. In the next section, we consider various strategies to increase the efficiency of the deployed relays.

V. Lifetime-Maximization Strategies

In this section, we build upon the network maintenance algorithm described in Table I and propose two strategies that can extend the network lifetime. First, we propose the WMPR algorithm, which efficiently utilizes the deployed relays in a wireless network. Second, we propose an adaptive network maintenance algorithm that relocates the relays based on the network status.

A. Weighted Minimum Power Routing (WMPR) Algorithm

In this subsection, first we explain the conventional Minimum Power Routing (MPR) algorithm then we present the proposed WMPR algorithm. The MPR algorithm constructs the minimum-power route from each sensor to the central unit, by utilizing the conventional Dijkstra's shortest-path algorithm [25]. The cost (weight) of a link (v_i, v_j) is given by

$$w_{i,j}|_{MPR} = P_i^o + P_r \ , \tag{13}$$

where P_i^o is the transmission power given in (3) and P_r denotes the receiver processing power, which is assumed to be fixed for all the nodes.

In (13), it is obvious that the MPR algorithm does not differentiate between the original sensors and the deployed relays while constructing the minimum-power route. In most of the applications, it is very possible that the few deployed relays have higher initial energy than that of the many existing sensors. Intuitively to make the network live longer, the relays should be utilized more often than the sensors. Consequently, the loads of the sensors and relays will be proportional to their energies, which results in more balanced network. The WMPR algorithm achieves this balance by assigning weights to the sensors and the relays, and the cost of each link depends on these weights. Therefore, we propose to have the weight of the link (v_i, v_j) given by

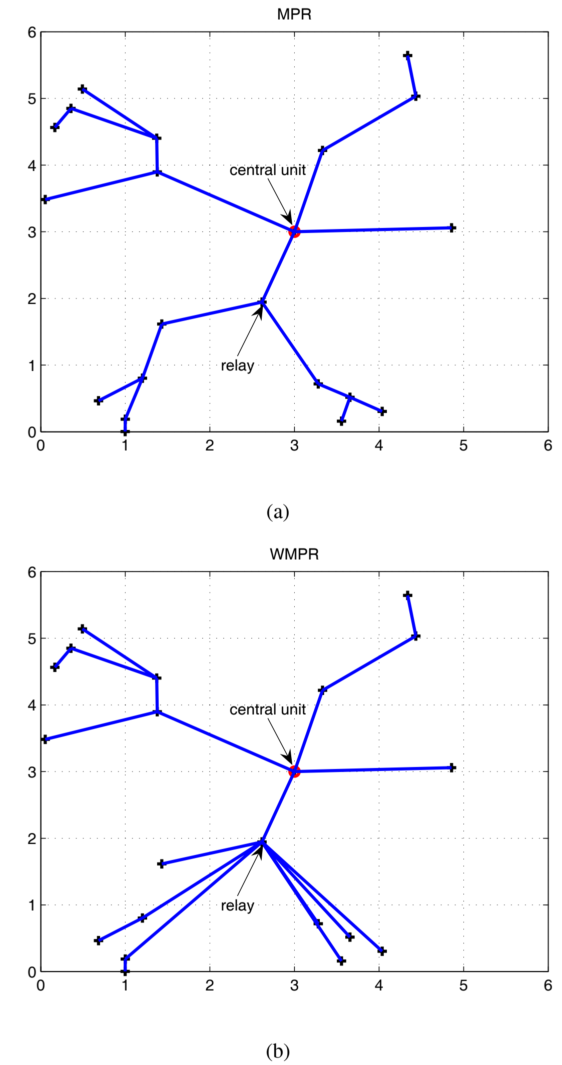

Fig. 1. Example of routing trees for $n = 20$ sensors deployed randomly in 6×6 square field (a) MPR-based constructed routing tree and (b) WMPR-based constructed routing tree.

$$w_{i,j}|_{WMPR} = e_i \, P_i^o + e_j \, P_r \ , \tag{14}$$

where e_i denotes the weight of node v_i. By assigning the relays smaller weight than that of the sensors, the network becomes more balanced and the network lifetime is increased. In summary, the WMPR utilizes the Dijkstra's shortest-path algorithm to compute the route from each sensor to the central unit using (14) as the link cost. More importantly, weights of the relays should be smaller than that of the sensors.

TABLE I
Proposed network maintenance algorithm.

```
Let $G_b = (V_b, E_b)$ be the original graph, $\mathbf{L}(K)$ be the Laplacian matrix of the resulting graph after adding the available $K$ relays, and $\lambda_{2,t}\big(\mathbf{L}(K)\big)$ be the Fiedler value at the $t$-th level (iteration).
1) Initialization: Set $t = 1$ and $\lambda_{2,0}(\mathbf{L}(K)) = \lambda_2(\mathbf{L_b}(0))$, where $\mathbf{L_b}$ is the Laplacian matrix of $G_b$.
2) Divide the network area into $n_c$ equal square regions. Each region is represented by a relay at its center.
3) Solve the optimization problem in (12) and obtain the best $K < n_c$ relays among the $n_c$ relays defined in 2. Denote the solutions as $x_k$, $k = 1, 2, \cdots, K$.
4) Calculate $\lambda_{2,t}\big(\mathbf{L}(K)\big)$, which is the Fiedler value of the resulting graph by constructing the Laplacian matrix of the resulting graph.
5) While $\Big(\lambda_{2,t}\big(\mathbf{L}(K)\big) > \lambda_{2,t-1}\big(\mathbf{L}(K)\big)\Big)$
     a) Increment the level index as: $t = t + 1$.
     b) For each solution $x_k$,
          i) Divide the $k$-th region into $n_c$ equal square regions and obtain the best location for this relay. This can be solved using (12) by setting $K = 1$.
        End For
     c) Calculate $\lambda_{2,t}\big(\mathbf{L}(K)\big)$ of the resulting graph.
   End While
6) The obtained solutions $x_k$, $k = 1, 2, \cdots, K$, represent the required locations of the relays.
```

Figure 1 depicts a sensor network of $n = 20$ nodes deployed randomly in 6×6 area. The central unit is located in the center of the network and we assume that $K = 1$ relay is available. The location of the relay is determined via the network maintenance algorithm, proposed in Table I. Each routing algorithm, either the MPR or the WMPR, constructs a tree connecting all the nodes together that has the minimum weight between each two nodes. In Figure 1 (a), the relay is treated in a similar fashion to that of the sensors in the MPR-based constructed routing tree. On the other hand Figure 1 (b) depicts that in the WMPR-based constructed routing tree, most of the sensors tend to send their packets to the relay rather than the neighboring sensors. As will be shown in Section VII, the WMPR algorithm achieves higher lifetime gain than that achieved by the MPR algorithm, when the deployed relays have more initial energy than the sensors. Finally, we point out that many of the lifetime-maximization routing algorithms [4], [5], [26] can be modified in a similar way to that of the WMPR algorithm.

B. Adaptive Network Maintenance Algorithm

In this subsection, we consider the possibility of relocating the deployed relays. In the fixed network maintenance strategy, as described in Table I, each relay will be deployed in a particular place and will be there until the network dies. Intuitively, the network lifetime can be increased by adaptively relocating the relays based on the status of the network. Such a scheme can be implemented via low-altitude Unmanned Air Vehicles (UAVs) or movable robots depending on the network environment. For instance, we can utilize one UAV or more, which can fly along the obtained relays' locations to improve the connectivity of the ground network. In each location, UAV acts exactly as a fixed relay connecting a set of sensors through multi-hop relaying.

The proposed adaptive network-maintenance algorithm is implemented as follows. First, the initial locations of the deployed relays are determined using the network-maintenance algorithm described in Table I. Whenever a node dies, the Fiedler value of the remaining network is calculated. If it is greater than certain threshold, then the network is likely to be disconnected soon. Therefore, the deployment algorithm is calculated again and the new relays' locations are obtained. Finally each relay is relocated to the new location, if it is different from its current one. The algorithm is repeated until the network is disconnected.

In the sequel, we present an example to illustrate how effective the adaptive network maintenance algorithm can be. Consider a wireless sensor network of $n = 20$ nodes deployed randomly in a 6×6 square area. We assume that only $K = 1$ relay is available. Whenever a node sends a packet, the remaining energy is decreased by the amount of the transmission energy and it dies when it has no remaining energy. In addition, the Fiedler value threshold is chosen to be 0.03.

Figure 2 depicts the Fiedler value of the network as a function of the number of dead nodes utilizing the MPR algorithm. The original network is disconnected after the death of 8 nodes. By adding a fixed relay, the network lifetime increases, resulting in a network lifetime gain of 31%. The network lifetime gain due to adding K relays is defined as $G(K) = \frac{T(K)-T(0)}{T(0)}$, where $T(K)$ is the network lifetime after deploying K relays. By considering $K = 1$ relay, the adaptive network-maintenance algorithm achieves lifetime gain of 70%. This example shows that the proposed adaptive network maintenance algorithm can significantly increase the network lifetime. We clarify that these lifetime gains are specific to that particular example and do not represent the average results. The average results of the various proposed network maintenance strategies are provided in Section VII.

It is worth to note that Figure 2 shows that the Fiedler value of the living network can be thought of as a *health indicator* of the network. If the network health is below certain

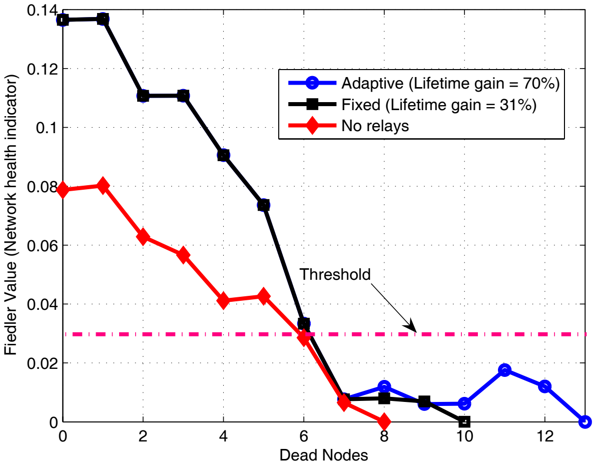

Fig. 2. Fiedler value (Network health indicator) versus the number of dead nodes, for $n = 20$ sensors deployed randomly in 6×6 square field, is plotted. Effects of adaptive and fixed network maintenance algorithms are illustrated.

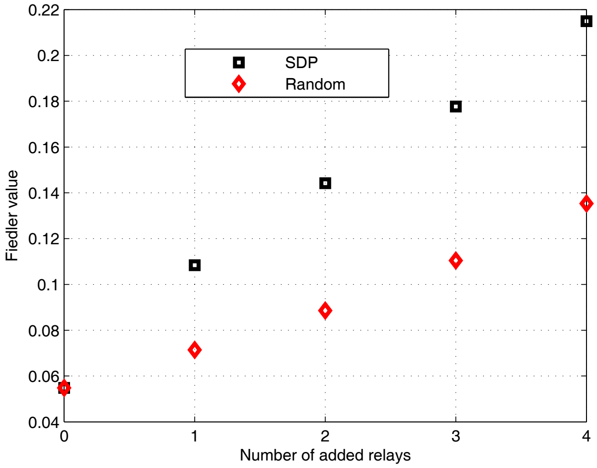

Fig. 3. The average Fiedler value versus the added number of relays, for $n = 20$ distributed randomly in 6×6 square field, is plotted. Effect of deploying relays is illustrated.

threshold, then the network is in danger of being disconnected. Thus, a network maintenance strategy, either fixed or adaptive, should be implemented. However, if the network becomes disconnected then intuitively we can consider reconnecting the network again via deploying the minimum number of relays. This is the network repair problem and it is discussed in the following section.

VI. Network Repair

In this section, we consider the network repair problem. In particular, the network is initially disconnected and we need to find the minimum number of relays along with their optimum locations in order to reconnect the network. Let a disconnected base network deployed in a $g \times g$ square area be represented by the graph $G_b = (V_b, E_b)$. Hence, $\lambda_2\Big(\mathbf{L}\big(E_b\big)\Big) = 0$. The network repair problem can be formulated as

$$\begin{aligned} &\min\ K \\ &\text{s.t.}\ \ \lambda_2\Big(\mathbf{L}\big(E_b \cup E_c(K)\big)\Big) > \delta\ , \end{aligned} \tag{15}$$

where $\delta > 0$ is referred to as *connectivity threshold* and it reflects the degree of desired robustness of the network connectivity and $E_c(K)$ denotes the set of edges resulting from adding a candidate set of K relays. We note that as δ increases the number of relays, required to satisfy the connectivity constraint in (15), increases.

In [3], it was shown that the network repair problem is NP-complete and hence we propose a heuristic algorithm to solve it. We utilize our proposed solution for the network maintenance problem in solving the network repair problem. More precisely, we propose an iterative network repair algorithm, which is implemented as follows. First, we assume that $K = 1$ relay is enough to reconnect the network. Second, we solve the network maintenance problem in Table I to find the location for that relay. If the Fiedler value of the resulting network is strictly greater than zero then the network is reconnected and the algorithm stops. Otherwise, the number of candidate relays is incremented by one and the algorithm is repeated.

Table II summarizes the proposed network repair algorithm. Similar to the network maintenance algorithm, the network repair algorithm is implemented in polynomial time. In this section, we have presented our proposed network repair problem and in the following section, we show some simulation results for the network maintenance and network repair proposed strategies.

VII. Simulation Results

In this section, we present some simulation results to show the performance of our proposed algorithms. We consider $n = 20$ nodes deployed randomly in 6×6 square area and the central unit is assumed to be in the center of the network. Data generated at the sensors follows a Poisson process with rate 10 packets per unit time and the path loss exponent is $\alpha = 2$. The desired BER for the transmissions over any link is $\varepsilon^o = 10^{-4}$, the noise variance $N_0 = -20$dBm, the maximum power $P_{max} = 0.15$ units, the receiver processing power is $P_r = 10^{-4}$ units, and the initial energy of every sensor is 0.1 unit. The number of candidate relays locations utilized in the network maintenance algorithm, described in Table I, is chosen to be $n_c = 25$ locations. The SDPA-M software package [24] has been utilized to solve the SDP optimization problem in (12). The following results are averaged over 1000 independent network realizations.

Figure 3 depicts the increase of the Fiedler value as the number of added relays increases. For comparison purposes, we also plot the effect of randomly adding relays. As shown, the random addition performs poorly compared to our proposed algorithm. In Section IV, we have chosen the Fiedler value as an intuitive and good measure of the network lifetime, which is our main objective. Figure 4 depicts the network lifetime gain as a function of the added number of relays. The network lifetime gain due to adding K relays is defined as

$$G_T(K) = \frac{T(K) - T_{MPR}(0)}{T_{MPR}(0)}\ \%\ , \tag{16}$$

TABLE II
Proposed network repair algorithm.

Let $G_b = (V_b, E_b)$ be the original graph, $\mathbf{L}(K)$ be the Laplacian matrix of the resulting graph after adding the available K relays, and $\lambda_2(\mathbf{L}(K))$ be its Fiedler value.

```
1) Initialization: Set $K = 0$.
2) While $(\lambda_2(\mathbf{L}(K) \leq \delta)$
     a) Increment the number of relays as: $K = K + 1$.
     b) Implement the network maintenance algorithm, described in Table I, utilizing $K$
        candidate relays.
     c) Calculate $\lambda_2(\mathbf{L}(K))$ of the resulting graph.
   End While
3) The obtained $K$ represents the minimum number of required relays.
```

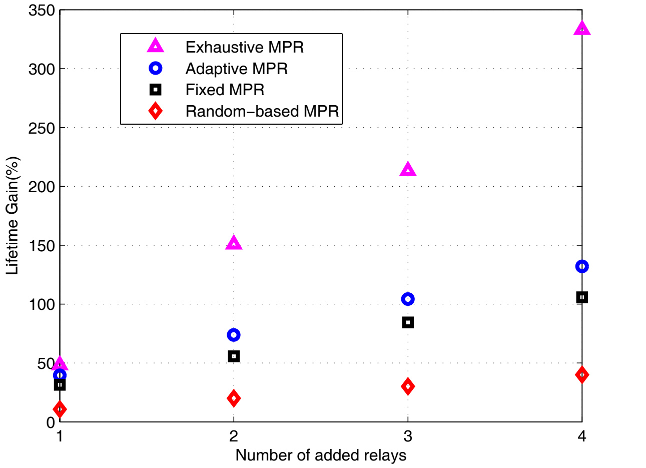

Fig. 4. The average network lifetime gain versus the added number of relays, for $n = 20$ distributed randomly in 6×6 square field, is plotted. Effect of deploying relays is illustrated.

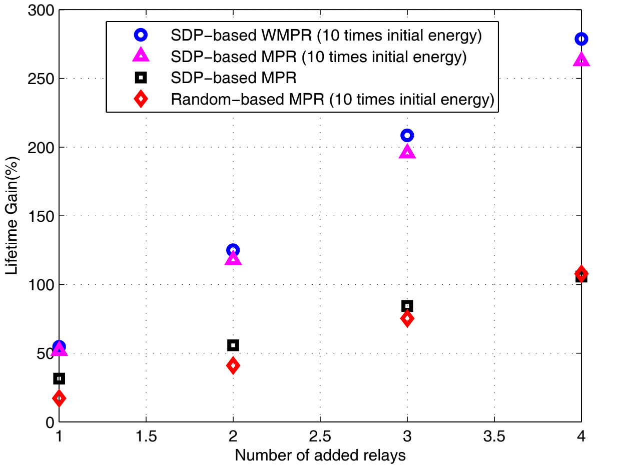

Fig. 5. The average network lifetime gain versus the added number of relays, for $n = 20$ distributed randomly in 6×6 square field, is plotted. Effect of increasing the relays' initial energy 10 times is illustrated.

where $T(K)$ is the network lifetime after deploying K relays and $T_{MPR}(0)$ denotes the network lifetime of the original network utilizing the MPR algorithm. As shown, the proposed SDP-based network maintenance algorithm achieves significant network lifetime gain as the number of added relays increases, which is a direct consequence of increasing the Fiedler value as shown previously in Figure 3. At $K = 4$ and by employing the MPR algorithm, the proposed network maintenance algorithm achieves lifetime gain of 105.8%, while the random deployment case achieves lifetime gain of 40.09%.

In Figure 4, we also illustrate the impact of the adaptive network maintenance algorithm on the network lifetime gain. At $K = 4$ relays, the lifetime gain jumps to 132.1% for the MPR algorithm. We also compare the performance of our proposed algorithm with the exhaustive search scheme. For practical implementation of the exhaustive search scheme, the optimum locations for a given set of relays are determined consecutively, i.e., one relay at a time. We have implemented the exhaustive search scheme by dividing the network area into many small regions and each region is represented by a relay at its center. The optimum location for the first relay is determined by calculating the lifetime of all the possible locations and choosing the one that results in maximum lifetime. Given the updated network including the first relay, we find the optimum location for the second relay via the same exhaustive search scheme. This algorithm is repeated until all the relays are deployed. In Figure 4, we show the network lifetime gain of the exhaustive search case utilizing the MPR algorithm.

As indicated in Section V-A, the proposed WMPR algorithm should intuitively outperform the MPR algorithm when relays have higher initial energy than that of the sensors. We set the weights of the deployed relays to be 0.1, while the weights of the original sensors to be 1. Therefore, sensors tend to send their data to the deployed relays rather than the neighboring sensors. In addition, the relays' energy are set to be 10 times that of the sensors. As a result, the WMPR algorithm achieves higher gain compared to that achieved by the MPR algorithm as shown in Figure 5. At $K = 4$, the WMPR and MPR algorithms achieve network lifetime gains of 278.8% and 262.7%, respectively. In Figure 5, we notice that the difference between the WMPR and the MPR performance curves increases as the number of relays increases. Intuitively, the WMPR algorithm utilizes the relays more frequently than the MPR algorithm. Hence it achieves higher lifetime gain by

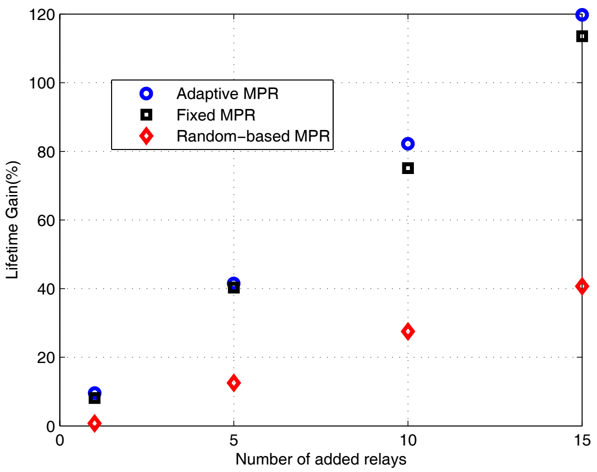

Fig. 6. The average network lifetime gain versus the added number of relays, for $n = 50$ distributed randomly in 15×15 square field, is plotted. Effect of deploying relays is illustrated.

increasing the the relays' initial energy.

We also consider a larger sensor network of $n = 50$ nodes deployed randomly in 15×15 square area. Figure 6 shows the network lifetime gain. At $K = 15$ and by employing the MPR algorithm, the proposed network maintenance algorithm achieves lifetime gain of 113.6%, while the random deployment case achieves lifetime gain of 40.7%. In Figure 6, we also illustrate the impact of the adaptive network maintenance algorithm on the network lifetime gain. At $K = 15$ relays, the lifetime gain jumps to 119.7% for the MPR algorithm.

A. *Interference-based Transmission Scenario*

In this subsection, we consider a different transmission scenario where some of the sensors are allowed to send their data simultaneously over the same channel. Assuming that node v_i is sending its data to node v_j and the total number of simultaneous transmissions is s. The received symbol can be modeled as

$$y_j = \sqrt{P_i}\, h_{i,j}\, x_i + \sum_{k=1,k\neq i}^{s} \sqrt{P_k}\, h_{k,j}\, x_k + n_j\,. \tag{17}$$

Let $m_j = \sum_{k\neq i} \sqrt{P_k}\, h_{k,j}\, x_i + n_j$ denote the random variable representing the summation of the noise and interference terms. For a large enough number of simultaneous transmissions, m_j can be modeled as a complex Gaussian random variable with zero-mean and variance $N_0 + \sum_{k\neq i} P_k\, d_{k,j}^{-\alpha}$ via the central limit theorem [[18], Ch.2], i.e., $m_j \sim CN(0, N_0 + \sum_{k\neq i} P_k\, d_{k,j}^{-\alpha})$. This is a reasonable assumption as the number of sensors, deployed in a sensor network, is often large. Thus, (17) can be written as (1) with different noise term, which is m_j. Consequently and similar to (3), the required power to achieve a desired BER of ε^o can be given by

$$P_i^o = d_{i,j}^{\alpha}\left(N_0 + \sum_{k\neq i} P_k\, d_{k,j}^{-\alpha}\right)\frac{(1-2\,\varepsilon^o)^2}{1-(1-2\,\varepsilon^o)^2}\,. \tag{18}$$

In (18), it is obvious that the transmission power required by each node depends on the transmission powers of the other nodes sending simultaneously over the same channel. We obtain an approximated power expression, by first approximating (18) as follows. At low BER, it can be easily shown that

$$P_i \approx \frac{N_0 + \sum_{k\neq i} P_k\, d_{k,j}^{-\alpha}}{4\,\varepsilon^o\, d_{i,j}^{-\alpha}}\,. \tag{19}$$

The transmission power can be determined through a power control problem, which can be formulated as the following optimization problem

$$\min \sum_i P_i \quad \text{s.t.} \quad \frac{N_0 + \sum_{k\neq i} P_k\, d_{k,j}^{-\alpha}}{4\, P_i\, d_{i,j}^{-\alpha}} \le \varepsilon^o\,, \tag{20}$$

Let $\mathbf{p} \in \mathbf{R}^s$ be the power vector, containing the transmission powers P_i, that needs to be calculated. Hence, (20) can be formulated in a matrix form as

$$\min \sum_i P_i \quad \text{s.t.} \quad \left(\mathbf{I} - \frac{1}{4\,\varepsilon^o}\mathbf{F}\right)\mathbf{p} \ge \mathbf{u}\,, \tag{21}$$

where $\mathbf{I} \in \mathbf{R}^{s\times s}$ is the identity matrix and the i-th element of the vector $\mathbf{u} \in \mathbf{R}^s$ is $u_i = \frac{N_0}{4\,\varepsilon^o\, d_{i,j}^{-\alpha}}$. With respect to $\mathbf{F} \in \mathbf{R}^{s\times s}$, $F_{i,j} = 0$ if $i = j$ and $F_{i,j} = (\frac{d_{k,j}}{d_{i,j}})^{-\alpha}$ elsewhere. If the spectral radius of $\mathbf{F}$, which is its largest eigenvalue, is less than $(4\,\varepsilon^o)$, then the minimum power set is given by [27], [28]

$$\mathbf{p}^o = \left(\mathbf{I} - \frac{1}{4\,\varepsilon^o}\mathbf{F}\right)^{-1}\mathbf{u}\,. \tag{22}$$

At low BER, it can be shown the zero-interference required transmission power given in (3) can be approximated as $P_i \approx \frac{N_0}{4\,\varepsilon^o\, d_{i,j}^{-\alpha}}$. By comparing this power with that required for the interference-based transmission scenario given in (19), it is obvious that the interference-based transmission scenario requires more transmission power per node than that required in the zero-interference scenario for the same desired BER. Therefore, nodes will lose their energies with a faster rate in the interference-based transmission scenario. Consequently, the network lifetime is shorter in the interference-based transmission scenario. Therefore if limited batteries is a concern such as in sensor network, it is recommended to have orthogonal transmission between the nodes to maximize the network lifetime.

We consider a network of $n = 10$ nodes deployed randomly in 4×4 area. All the nodes operate in half duplex mode, i.e., no node is allowed to transmit and receive at the same node. In addition, nodes sending their data to the same destination are not allowed to send their data at the same time since this requires more complex receiver such as successive interference cancelation (SIC) decoder, which may not be possible for a simple sensor node to have. The route from each sensor to the central unit is determined based on the zero-interference transmission powers, given in (3). Then the transmission powers are modified according to (22) to represent the interference-based case.

In addition to the network lifetime, the number of the delivered packets from all the sensors to the central unit before the network dies is an important measure of the network performance. Figure 7 depicts the number of delivered packets versus the added number of relays for both the

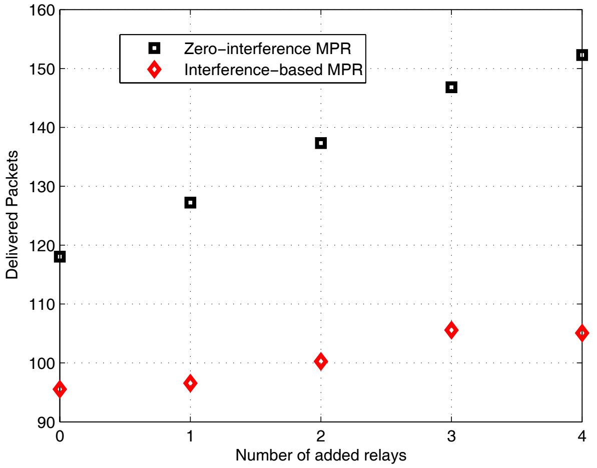

Fig. 7. The average number of delivered packets versus the added number of relays, for $n = 10$ distributed randomly in 4×4 square field, is plotted.

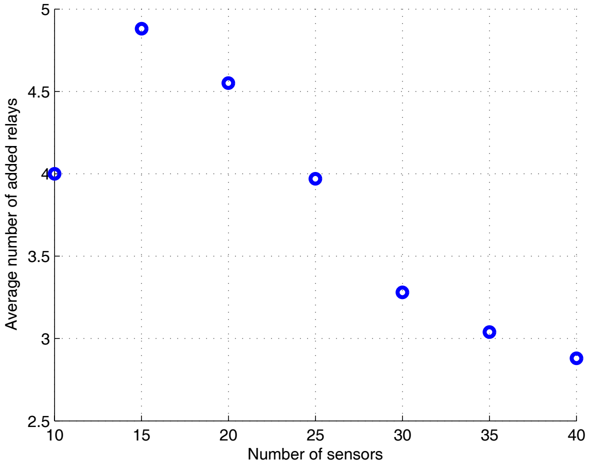

Fig. 8. The average minimum number of added relays required to reconnect a network versus the number of sensors in the network is plotted.

zero-interference and interference-based transmission scenarios utilizing the MPR algorithm. First, it is shown for the interference-based scenario that the number of delivered packets slightly increasing as the number of added relays increases. Generally, there are two main factors affecting the net result of the interference-based scenario whenever relays are deployed. Deploying relays increases the number of delivered packets due to performing the relaying task along with the extra energy that the deployed relays have. So, adding more relays increases the number of delivered packets, as shown previously for the zero-interference transmission scenario. On the other hand, deploying relays causes interference to the other existing nodes and forces each existing node to raise its transmission power to overcome the interference effect of the recently added relays. Thus, deploying relays will cause nodes to die faster and consequently will decrease the number of delivered packets. This is the main reason that the network lifetime gains are higher in the zero-interference transmission scenario compared to the interference-based scenarios. We note that the net result of these two factors will determine the performance of the interference-based network maintenance algorithms.

B. Network Repair

We consider the network repair problem where the network is originally disconnected. In Figure 8, we show the average number of added relays required to reconnect a disconnected network, assuming $\delta = 0$ in (15). n sensors are randomly distributed in 6×6 square area. The maximum transmission power of any node is $P_{max} = 0.07$. It is shown that for a disconnected network of $n = 25$ nodes deployed randomly in 6×6 area, the average number of added relays is 4. For $n < 15$, Figure 8 depicts that the average number of added relays increases as n increases. This is because for small n, it is more likely that the added sensors will be deployed in new regions where there are very few or no sensors. Thus, more relays need to be deployed to connect these added sensors. On the other hand, as n increases beyond $n = 15$, the average number of added relays decreases. This is intuitive because as the the number of sensors increases to a moderate state, the network becomes more balanced, i.e., the sensors are uniformly deployed in the whole area. Beyond this moderate state, increasing the number of sensors keeps filling the gaps in the network. Consequently, the average number of needed relays decreases as n increases.

VIII. Conclusion

In this paper, we have addressed the problems of network maintenance and network repair in wireless sensor networks. We have considered the Fiedler value, which is the algebraic connectivity of a graph, as a network health indicator. First, we have proposed a network maintenance algorithm, which finds the locations for an available set of relays that result in the maximum possible Fiedler value. This algorithm finds the location through a small number of levels. In each level, the network maintenance problem is formulated as a semi-definite programming (SDP) optimization problem, which can be solved using the available standard SDP solvers. In a sensor network of $n = 50$ sensors deployed in a 15×15 area, the network lifetime has increased by 113.6% due to the addition of 15 relays.

Second, we have proposed an adaptive network maintenance algorithm, where the relays' locations can be changed depending on the network health indicator. We have shown that a lifetime gain of 119.7% is achieved due to the proposed adaptive network maintenance algorithm. Third, we have proposed the Weighted Minimum Power Routing (WMPR) algorithm, which balances the load of the network among the sensors and the relays. We have also illustrated that in sensor networks, where sensors have limited supplies, nodes should transmit their data over orthogonal channels with no interference from the other nodes. Finally, we have proposed an iterative network repair algorithm, which finds the minimum number of relays needed to connect a disconnected network.

References

[1] I.F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "A survey on sensor networks," *IEEE Commun. Mag.*, vol. 40, pp. 102-114, Aug. 2002.

[2] Y. T. Hou, Y. Shi, H. D. Sherali, and S. F. Midkiff, “Prolonging sensor network lifetime with energy provisioning and relay node placement," in *Proc. IEEE Sensor and Ad Hoc Commun. and Networks (SECON’05)*, pp. 295-304, Sept. 2005.

[3] N. Li and J. C. Hou, “Improving connectivity of wireless ad hoc networks," in *Proc. Second Annual International Conference on Mobile and Ubiquitous Systems: Networking and Services (MobiQuitous’05)*, pp. 314-324, July 2005.

[4] C. Pandana and K. J. R. Liu, “Maximum connectivity and maximum lifetime energy-aware routing for wireless sensor networks," in *Proc. IEEE Global Telecommun. Conf. (Globecom’05)*, vol. 2, pp. 1034-1038, Nov. 2005.

[5] C. Pandana and K. J. R. Liu, “Robust connectivity-aware energy-efficient routing for wireless sensor networks," to appear in *IEEE Trans. Wireless Commun.*

[6] T. Himsoon, W. P. Siriwongpairat, Z. Han, and K. J. R. Liu, “Lifetime maximization via cooperative nodes and relay deployment in wireless networks," *IEEE J. Select. Areas Commun.,* special issue on cooperative commun. networking, vol. 25, no. 2, pp. 306-317, Feb. 2007.

[7] A. S. Ibrahim, K. G. Seddik, and K. J. R. Liu, “Connectivity-aware network maintenance via relays deployment," in *Proc. IEEE Wireless Comm. and Networking Conf. 2008 (WCNC’08)*, pp. 2573-2578, Apr. 2008.

[8] Y. Xin, T. Guven, and M. Shayman, “Relay deployment and power control for lifetime elongation in sensor networks," in *Proc. IEEE International Conf. on Commun. (ICC’06)*, vol. 8, pp. 3461-3466, June 2006.

[9] S. Maruyama, K. Nakano, K. Meguro, M. Sengoku, and S. Shinoda, “On location of relay facilities to improve connectivity of multi-hop wireless networks," in *Proc. 10th Asia-Pacific Conf. Commun. and 5th International Symposiym on Multi-Dimensional Mobile Commun.*, pp. 749-753, 2004.

[10] A. S. Ibrahim, K. G. Seddik, and K. J. R. Liu, “Improving connectivity via relays deployment in wireless sensor networks," in *Proc. IEEE Global Telecommun. Conf. (Globecom’07)*, pp. 1159-1163, Nov. 2007.

[11] K. Xu, H. Hassanein, and G. Takahara, “Relay node deployment strategies in heterogeneous wireless sensor networks: multiple-hop communication case," in *Proc. IEEE Sensor and Ad Hoc Commun. Networks (SECON’05)*, pp. 575-585, Sept. 2005.

[12] Z. Han, A. L. Swindlehurst, and K. J. R. Liu, “Smart deployment/movement of unmanned air vehicle to improve connectivity in MANET," in *Proc. IEEE Wireless Commun. Networking Conf.(WCNC’06)*, vol. 1, pp. 252-257, Apr. 2006.

[13] G. Lin and G. Xue, “Steiner tree problem with minimum number of steiner points and bounded edge-length," in *Inform. Processing Lett.*, vol. 69, pp. 53-57, 1999.

[14] D. Chen, D. Z. Du, X.-D. Hu, G. H. Lin, L. Wang, and G. Xue, “Approximations for steiner trees with minimum number of steiner points," *J. Global Optimization*, vol. 18, pp. 17-33, 2000.

[15] X. Cheng, D. Z. Du, L. Wang, and B. Xu, “Relay sensor placement in wireless sensor networks," *ACM/Springer J. Wireless Networks*, 2004.

[16] E. Lloyd and G. Xue, “Relay node placement in wireless sensor networks," *IEEE Trans. Comp.*, vol. 56, no. 1, pp. 134-138, Jan. 2007.

[17] A. Kashyap, S. Khuller, and M. Shayman, “Relay placement for higher order connectivity in wireless sensor networks," in *Proc. IEEE International Conf. Computer Commun. (INFOCOM’06)*, pp. 1-12, Apr. 2006.

[18] J. G. Proakis, *Digital Communications*, 4th ed. McGraw-Hill, 2000.

[19] M. Fiedler, “Algebraic connectivity of graphs," *Czechoslovak Lathematics J.*, pp. 298-305, 1973.

[20] A. Ghosh and S. Boyd, “Growing well-connected graphs," in *Proc. IEEE Conf. on Decision and Control (CDC’06)*, pp. 6605-6611, Dec. 2006.

[21] B. Mohar, “Some applications of laplace eigenvalues of graphs," in *Graph Symmetry: Algebraic Methods and Applications*, G. Hahn and G. Sabidussi, editors, vol. 497 of NATO ASI Series C, pp. 227-275. Kluwer, Dordrecht, 1997.

[22] S. Boyd, “Convex optimization of graph laplacian eigenvalues," in *Proc. International Congress of Mathematicians*, vol. 3, pp. 1311-1319, 2006.

[23] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge University Press, 2006.

[24] SDPA-M package, [Online]. Available: http://grid.r.dendai.ac.jp/sdpa/

[25] D. Bertsekas and R. Gallager, *Data Networks*, 2nd ed. Prentice Hall, 1991.

[26] J.-H Chang and L. Tassiulas,“Energy conserving routing in wireless ad hoc networks," in *Proc. IEEE International Conf. on Computer Communications (INFOCOM’00)*, pp. 21-30, Mar. 2000.

[27] F. Rashid-Farrokhi, L. Tassiulas, and K. J. R. Liu, “Joint optimal power control and beamforming in wireless networks using antenna arrays," *IEEE Trans. Commun.*, vol. 46, no. 10, pp. 1313-1324, Oct. 1998.

[28] F. Rashid-Farrokhi, K. J. R. Liu, and L. Tassiulas, “Transmit beamforming and power control for cellular wireless systems," *IEEE J. Select. Areas Commun.*, special issue signal processing for wireless commun., vol. 16, no. 8, pp. 1437-1450, Oct. 1998.

Ahmed S. Ibrahim (S’04) received the B.S. (with highest honors) and M.S. degrees in electrical engineering from Cairo University, Cairo, Egypt, in 2002 and 2004, respectively, and he is currently pursuing his Ph.D. degree at the Electrical and Computer Engineering Department, University of Maryland, College Park. His research interests include cooperative communications and networking, cross-layer design of wireless networks, relay deployment and selection in wireless networks, and MIMO-OFDM systems.

Mr. Ibrahim is a recipient of the best speaker award of the Electrical and Computer Engineering Department seminar series at the University of Maryland in 2008. He was a nominee for the best student paper award of the IEEE Conference on Acoustics, Speech, and Signal Processing (ICASSP) in 2007. He is also a recipient of the Graduate School Fellowship from the University of Maryland in 2004 and 2005.

Karim G. Seddik (S’04) is an assistant professor in the Electrical Engineering Department at Alexandria University, Egypt. Dr. Seddik received the B.S. (with highest honors) and M.S. degrees in electrical engineering from Alexandria University, Alexandria, Egypt, in 2001 and 2004, respectively. He received his Ph.D. degree at the Electrical and Computer Engineering Department, University of Maryland, College Park 2008.

His research interests include cooperative communications and networking, MIMO-OFDM systems, and distributed detection in wireless sensor networks.

Dr. Seddik is a recipient of the Graduate School Fellowship from the University of Maryland in 2004 and 2005 and the Future Faculty Program Fellowship from the University of Maryland in 2006.

K. J. Ray Liu (F’03) is Distinguished Scholar-Teacher of University of Maryland, College Park. He leads the Maryland Signals and Information Group conducting research encompassing broad aspects of information technology including communications and networking, information forensics and security, multimedia signal processing, biomedical imaging, and bioinformatics.

Dr. Liu is the recipient of numerous honors and awards including best paper awards from IEEE Signal Processing Society (twice), IEEE Vehicular Technology Society, and EURASIP; IEEE Signal Processing Society Distinguished Lecturer, EURASIP Meritorious Service Award, and National Science Foundation Young Investigator Award. He also received various teaching and research recognitions from University of Maryland including university-level Invention of the Year Award and both Outstanding Research Award and Poole and Kent Senior Faculty Teaching Award from A. James Clark School of Engineering Faculty.

Dr. Liu is Vice President - Publications and on the Board of Governor of IEEE Signal Processing Society. He was the Editor-in-Chief of IEEE SIGNAL PROCESSING MAGAZINE and the founding Editor-in-Chief of EURASIP JOURNAL ON APPLIED SIGNAL PROCESSING.

His recent books include *Cooperative Communications and Networking* (Cambridge University Press, 2008); *Resource Allocation for Wireless Networks: Basics, Techniques, and Applications* (Cambridge University Press, 2008); *Ultra-Wideband Communication Systems: The Multiband OFDM Approach* (IEEE-Wiley, 2007); *Network-Aware Security for Group Communications* (Springer, 2007); *Multimedia Fingerprinting Forensics for Traitor Tracing* (Hindawi, 2005).