Optimal path discovery for two moving sinks with a common junction in a wireless sensor network

Satish Tunga[1], Sadashiva V. Chakrasali[2], N. Shylashree[3], Latha B. N.[4], Mamatha A. S.[5]
[1]Department of Electronics & Telecommunication Engineering, M S Ramaiah Institute of Technology, Bengaluru, India
[2]Department of Electronics & Communication Engineering, M S Ramaiah Institute of Technology, Bengaluru, India
[3]Department of Electronics & Communication Engineering, RV College of Engineering, Bengaluru, India
[4]Department of Electronics & Communication Engineering, JSS Academy of Technical Education, Bengaluru, India
[5]Department of Electronics & Communication Engineering, St. Joseph Engineering College, Mangaluru, India

Article Info

Article history:

Received Oct 15, 2020
Revised Jun 1, 2021
Accepted Jun 19, 2021

Keywords:

Balanced loading
Disjoint paths
Participating edges
Participating nodes
Single common junction
Two moving sinks

ABSTRACT

A new algorithm is described for determining the optimal round-trip paths for two moving sinks in a wireless sensor network. The algorithm uses binary integer programming to select two non-overlapping shortest paths except having a common junction node to cover all the sensor nodes. The two paths are balanced as nearly equal as possible. That is the sensor nodes along each path are equal or differ by just one depending on whether the total number of sensor nodes excluding the junction node is even or odd. In this method, both the path lengths are made equal or very nearly equal while the total length is minimized. This integrated approach is a novel and unique solution to solve the dual moving sink path problem in a wireless sensor network.

Corresponding Author:

Shylashree N.
Department of Electronics & Communication Engineering
R V College of Engineering
R V Vidyanikethan Post, 8th Mile, Mysuru Road, Bengaluru 560059, Karnataka, India
Email: shylashreen@rvce.edu.in

1. INTRODUCTION

In general, a wireless sensor network (WSN) consists of static sensor nodes and one or more static sinks. The sensors send their sensed data to the designated sinks over multi-hop transmission. But, in those situations where the sensors are spread over a wide geographic area with large inter sensor distances, the sensors and the sinks may not be fully connected because of the insufficient communication range of the sensor nodes. Then the multi-hop transmission from sensors to the static sinks may not be possible. In such a situation, one mobile sink (MS) or more MSs (mobile collectors) are used to gather data from the sensors [1]-[13]. Mobile sinks physically move around the WSN and collect the data from the sensors. Mobile units carrying the sink can be ground based or airborne where unmanned aerial vehicles (UAVs) are utilized [14]-[17]. Here, we use multiple sinks (data collectors) mounted on ground based moving units controlled manually or by robots. It is assumed that the geographical region covered by the WSN is amenable for the physical navigation of the mobile sinks and they can smoothly approach the vicinity of the individual sensors. Our proposed method determines the best path for two mobile sinks to cover the WSN area satisfying the given constraints. The method can be extended to handle multiple sinks and also can optimize the paths of the moving sinks in the presence of known obstacles [18]-[21] in the WSN area. Generally, the scheduling and the mobility of mobile sinks are deterministically controlled to suit the application. The mobile sinks increase the life of static sensors by reducing the transmission load of sensors by reducing the intermediate relaying task of them.

2. STEM MODEL, SYMBOLS AND DEFINITIONS

2.1. Moving sink paths

In our scheme, we have two managed service providers (MSPs) covering the sensor node grid points of the entire WSN. The two paths are designated as MSP1 and MSP2. A path is made up of a sequence of non-repeating edges. In the present scheme only straight and diagonal edges are allowed. As an example, two paths MSP1 and MSP2 are shown in Figure 1. Here, MSP1 and its edges are marked in red while those of MSP2 are marked in blue.

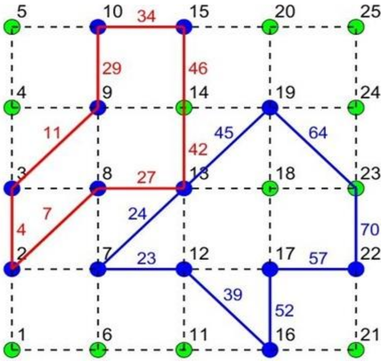


Figure 1. Two moving sink paths MSP1 and MSP2

Let the set of edges forming MSP1 and MSP2 be represented by MSP1_E and MSP2_E, respectively. The individual edges of MSP1_E and MSP2_E are represented as (1) and (2).

$$MSP1\_E = [msp1\_e(1), msp1\_e(2), \ldots., msp1\_e(k), \ldots., msp1\_e(L_1)] \quad (1)$$

$$MSP2\_E = [msp2\_e(1), msp2\_e(2), \ldots., msp2\_e(k), \ldots., msp2\_e(L_2)] \quad (2)$$

Here L_1 and L_2 are the sizes of MSP1_E and MSP2_E, respectively. Individual elements $msp1\_e(k)'s$ and $msp2\_e(k)'s$ are the edges of MSP1_E and MSP2_E, respectively. In the graph context, L1 and L2 are the number of edges in MSP1_E and MSP2_E, respectively. MSP1_E and MSP2_E are the non-overlapping subsets of the edge set E of the graph.

2.2. Index format representation of MSP1_E and MSP2_E

A subset can be represented in index format in terms of the full set. Index format of a subset is a binary vector of length that is equivalent to the size of the full set. Let the binary vector x_1 represent the index format of MSP1_E. Then the size of x_1 is M which is the size of E. The binary vector x_1 is formed as follows. The j^{th} element of x_1 is set to 1, if $e(j)$ of E belongs to MSP1_E, else it is set to zero. That is,

$$x_1(j) = \begin{cases} 1, & \text{if edge } j \in MSP1\_E \\ 0, & \text{otherwise} \end{cases} \quad (3)$$

for $j = 1$ to M. From (3), it can be seen that $sum(x_1) = L_1$. Another way of representing x_1 is (4).

$$\left. \begin{array}{l} x_1\big(msp1\_e(k)\big) = 1 \ \text{ for } \ k = 1 \text{ to } L_1 \\ x_1(other\ elements) = 0 \end{array} \right\} \quad (4)$$

Similarly, the index format of MSP2_E represented by x_2 is a binary vector of length M whose j^{th} element is given by (5).

$$x_2(j) = \begin{cases} 1, & \text{if edge } j \in MSP2\_E \\ 0, \text{otherwise} \end{cases} \quad (5)$$

for $j = 1$ to M. Here, $sum(x_1) = L_2$. Another way of representing x_2 is (6).

$$\left. \begin{array}{l} x_2\big(msp2\_e(k)\big) = 1 \ \text{ for } \ k = 1 \ \text{ to } \ L_2 \\ x_2(other\ elements) = 0 \end{array} \right\} \quad (6)$$

2.3. Decision vectors x_1 and x_2

Binary vectors x_1 and x_2 are also the decision vectors in determining MSP1_E and MSP2_E using an optimization program. This means, once x_1 and x_2 are determined using say the binary integer program (BIP), then MSP1_E and MSP2_E are obtained based on (4) and (6). In fact, we can express MSP1_E and MSP2_E as (7).

$$MSP1\_E = find(x_1) \text{ and } MSP2\_E = find(x_2) \tag{7}$$

Here, find(x) is a MATLAB function that gives the index locations of 1's in vector x.

2.4. Moving sink paths as the sequence of nodes

The paths MSP1 and MSP2 can also be represented as the sequences of nodes. The corresponding set of nodes which form MSP1 and MSP2 are represented by sets MSP1_V and MSP2_V as (8) and (9).

$$MSP1\_V = [msp1\_v(1), msp1\_v(2), \ldots., msp1\_v(k), \ldots., msp1\_v(L_1)] \tag{8}$$

$$MSP2\_V = [msp2\_v(1), msp2\_v(2), \ldots., msp2\_v(k), \ldots., msp2\_v(L_2)] \tag{9}$$

Since, MSP1 and MSP2 are closed paths, for each path, the number of edges is equal to the number of vertices. Therefore, L_1 and L_2 are the number of vertices in MSP1 and MSP2. In (13) and (14), $msp1\_v(k)'s$ and $msp2\_v(k)'s$ are the vertices of MSP1_V and MSP2_V, respectively.

2.5. Index format representation of MSP1_V and MSP2_V

Index format of MSP1_V is represented by the binary vectors y_1 as (10).

$$y_1(j) = \begin{cases} 1, & \text{if edge } i \ MSP1\_V \\ 0, & \text{otherwise} \end{cases} \tag{10}$$

for $i = 1$ to N. The length of $y_1 = N = |V|$ where V is the vertex (node) set of the graph. From (10), it can be seen that $sum(y_1) = L_1$. Another way of representing y_1 is (11).

$$\left. \begin{array}{l} y_1\big(msp1\_v(k)\big) = 1 \text{ for } k = 1 \text{ to } L_1 \\ y_1(other\ elements) = 0 \end{array} \right\} \tag{11}$$

Similarly, the index format of MSP2_V is represented by the binary vector y_2 as (12):

$$y_2(i) = \begin{cases} 1, & if\ node\ i \in MSP2\_v \\ 0, \text{otherwise} \end{cases} \tag{12}$$

for $i = 1$ to N. The length of $y_2 = N = |V|$. From (12), it can be seen that $sum(y_2) = L_2$. Another way of representing y_2 is (13).

$$\left. \begin{array}{l} y_2\big(msp2\_v(k)\big) = 1 \ \text{ for } \ k = 1 \ \text{ to } \ L_2 \\ y_2(other\ elements) = 0 \end{array} \right\} \tag{13}$$

2.6. Decision vectors y_1 and y_2

Binary vectors y_1 and y_2 are also the decision vectors in determining MSP1_V and MSP2_V when an optimization program is used to determine them. This means, once y_1 and y_2 are determined using say the BIP, then MSP1_V and MSP2_V are obtained using the MATLAB find(…) function as (14).

$$MSP1\_V = find(y_1) \ \ andMSP2\_V = find(y_2) \tag{14}$$

2.7. Participating nodes and participating node set

A node (vertex) that belongs to MSP1_V or MSP2_V is defined as a participating node. That is, a participating node $\in \{MSP1\_V\} \cup \{MSP2\_V\}$. The participating node set is the collection of all the participating nodes. Participating node set, represented by PN, is given by the union of MSP1_V and MSP2_V as:

$$PN = \{MSP1\_V\} \cup \{MSP2\_V\}$$

2.8. Sensor node vertex set

The sensor nodes of the network are represented by the set S. All the nodes (grid points) need not be sensor nodes. Paths MSP1 and MSP2 should fully cover S. Thus S is a subset of PN which in turn is a subset of V.

2.9. Common node

The WSN has a single common node or a junction node denoted by cn, It is normally a node at the centroid of the WSN or very near to it. The grid point location of the cn is the designer's choice. cn may or may not belong to S. In the example of Figure 1, node 13 is the cn and it belongs to S. The paths MSP1 and MSP2 should compulsorily visit the nodes of S including cn.

2.10. Participating edges and participating edge set

An edge that belongs to MSP1_E or MSP2_E is defined as a participating edge. That is, edge j is a participating edge if $j \in MSP1\_E$ or $j \in MSP2\_E$. Collection of all the participating edges of the forms the participating edge set denoted by PE. Then, PE is given by (15).

$$PE = \{MSP1\_E\} \cup \{MSP2\_E\} \tag{15}$$

From (3) and (5), we know that $x_1(j) = 1$ if $j \in MSP1\_E$ and $x_2(j) = 1$ if $j \in MSP2\_E$. Therefore, j is a participating edge if $x_1(j) = 1$ or $x_2(j) = 1$. This fact can be represented as (16).

$$edge\ j \in PE \text{ if } x_1(j) + x_2(j) = 1 \tag{16}$$

2.11. Participating degree of a vertex

Consider a vertex $v(i)$ of the MSP graph. The number of edges connected to that vertex is defined as the conventional degree of that vertex. Conventional degree of vertex i denoted by $d(i)$ is given by:

$$d(i) = \text{Number of edges connected to node } i$$

All the edges of node i may not participate in forming the MSP. We define the participating degree of a node (vertex) i , represented by $pd(i)$ as (17).

$$pd(i) = \text{Number of participating edges connected to node } i \tag{17}$$

Based on this definition, in Figure 1, $pd(3) = 2$ because only two edges (3, 2) and (3, 9) of node 3 participate in MSP1. Also, in Figure 1, $pd(1) = 0$ because vertex 1 does not host any participating edge.

2.12. Participating edge set of vertices

Those edges of the graph, connected to vertex i form the conventional edge set of vertex i and it is denoted by $CE(i)$. It is defined as (18).

$$CE(i) = \{\text{Those edges connected to node } i\} \tag{18}$$

The participating edges which pass through node i form the participating edge set of that node. Thus $PE(i)$ which denotes the participating edge set of node i is defined as:

$$PE(i) = \{\text{Those participating edges connected to node } i\}$$

To determine $pd(i)$for a given i , we have to count those edges which are connected to vertex i and which are also participating edges (those who belong to PE). Obviously, these edges are the members of $PE(i)$. An edge j belongs i if $vei(i,j) = 1$ (property of VEI) and also belongs PE if $X_1(j) + x_2(j) = 1$ (see (16)). Therefore,

$$j \in PE(i) \text{ if } vei(i,j) = 1 \text{ and } \left(x_1(j) + x_2(j)\right) = 1 \tag{19}$$

Since both $vei(i,j)$ and $x_1(j) + x_2(j)$ are binary variables, and operation can be replaced by multiplication operation. Therefore, (19) can be rewritten as (20).

$$j \in PE(i) if vei(i,j) * \left(x_1(j) + x_2(j)\right) = 1 \tag{20}$$

for $j = 1$ to M. Now, for $i = 1$ to N, consider the summation,

$$L(i) = \sum_{j=1}^{M} vei(i,j) * \left(x_1(j) + x_1(j)\right) \tag{21}$$

Here, $vei(i,j) * \left(x_1(j) + x_2(j)\right)$ is a binary variable which can take the value 1 or 0. Therefore (21) implies that $vei(i,j) * \left(x_1(j) + x_2(j)\right)$ is 1 for $L(i)$ distinct values of j. This means, from (20), that, $j \in PE(i)$ is satisfied for $L(i)$ distinct values of j out of M. Therefore, the number of elements in $PE(i)$ is $L(i)$. Therefore,

$$pd(i) = |PE(i)| = L(i) \tag{22}$$

From (21) and (22), we get,

$$pd(i) = \sum_{j=1}^{M} vei(i,j) * \left(x_1(j) + x_1(j)\right) \tag{23}$$

3. DUAL MOVING SINK PATH DISCOVERY PROBLEM

Our objective is to find MSP1 and MSP2 satisfying the following constraints.

– Both MSP1 and MSP2 should visit the common node cn.
– There should be single common node.
– MSP1 and MSP2 each should form round trip paths visiting every sensor node exactly once in each trip, travelling along the edges of the graph.
– MSP1 and MSP2 are to be edge disjoint.
– MSP1 and MSP2 should not cross each other. If the paths cross, there is a possibility of the physical collision of the moving sink vehicles.
– MSP1 and MSP2 should not cross themselves. Then the length of that path will be unnecessarily longer compared to the non- crossing case.
– MSP1 and MSP2 are to be node disjoint except for a common node.
– MSP1 and MSP2 should not have any sub-tours [22].
– Total length of the trip by MSP1 and MSP2 should be minimum.
– Length (MSP1) should be equal or very nearly equal to length (MSP2).
– The intermediate non-sensor vertices should not repeat along the paths travelled either by MSP1 or MSP2.

3.1. Proposed solution

The method proposed to solve the DMSP is similar to solving TSP [23] with two salespeople. As in TSP [22], here also we use binary integer programming [24]. Our major contribution is to express the constraints in the algebraic form in terms of the 4 decision vectors x1, x2, y1 and y2.

4. FORMULATION OF CONSTRAINS

The constraints of section 3 are expressed in proper algebraic formats suitable for the binary integer programming solver. We assume that MSP1 and MSP2 are the optimum paths satisfying the constraints of section 3.

4.1. Constraints on common node

Consider the common node cn through which MSP1 and MSP2 both enter once and leave once. Thus the $pd(i) = 4$ if i is the common node. (See node 13 in Figure 1) Now, consider other participating nodes. Since paths MSP1 and MSP2 are closed paths without any sub loops are tree segments, a path has to enter a non-common participating node once and leave also once. Therefore, the participating degree $pd(i)$ has to be 2 if node i belongs $\{PN - cn\}$ (Set difference). MSP1 and MSP2 do not visit non-participating nodes. Therefore, there is neither an entry nor an exit for these nodes. Therefore, $pd(i) = 0$ for $i's$ which do not belong to PN. These participating degree values are surmised as:

- Case 1) Node i is the common node. Then, $pd(i) = 4$ for $i = cn$.
- Case 2) Node i belongs to PN except the common node. Then, $pd(i) = 2$ for $i \in \{PN - cn\}$.
- Case 3) Node i does not belong to PN. Then, $pd(i) = 0$ for $i \notin PN$.

Using (23) to represent $pd(i)$, the above three cases equations can be expressed as:

$$\sum_{j=1}^{M} vei(i,j) * \left(x_1(j) + x_2(j)\right) = 4 \text{ for } i = cn \tag{24}$$

$$\sum_{j=1}^{M} vei(i,j) * \left(x_1(j) + x_2(j)\right) = 2 \text{ for } \forall i \notin \{PN - cn\} \tag{25}$$

$$\sum_{j=1}^{M} vei(i,j) * \left(x_1(j) + x_2(j)\right) = 0 \text{ for } \forall i \notin PN \tag{26}$$

Now, consider the common node cn. It belongs to both MSP1_V and MSP2_V. Therefore, from (10), $y_1(cn) = 1$ and from (12), $y_2(cn) = 1$. Therefore,

$$y_1(i) + y_2(i) = 2 \text{ for } i = cn \tag{27}$$

Now, consider node i which belong to $\{MSP1\_V - cn\}$. Then from (10), $y_1(i) = 1$ but from (12), $y_2(i) = 0$ because now $i \notin \{MSP2\_V\}$ because, $\{MSP1\_V - cn\}$ and $\{MSP2\_V\}$ are disjoint. Therefore $y_1(i) + y_2(i) = 1$ for $i \in \{MSP1\_V - cn\}$.

Similarly, it can be shown that $y_1(i) = 0$ but $y_2(i) = 1$ and $y1(i) + y_2(i) = 1$ when $i \in \{MSP2\_V - cn\}$. Thus, $y_1(i) + y_2(i) = 1$ when $i \in \big\{\{MSP1\_V - cn\} \cup \{MSP2\_V - cn\}\big\}$.

Since, $i \in \big\{\{MSP1\_V - cn\} \cup \{MSP2\_V - cn\}\big\}$ is equivalent to $\{PN - cn\}$,

$$y_1(i) + y_2(i) = 1 \text{ for } \forall i \in \{PN - cn\} \tag{28}$$

for $i \notin PN$ from (10) and (12), $y_1(i) = 0$ and $y_2(i) = 0$. Therefore,

$$y_1(i) + y_2(i) = 0 \text{ for } \forall i \notin PN \tag{29}$$

Comparing the RHS of (24), (25) and (26), with (27), (28) and (29), it is clear that (24), (25) and (26) can be replaced by a single equation as (30).

$$\textstyle\sum_{j=1}^{M} vei(i,j) * \big(x_1(j) + x_2(j)\big) = 2 * \big(y_1(i) + y_2(j)\big) \text{ for } \forall i \in V \tag{30}$$

4.2. Constraints regarding the common node

Common node cn is selected by the designer and is known a priori. cn belongs to MSP1_V and MSP2_V. This constraint is expressed as (31) and (32).

$$y_1(cn) = 1 \tag{31}$$

$$y_2(cn) = 1 \tag{32}$$

4.3. Constraints on sensor nodes

The moving sink path, either MSP1 or MSP2 has to visit the sensor nodes represented by the set S. Therefore the $pd(i)$ of these nodes has to be exactly two for non-common nodes and four for the common node. (For other nodes it can be two or zero.) This constraint is expressed as (33) and (34).

$$\textstyle\sum_{j=1}^{M} vei(i,j) * \big(x_1(j) + x_2(j)\big) = 4 \text{ for } i = cn \tag{33}$$

$$\textstyle\sum_{j=1}^{M} vei(i,j) * \big(x_1(j) + x_2(j)\big) = 2 \text{ for } \forall i \in i\{s - cn\} \tag{34}$$

4.4. Non-crossing criterion

MSP1 and MSP2 should not cross each other or themselves. (They can meet at the common node). Since MSP1 and MSP2 are edge disjoint, the self-crossing of MSP1 and MSP2 can occur only along the diagonal edges of grid cells. When a crossing occurs in a grid cell, both the diagonal edges act as the participating edges. Let $j1$ and $j2$ be the two diagonal edges of a specific grid cell. Then, from (16):

$$j1 \in PEif x_1(j1) + x_2(j1) = 1$$
$$j2 \in PEif x_1(j2) + x_2(j2) = 1$$

If both $j1$ and $j2$ were to belong to PE, then both the above condition should be true. Then, $x_1(j1) + x_2(j1) + x_1(j2) + x_2(j2) = 2$. To prevent this event and to avoid crossing,

$$x_1(j1) + x_2(j1) + x_1(j2) + x_2(j2) < 2 \tag{35}$$

Constraint (35) avoids diagonal crossing in a specific grid cell. To extend this to all grid cells, we apply constraint (35) to all the grid cells in the graph. Let the grid cells of the graph be denoted as $C = [c(1), c(2), \ldots., c(u), \ldots., c(T)]$ where T is the total number of cells.

Let $c(u).j1$ and $c(u).j2$ be the two diagonal edges of cell $c(u)$ for $u = 1$ to T. Then the extension of (35) for all the cells can be expressed as (36).

$$x_1(c(u).j1) + x_2(c(u).j1) + x_1(c(u).j2) + x_2(c(u).j2) < 2 \tag{36}$$

4.5. Objective function to be minimized

The objective function to be minimized in DMSP is the total length of the moving sink paths MSP1 and MSP2. In our scheme, the horizontal and vertical (HV) edge lengths are normalized to 1 and hence the length of diagonal edges are $\sqrt{2}$. Therefore, the total length of MSP1 and MSP2 represented by F is given by (37).

$$\begin{aligned} F &= 1 * \text{sum}(HV \text{ edges of } x_1) + \sqrt{2}\text{sum}(\text{diagonal edges of } x_1) \\ &+1 * sum(HV\ edge\ of\ x_2) + \sqrt{2}\ (diagonal\ edges\ of\ x_2) \end{aligned} \tag{37}$$

F is a linear function of the decision variables and it is the objective function to be minimized.

5. BINARY INTEGER PROGRAM FORMULATION AND SOLUTION FOR DMSP

The DMSP is solved using BIP that solves for x_1, x_2, y_1 and y_2 to minimize F subjected to the constraints, (30) to (34) and (36). All the constraints are linear and from (37), it can be seen that the objective function F is also linear [25]. BIP is applied iteratively until all the sub-tours are eliminated [23]. BIP gives the final optimum values of x_1, x_2, y_1 and y_2. Then the optimum paths MSP1 and MSP2 are obtained in terms of edges as given by (4), (6) and in terms of nodes as given by (11), (13). DMSP algorithm is discussed in the section 5.1.

5.1. DMSP algorithm

Inputs: Grid based WSN with known sensor node locations. Common node (cn), as selected by the designer. In general, cn is selected to be at the center of the WSN area.

Output: Optimal paths MSP1 and MSP2.

- Identify N nodes and M edges of the grid graph.
- Initially set the decision vectors x_1, x_2 and y_1 and y_2 as all zero vectors of length M and N, respectively.
- Minimize the total length F as given by (37), subjected to constraints (30), (31), (32), (33), (34) and (36) using BIP.
- Repeat step 4 until all subtours are eliminated.
- Get the final optimum values of x_1, x_2, y_1 and y_2.
- Get MSP1 and MSP2 in terms of edges and nodes using (4), (6) and (11), (13).
- Over.

Example 1

Here, $M = 8$, $N = 6$, and the common node $cn = 22$.

$$S = [22\ \ 39\ \ 10\ \ 23\ \ 42\ \ 16\ \ 46\ \ 18\ \ 9\ \ 44\ \ 41\ \ 12\ \ 7\ \ 2\ \ 38\ \ 8\ \ 25\ \ 20\ \ 34\ \ 17\ \ 15\ \ 3]$$

After solving the BIP for this problem, the resulting paths MSP1 (in red) and MSP2 (in blue) are shown in Figure 2. Here, $MSP1\_V = [2\ \ 3\ \ 10\ \ 11\ \ 12\ \ 18\ \ 17\ \ 23\ \ 22\ \ 16\ \ 9\ \ 8\ \ 7]$ and $L_1 = 13$. In Figure 2, $MSP2\_V = [15\ \ 22\ \ 28\ \ 34\ \ 41\ \ 42\ \ 47\ \ 39\ \ 44\ \ 38\ \ 32\ \ 28\ \ 20]$ and $L_2 = 14$. From MSP1_V and MSP2_V, the corresponding path lengths can be obtained based on (37).

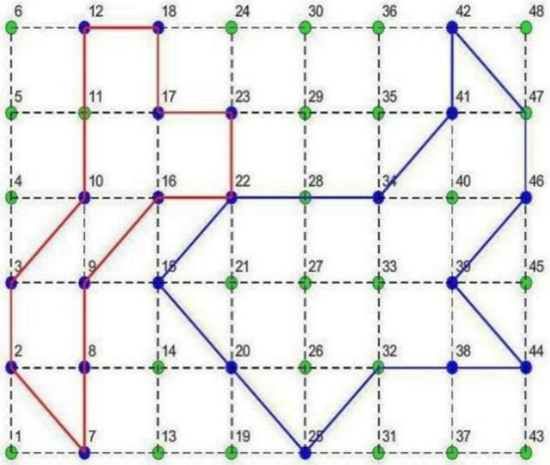
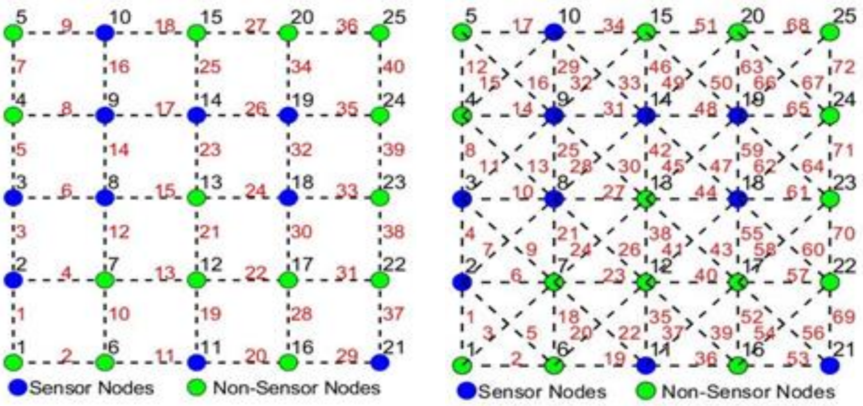


Figure 2. Minimum length dual moving sink paths

6. EXPERIMENTAL RESULTS AND DISCUSSION

Consider the following two modes of the moving sink unit. Mode 1: The moving sinks have only horizontal and vertical movements along the grid nodes (see Figure 3(a)). Mode 2: The moving sinks have diagonal movement capability in addition to horizontal and vertical movements (see Figure 3(b)). MSDN can be applied to both the modes. In mode 1, diagonal edges are ignored while applying MSDN. The total length of the optimal travel paths which cover the sensor nodes of the given WSN area, in mode 2, is shorter compared to that of mode 1. Experiment 1 compares the optimal round trip travel distances of these two modes.



(a) (b)

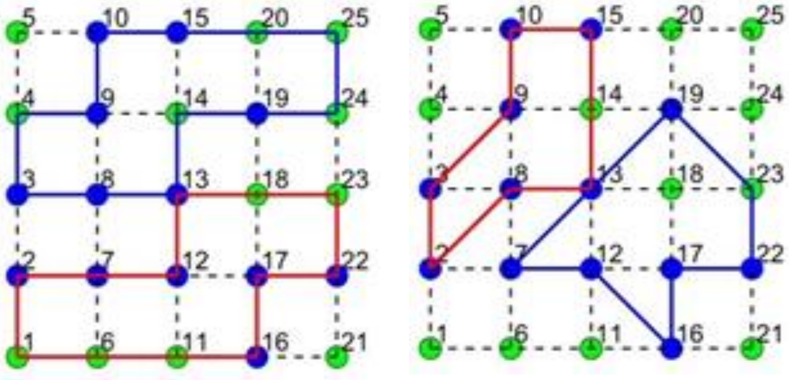
Figure 3. These figures are; (a) 4-connected grid layout and (b) 8-connected grid layout

Experiment 1

Here, we consider 3 WSNs with progressively increasing sizes. The number of sensor nodes is taken as round (45% of number of grid points) in each WSN. The sensor nodes are distributed randomly among the grid points. The optimal lengths of MSP1 and MSP2 in both the modes in all the 3 WSNs are given in Table 1. MSP1 and MSP2 are determined for two modes where, 4-connectivity is mode 1 and 8-connectivity is mode 2. The optimal paths obtained using DMSP are shown in Figures 4 to 6 for all the 3 WSNs. From Table 1, we see that the optimal length of paths in mode 2 is shorter compared to those of mode 1.

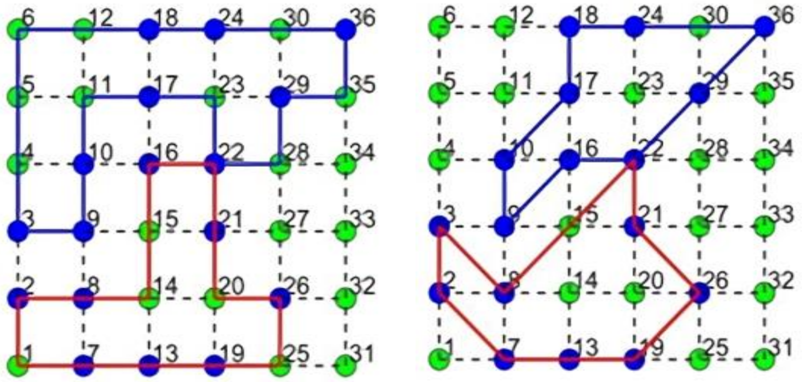
Table 1. Optimal length of MSP1 and MSP2 in two modes for 1 the 5 WSN’s

Grid size of WSN	Length of MSP1		Length of MSP2		Length of MSP2	
	Mode 1	Mode 2	Mode 1	Mode 2	Mode 1	Mode 2
5 × 5	12.0000	9.6560	12.0000	8.8280	24.0000	18.4840
6 × 6	18.0000	11.6560	14.0000	12.4840	32.0000	24.1400
7 × 7	24.0000	20.4840	14.0000	12.8280	38.0000	33.3120



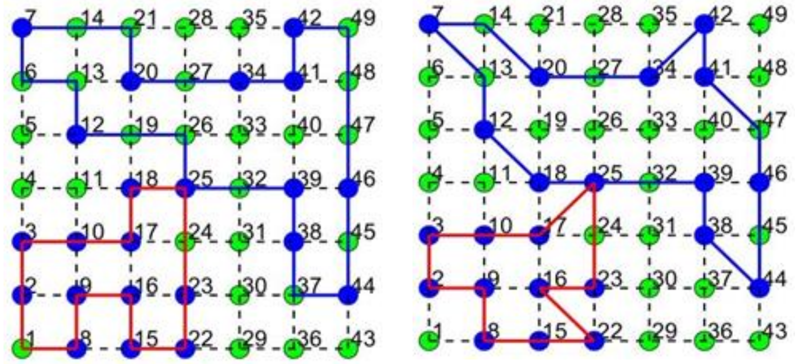
(a) (b)

Figure 4. Grid size 5×5; (a) mode 1 and (b) mode 2



(a) (b)

Figure 5. Grid size 6×6; (a) mode 1 and (b) mode 2



(a) (b)

Figure 6. Grid size 7 × 7; (a) mode 1 and (b) mode 2

7. CONCLUSION

A new technique is presented to determine the optimal paths for two moving sinks with a common node between the two paths. The paths can be along the horizontal/vertical edges and/or immediate diagonal edges based on the optimality criterion. The paths are so determined that the two paths do not cross each other or overlap except at the common node. The linear integer program is solved using 4 decision vectors which make the specifications of constraints easy and flexible. In this method, both the path lengths are made equal or very nearly equal while the total length is minimized. This integrated approach is a novel and unique solution to solve the dual moving sink path problem in a WSN.

REFERENCES

[1] C. Tunca, S. Isik, M. Y. Donmez and C. Ersoy, "Distributed Mobile Sink Routing for Wireless Sensor Networks: A Survey," *IEEE Communications Surveys & Tutorials*, vol. 16, no. 2, pp. 877-897, Second Quarter 2014, doi: 10.1109/SURV.2013.100113.00293.

[2] M. Di Francesco, S. K. Das, and G. Anastasi, “Data collection in wireless sensor networks with mobile elements: A survey,” *ACM Trans. Sensor Networks*, vol. 8, no. 1, pp. 1–31, 2011.

[3] H. Huang, A. V. Savkin, M. Ding, and C. Huang, “Mobile robots in wireless sensor networks: A survey on tasks,” *Computer Networks*, vol. 148, pp. 1–19, 2019, doi: 10.1016/J.COMNET.2018.10.018.

[4] W. Liang, J. Luo, and X. Xu, "Prolonging Network Lifetime via a Controlled Mobile Sink in Wireless Sensor Networks," in *2010 IEEE Global Telecommunications Conference GLOBECOM 2010*, 2010, pp. 1-6, doi: 10.1109/GLOCOM.2010.5683095.

[5] Z. Wang, S. Basagni, E. Melachrinoudis, and C. Petrioli, “Exploiting Sink Mobility for Maximizing Sensor Networks Lifetime,” *Proceedings of the 38th Annual Hawaii International Conference on System Sciences*, doi: 10.1109/hicss.2005.259.

[6] H. Huang, “Performance Improvement by Introducing Mobility in Wireless Communication Networks,” *arXiv: 1712.02436*, 2017.

[7] S. M. A. Akber, *et al*., “Data Volume Based Data Gathering in WSNs using Mobile Data Collector,” in *Proceedings of the 22nd International Database Engineering & Applications Symposium on - IDEAS 2018*, 2018, doi: 10.1145/3216122.3216166.

[8] S. Basagni, A. Carosi, C. Petrioli, “Mobility in wireless sensor networks,” in: *Algorithms and Protocols for Ad Hoc and Sensor Networks*, John Wiley & Sons, Inc., 2007.

[9] M. Koç and I. Korpeoglu, “Coordinated movement of multiple mobile sinks in a wireless sensor network for improved lifetime,” in *EURASIP Journal on Wireless Communications and Networking*, vol. 2015, no. 1, 2015, doi: 10.1186/s13638-015-0472-5.

[10] U. Hariharan, “Multi Sink Scheduling Scheme for Wireless Sensor Networks,” *International Journal of Advanced and Applied Sciences*, vol. 3, no. 2, pp. 1-8, 2014.

[11] H. A. Al-Behadili, S. K. Alwane, Y. I. Al-Yasir, N. O. Parchin, P. Olley, and R. A. Abd-Alhameed, “Use of multiple mobile sinks in wireless sensor networks for large-scale areas,” *IET Wireless Sensor Systems*, vol. 10, no. 4, pp. 175–180, 2020, doi: 10.1049/iet-wss. 2019.0208.

[12] P. Zhong and F. Ruan, “An energy efficient multiple mobile sinks based routing algorithm for wireless sensor networks,” *IOP Conference Series: Materials Science and Engineering*, vol. 323, p. 012029, 2018, doi: 10.1088/1757-899x/323/1/012029.

[13] W. Wen, C.-Y. Chang, S. Zhao, and C. Shang, “Cooperative Data Collection Mechanism Using Multiple Mobile Sinks in Wireless Sensor Networks,” *Sensors*, vol. 18, no. 8, p. 2627, 2018, doi: 10.3390/s18082627.

[14] S. Liu, Z. Wei, Z. Guo, X. Yuan, and Z. Feng, "Performance Analysis of UAVs Assisted Data Collection in Wireless Sensor Network," in *2018 IEEE 87th Vehicular Technology Conference (VTC Spring)*, 2018, pp. 1-5, doi: 10.1109/VTCSpring.2018.8417673.

[15] A. Mazayev, N. Correia, and G. Schütz, “Data Gathering in Wireless Sensor Networks Using Unmanned Aerial Vehicles,” *International Journal of Wireless Information Networks*, vol. 23, no. 4, pp. 297–309, 2016, doi: 10.1007/s10776-016-0319-y.

[16] C. Luo, M. N. Satpute, D. Li, Y. Wang, W. Chen, and W. Wu, "Fine-Grained Trajectory Optimization of Multiple UAVs for Efficient Data Gathering from WSNs," *IEEE/ACM Transactions on Networking*, vol. 29, no. 1, pp. 162-175, Feb. 2021, doi: 10.1109/TNET.2020.3027555.

[17] O. Bouhamed, H. Ghazzai, H. Besbes and Y. Massoud, "A UAV-Assisted Data Collection for Wireless Sensor Networks: Autonomous Navigation and Scheduling," in *IEEE Access*, vol. 8, pp. 110446-110460, 2020, doi: 10.1109/ACCESS.2020.3002538.

[18] K. Clarkson, S. Kapoor, and P. Vaidya, “Rectilinear shortest paths through polygonal obstacles in O(n(logn)2) time,” *Proceedings of the third annual symposium on Computational geometry - SCG 87*, 1987, doi: 10.1145/41958.41985.

[19] R. Kumar, T. Amgoth and D. Das, "Obstacle-Aware Connectivity Establishment in Wireless Sensor Networks," *IEEE Sensors Journal*, vol. 21, no. 4, pp. 5543-5552, 15 Feb.15, 2021, doi: 10.1109/JSEN.2020.3032144.

[20] G. Xie, K. Ota, M. Dong, F. Pan, and A. Liu, “Energy-efficient routing for mobile data collectors in wireless sensor networks with obstacles,” *Peer-to-Peer Networking and Applications*, vol. 10, no. 3, pp. 472–483, 2016, doi: 10.1007/s12083-016-0529-1.

[21] G. Xie and F. Pan, "Cluster-Based Routing for the Mobile Sink in Wireless Sensor Networks with Obstacles," *IEEE Access*, vol. 4, pp. 2019-2028, 2016, doi: 10.1109/ACCESS.2016.2558196.

[22] M. Diaby, “The Travelling Salesman Problem: A Linear Programming Formulation,” *WSEAS Transactions on Mathematics*, vol. 6, no. 6, pp. 745-754, 2007.

[23] Y. Shuai, S. Yunfeng, and Z. Kai, “An effective method for solving multiple travelling salesman problem based on NSGA-II,” *Systems Science & Control Engineering*, vol. 7, no. 2, pp. 108–116, 2019, doi: 10.1080/21642583.2019.1674220.

[24] G. Pataki, “Teaching Integer Programming Formulations Using the Traveling Salesman Problem,” *SIAM Review*, vol. 45, no. 1, pp. 116–123, 2003, doi: 10.1137/s00361445023685.

[25] V. G. Deineko, B. Klinz, A. Tiskin, and G. J. Woeginger, “Four-point conditions for the TSP: The complete complexity classification,” *Discrete Optimization*, vol. 14, no. C, pp 147–159, 2014, doi: 10.1016/j.disopt.2014.09.003.

BIOGRAPHIES OF AUTHORS

Dr. Satish Tunga received his Ph.D. in Electronics Engineering from Jain University, Bangalore in 2018. He did his B.E. and M.E. in Electronics, in 1984 and 1991, respectively, from University Visvesvaraya College of Engineering, Bangalore. He is presently working as associate professor in Department of Electronics and Telecommunication Engineering, M S Ramaiah Institute of Technology, Bangalore. He has published more than 10 papers in various international conferences and journals. His areas of interests are image processing, signal processing, communication systems, antennas, and electronic circuits.
Email: satish.tunga@msrit.edu

Prof. Sadashiva V. Chakrasali is currently working as an assistant professor in E & C Department, M S Ramaiah Institute of Technology, Bangalore. His research interests are statistical signal processing and communication. He has completed B.E. and M.Tech. from Kuvempu University and VTU Belgaum in the year 2000 and 2004, respectively.
Email: sadashiva.c@msrit.edu

Dr. Shylashree N. is currently working as Associate Professor in the Department of Electronics and Communication Engineering at RV College of Engineering, Bengaluru. She is having 15 years of teaching experience. She was a recipient of the best PhD thesis award for the year 2016-2017 in Electronics and Communication Engineering from BITES. She has received the best IEEE researcher award in IEEE-AGM meeting held during 2021 from Bangalore IEEE section. She has also received the best paper award in IEEE-ICERECT held during 2015 at Mandya. She has research publication in 18 International Journals (out of which 5 journals are Springer-SCI journals), 4 Springer book chapters and 9 International conferences. She has published 2 patents and filed 1 German patent in the area of cryptography. She has also published 2 patents in the area of VLSI out of which one of them got a grant. She is also the author of the Network Theory, Engineering Statistics and Linear Algebra textbook and Control Engineering textbook. Her areas of interest include Network Analysis, Analysis and Design of Digital Circuits, Digital VLSI Design, Analog & Mixed mode VLSI design, Low power VLSI Design, Cryptography & Network security, Statistics and Linear algebra and Control Engineering.
E-mail:shylashashi@gmail.com; shylashreen@rvce.edu.in

Prof. Latha B. N. is currently working as Assistant Professor in the Department of Electronics and Communication Engineering at JSS academy of technical education, Bangalore. She has 30 years of teaching experience Her areas of interest are signal processing, power electronics, computer networks, HDL and control engineering. She has completed BE and M.Tech from Mysore university and NITK, Surthkal, Mangalore university in the year 1990 and 1997 respectively.
Email: lathabn@jssateb.ac.in; lathajss@gmail.com

Dr. Mamatha A. S. is currently working as Associate Professor in the Department of Electronics and Communication Engineering at St. Joseph Engineering College, Mangaluru. She has 22 years of teaching experience. She is the author of four international journals and six international conferences in the field of multispectral image compression. She is the author of the network theory, engineering statistics & linear algebra and control engineering textbook. Her areas of interest are signal processing, image compression, and control engineering. She is a senior IEEE member.
E-mail: mamathag@sjec.ac.in; mamathatanya@gmail.com