An Optimization-based Approach for Connecting Partitioned Mobile Sensor/Actuator Networks

Mustafa Y. Sir*, Izzet F. Senturk†, Esra Sisikoglu* and Kemal Akkaya†
*Department of Industrial and Manufacturing Systems Engineering, University of Missouri, Columbia, MO 65211
Email: {sir, sisikoglu}@missouri.edu
†Department of Computer Science, Southern Illinois University Carbondale, Carbondale, IL 62901
Email: {isenturk, kemal}@cs.siu.edu

***Abstract*—Wireless Sensor and Actuator Networks (WSANs) employ mobile nodes in addition to stationary tiny sensors. Similarly, mobile sensors make it possible to have the flexibility of mobility in mobile sensor network (MSN) applications. Mobility can be exploited to connect partitioned WSANs and MSNs due to large scale damages or deployment problems. However, since mobility consume significant energy and it can be limited due to terrain constraints, the travel distance for the mobile nodes should be minimized in such a recovery effort. In this paper, we present a mathematical model which minimizes the total travel distance for connecting a given number of partitions. The idea is based on network flows and the problem is modeled as a mixed integer nonlinear program. The nonlinear terms in the model are linearized using a polygon approximation for computational efficiency. We evaluated the performance of the proposed approach in terms of total distance as well as the time to reconnect the partitions. The results show that our approach outperforms the heuristic approach in terms of total distance and delay and reveals various trade-offs involved in connecting multiple partitions.**

I. INTRODUCTION

Wireless Sensor Networks (WSNs) have been of interest for numerous applications in the civil and military domains. In such networks, typically nodes are fixed and the gathered information was sent to a sink node which might be more powerful. Recently, WSNs have been enriched with the introduction of mobile nodes. These mobile nodes include sensors [1][2], robots [3], actuators [4], unmanned vehicles, etc. This gave rise even for more applications and new type of wireless networks which are referred to as Wireless Sensor and Actor (actuator) networks [4] or Mobile Sensor Networks [5][6]. Applications of WSANs include urban search and rescue, cyber-physical systems, forest fire fighting, destruction of mines in different environments. MSNs, on the other hand, can be used in applications with harsh environments such as environmental and habitat monitoring.

One of the main advantages of MSNs and WSANs is the ability to exploit node mobility for various purposes. Several performance metrics including dependability, connectivity, deployment convenience, coverage, energy, and accuracy can be improved by moving and relocating various nodes in these networks [7]. Our focus in this paper is connectivity which is crucial in WSANs and MSNs. In WSANs, the connectivity of actuators is needed to organize collaborative tasks. In MSNs such connectivity is solely for the purpose of relaying data to the base-station.

Mobility helps improving dependability of these networks in several ways. For instance, failed nodes can be replaced with the other spare nodes by moving the spare nodes to their locations. Similarly, if the network is partitioned, mobility can be exploited to restore connectivity by moving one or multiple nodes to the selected locations. Recently, the aforementioned dependability issues have been studied extensively in the context of WSANs and MSNs [8][9]. The main focus of these works was to deal with individual failures and restore the connectivity of the actuators or the connectivity of the WSN with the sink node.

However, WSANs or MSNs can experience massive node failures due to being employed in hostile environments. For instance due to a flood or fire, some nodes can be damaged creating two or multiple partitions in the network. A similar case occurs when the sensors are deployed randomly from a helicopter, creating multiple disjoint partitions in the region. In such cases, mobility can be exploited to restore connectivity assuming that the nodes are aware of the partitioning problem and thus the different partitions. However, there will be some cost associated with this approach in terms of energy consumption and delay. The movement of mobile nodes will incur energy overhead on battery-operated actuators and thus the travel distance for the nodes should be minimized. In addition, the time it takes to restore the connectivity should be minimized in order to resume the network operation as early as possible. This requires minimizing the maximum travel distance an actuator moves assuming that all the nodes will be moving independently and simultaneously.

In this paper, we investigate a centralized optimal solution for connecting multiple partitions in which the total travel distance of the nodes is minimized. Our approach is based on transportation network flow models. Specifically, we view a WSAN with n nodes as a transportation network where $n-1$ supplies at a designated actuator will be transported to the remaining actuators in the network (i.e., each will receive one supply at the end). If the network is connected, each actuator will be able to receive the supply. Otherwise, this will not be possible. We model the problem as a mixed integer nonlinear program. We use a polygon approximation to linearize the

---

This work is supported by the US National Science Foundation under the Award #: 1018404

nonlinear terms. We solve the mixed integer program using the default branch and bound algorithm implemented in the Gurobi 4.0 [10]. To the best of our knowledge the optimal solution for this problem has not been studied. We also investigate a second objective which minimizes the maximum travel distance of an actuator. This objective provides valuable information about time duration of the recovery problem. The results on several topologies with different number of actuators and partitions indicate that the optimal solution outperforms the heuristic solution in terms of both the total travel distance and delay.

This paper is organized as follows. Next section discusses the related work. Section III explains the assumptions and preliminaries. Section IV is dedicated to the explanation of the proposed approach. Validation is presented in Section V. Finally, the paper concludes in Section VI.

II. RELATED WORK

A. *Mobility in Wireless Networks*

Node mobility has been exploited in various wireless networks (e.g., WSNs, WSANs, MANETs) in order to improve performance metrics such as network lifetime, throughput, coverage and connectivity [7]. Two types of mobility were considered in these efforts: inherent and controlled mobility. Inherent mobility indicates natural movement of the nodes which changes the network topology and eventually affects both sensing coverage and communication connectivity. In controlled mobility on the other hand, nodes move only when needed and follow a prescribed travel path. For such a purpose either external mobile nodes are introduced into the system [11][12] or existing (internal) moveable nodes in the network are used [13][14][15][16][17].

B. *Mobility for Connectivity Restoration/Maintenance*

Since we deal with connecting a partitioned WSAN or MSN, our focus in this section will be on controlled mobility research regarding network connectivity. Depending on the application, this connectivity can be a major goal to be achieved or a constraint that needs to be maintained at all times.

1) Connectivity as a Goal: Several studies have been done in the past for specifically restoring connectivity due to single node failures. Two main methodologies can be identified in this category; block and sequential node movements. In the block movement, a set of connected nodes travel together as a unit. The idea is to re-link disjoint partitions by moving one partition towards another [17]. The second methodology works via relocating one or few nodes in a non-coordinated manner [8][9][18]. In these works, since the relocated nodes may get detached from their neighbors, a cascaded repositioning is pursued where neighbors follow through in order to sustain connectivity. [8] prefers the node that has fewer neighbors to keep the scope of the cascaded relocation limited whereas [9] solves this problem by utilizing the connected dominating set of the partition. Having reduced the energy consumption by engaging fewer nodes sharing the cost among them, shortest path between some nodes may be extended in cascading movement which influences the delay-sensitive applications adversely. This was addressed in [19] by identifying the smallest disjoint block to limit the relocation and reduce the recovery overhead.

Our work is different than the aforementioned works since they deal with connectivity problem due to single node failures that can be handled by the neighbors of the failed node locally. In our case, there may be multiple node failures which can create multiple partitions that cannot be handled locally. Therefore, these approaches cannot be applied directly. In addition, all of these approaches are distributed heuristics which cannot achieve the optimal solutions. One of our goals in this paper is to come up with an exact solution that can also be used as a benchmark for distributed heuristics in the future.

Multiple node failures have also been addressed in some works. For instance, in [9], multiple simultaneous but individual node failures were considered. This is different than our case since again the solution handles the failures individually using the techniques for single node cases. In [20], a special case of our problem was studied in a distributed manner considering two partitions. The idea is to first determine the connected dominating sets of the two partitions and then dominatee nodes are picked and moved to connect the involved partitions. However, the heuristic approach of this study cannot be extended to more than two partitions. We will compare our optimal solution with a variant of this heuristic for multiple partitions in Section V.

2) Connectivity as a Constraint: In several studies, controlled mobility is also used to improve the network performance while striving to preserve the network connectivity. For instance, in [13] nodes on a data route are repositioned in order to minimize the total transmission energy while still preserving the connectivity with the base-station. Meanwhile, the objective of C^2AP [14] is to maximize the coverage of actuator nodes while maintaining a connected inter-actuator topology. CORE [15] and COCOLA [16] deal with the effect of moving a node on the network connectivity. Basically, the goal is to avoid breaking any inter-node links. When a node moves, its neighbors follow it so that they stay reachable. CORE employs mutex to prevent race conditions when more than one node moves at the same time, while COCOLA implements a protocol that enables multiple nodes to move simultaneously without breaking communication links. Some authors [21] proposed using the optimal location information of each relay node as a guide for node movement while maintaining the connectivity of relay nodes along the data flow by combining two different schemes; (i) by avoiding the overreaction to the movements of its neighbors and (ii) by moving the nodes as close to their optimal positions as possible. Our work is different from these studies as our main goal is to achieve connectivity in the most efficient manner rather than requiring it just as a constraint.

III. PRELIMINARIES

A. *Assumptions*

We assume a WSAN application (i.e., urban search and rescue) where sensors and actuators are deployed together. The actuators will be placed in the area in order to collect data from sensors and take necessary actions based on the received data. The actuators are battery-operated but are assumed to be less-energy constrained and have larger transmission ranges than the sensors. They are assumed to have the ability to move and act on different areas on-demand basis. In addition, they are assumed to know their locations through mechanisms like GPS or other means [22]. The actuators form a network among themselves referred to as inter-actuator network which can be partitioned due to massive damage to the network or due to initial deployment. We note that we will use actuator and node interchangeably throughout the paper when we refer to the individual elements of WSANs.

A possible configuration is depicted in Fig. 1. In case of partition, connectivity among the command center and some of the actuator nodes may be lost. We assume that an Unmanned Aeiral Vehicle (UAV) may fly over the alive nodes and collect their information (i.e., locations). While this can provide connectivity among the actuators, this will be transient and thus a permanent connectivity is needed for collaboration and accordination among the actuators. For this work, we also assume an obstacle free environment where actuators can move to the desired location in a straight path. In the future, we plan to incorporate terrain information for path planning and consider any localization errors.

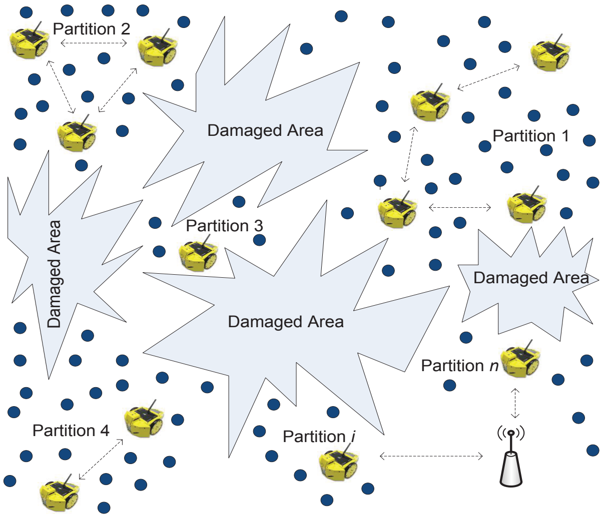


Fig. 1. A partitioned WSAN.

B. *Problem Definition*

Our problem can be defined formally as follows: Consider a WSAN with n identical actuators, a transmission range of r and multiple partitions, i.e., not all the actuators in the network can communicate with each other. Let $\mathcal{A} = \{1, 2, \dots, n\}$ be the set of actuators and $\mathbf{p}^i = \begin{bmatrix} l_1^i \\ l_2^i \end{bmatrix}$ be the location coordinates of actuator i, $i \in A$. Our problem is to move the actuators in the network so that the WSAN becomes connected (i.e., an actuator can communicate with all other actuators either directly or through other actuators) while minimizing the total travel distance of all actuators (i.e., $\sum_{i \in \mathcal{A}} d_i$) where d_i is the distance traveled by A_i." The problem is a typical multi-objective optimization problem [23].

Connectivity of the WSAN is defined as follows: Let $\mathbf{x}^i = \begin{bmatrix} x_1^i \\ x_2^i \end{bmatrix}$ be the position coordinates of actuator i, $i = 1, 2, \dots, n$, after it is relocated. Connectivity between actuator i and j are determined by the following variable:

$$y_{ij} = \begin{cases} 1, & \|\mathbf{x}^i - \mathbf{x}^j\| \le r, \\ 0, & \text{otherwise.} \end{cases}$$

Given the infinitely many number of locations where the actuators can be relocated, the problem is similar to Minimum Steiner Tree problem (MST) which connects k partitions using the least number of relay nodes. MST has been shown to be NP-Hard [24] and our problem is related to MST since once the relay nodes' locations are determined when finding MST, the appropriate actuators can be relocated to those locations afterwards and thus the network becomes connected again.

IV. FLOW-BASED MODEL

We model the connectivity problem using a flow-based approach. In other words, we view the wireless communication network between actuators as a transportation network, where a direct communication between two actuators is viewed as a bidirectional transportation link between those actuators. We assume that this "transportation" network can be used to transport items from one actuator to another. One of the actuators, chosen arbitrarily, assumed to have $n - 1$ items to supply (this actuator will be referred to as the "source" actuator) and that every other actuator requires exactly one item. Let f_{ij} be the amount of flow on the transportation link from actuator i to actuator j. If the flow model is feasible (*i.e.*, every actuator receives an item as a result of the flows) then the entire network is connected. Otherwise, the network must have partitions and therefore the items cannot be delivered to every actuator in the network. We consider two objectives: 1) Minimizing total distance traveled by all actuators; and 2) Minimizing maximum distance traveled by any actuator.

Letting M be a very large positive number and i^* be the index of the source actuator, the flow-based formulation for optimally relocating actuators to restore connectivity can be written as:

$$\min \sum_{i\in\mathcal{A}} \|\mathbf{x}^i - \mathbf{p}^i\| \quad (1)$$

$$\min \max_{i\in\mathcal{A}} \|\mathbf{x}^i - \mathbf{p}^i\| \quad (2)$$

subject to

$$\left\|\mathbf{x}^i - \mathbf{x}^j\right\| \leq r\ y_{ij} + M(1-y_{ij}),\ \forall i,j \in \mathcal{A},\ i \neq j, \quad (3)$$

$$\sum_{j\in\mathcal{A}\backslash\{i^*\}} f_{i^*j} = n-1, \quad (4)$$

$$\sum_{i\in\mathcal{A},i\neq j} f_{ij} = 1 + \sum_{k\in\mathcal{A},k\neq j} f_{jk}, \quad \forall j \in \mathcal{A}\backslash\{i^*\}, \quad (5)$$

$$f_{ij} \leq (n-1)y_{ij}, \quad \forall i,j \in \mathcal{A},\ i \neq j, \quad (6)$$

$$\mathbf{x}^i \in \mathbb{R}^2, \quad i \in \mathcal{A}, \quad (7)$$

$$y_{ij} \in \{0,1\}, \quad i,j \in \mathcal{A},\ i \neq j, \quad (8)$$

$$f_{ij} \geq 0, \quad i,j \in \mathcal{A},\ i \neq j. \quad (9)$$

As mentioned earlier objective functions (1) and (2) are to minimize total distance traveled by all actuators and maximum distance traveled by any actuator, respectively. Constraint (3) determines whether a pair of actuators communicates directly with each other based on the distance between their locations. In this set of constraints, if y_{ij} is 1 then the distance between the actuators i and j is forced to be less than r. On the other hand, if y_{ij} is 0 the constraints become non-binding since M is a very large number. Constraint (4) ensures that $n-1$ items are *shipped out* of the source actuator, which is chosen arbitrarily, to be delivered other actuators in the network. Constraints (5) are the flow-balance constraints which ensure that the total *flow into* an actuator is equal to one item retained by that actuator plus the total *flow out* of that actuator. Constraints (6) ensure that the flow from one actuator to another is zero if these two actuators do not communicate directly. Finally, the actuators are allowed to move anywhere on a two-dimensional plane (constraints (7)); y_{ij} variables representing direct communication between two actuators are required to be binary (constraints (8)); and the flow variables f_{ij} are all positive (constraints (9)).

As final note, the formulation given above (Equations (1)-(9)) is a mixed integer nonlinear program (MINLP). MINLPs are known to be among the most difficult optimization problems. Therefore, for computational efficiency, the Euclidean distances used in objective functions (1) and (2) and constraints (3) are linearized using a polygon approximation described in [25]. The polygon approximation is based on the observation that the Euclidean norm of a vector (*i.e.*, its length) is equal to the radius of the smallest circle centered at the origin containing the entire vector. This circle is replaced by a regular polygon centered at the origin (see Fig. 2). In this approximation scheme, the Euclidean norm of a vector is approximated by the apothem (*i.e.*, the distance from the center to a side) of the smallest polygon containing the entire vector.

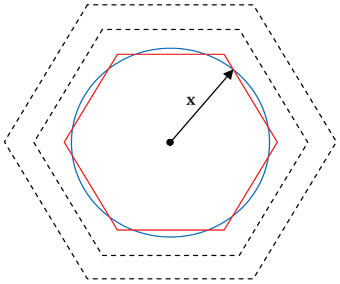


Fig. 2. Replacing the smallest circle containing a vector x with a regular polygon.

The error that results from this approximation is small and decreases as the number of polygon edges increases as shown in Fig 3. Numerical results show that the approximation error is negligible with polygons more than 20 edges.

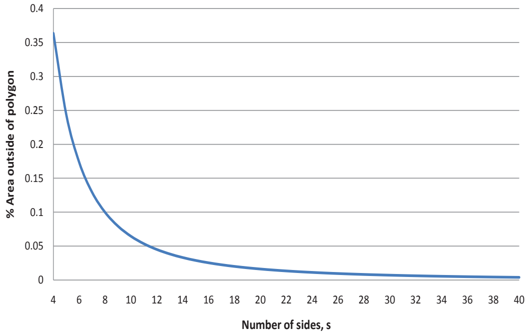


Fig. 3. Error with the increased number of sides of a polygon.

V. Experimental Evaluation

A. Experiment Setup and Performance Metrics

The optimization problem was modeled in AMPL [26], which is a mathematical programming language and solved with Gurobi Optimizer 4.0 [10]. We considered a region of size 800mX800m and generated random topologies with varying number of partitions (2 to 5) and number of actuators (10 to 30). The transmission range for the actuators was set to 50m. Each experiment was repeated with 30 different topologies and the average of the results was reported for significance. The following metrics are used for assessing the performance of the proposed solution:

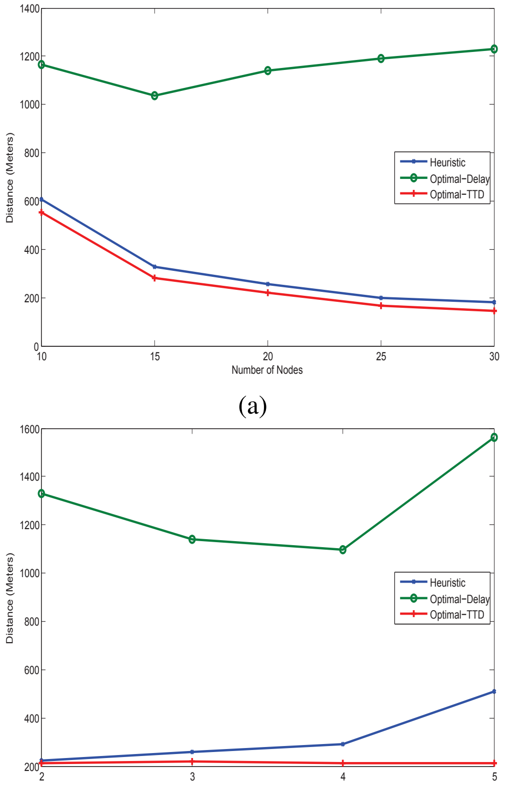
- *Total Travel Distance (TTD) of Actuators*: This is the total distance moved by all the actuators as a result of reconnecting the partitions. Our goal is to minimize this total distance.
- *Overall Delay for Connectivity Restoration*: This metric shows the total time it takes the actuators to restore the connectivity. We used the maximum travel distance among the all actuators to determine the delay since we assume that the actuators can simultaneously move to their final locations once they receive the location from the command node. In such a case, the total time will be the time it takes to travel the maximum distance. We assumed a speed of 1 meter/sec to calculate the delay.

We picked a heuristic to compare with the optimal solution. The heuristic idea is based on moving the actuators to the partition with the highest number of actuators. This may reduce the movement of large number of actuators and may save the total travel distance. Each partition determines its closest actuator to the largest one and fills the gap as in [20]. Finally, as a third approach we use our model with the objective function minimizing the maximum distance traveled by the actuators mainly for delay pruposes. However, we also report its performance in terms of total travel distance.

B. Experiment Results

This subsection presents the performance evaluation of our solutions and the heuristic with respect to total travel distance and delay. For both cases, we first fix the number of partitions and vary the number of actuators. Next, we fix the number of partitions and vary the number of actuators.

1) Total Travel Distance: The results of the experiments for total travel distance are shown in Figures 4a and 4b. The optimal solution provides the least travel distance.



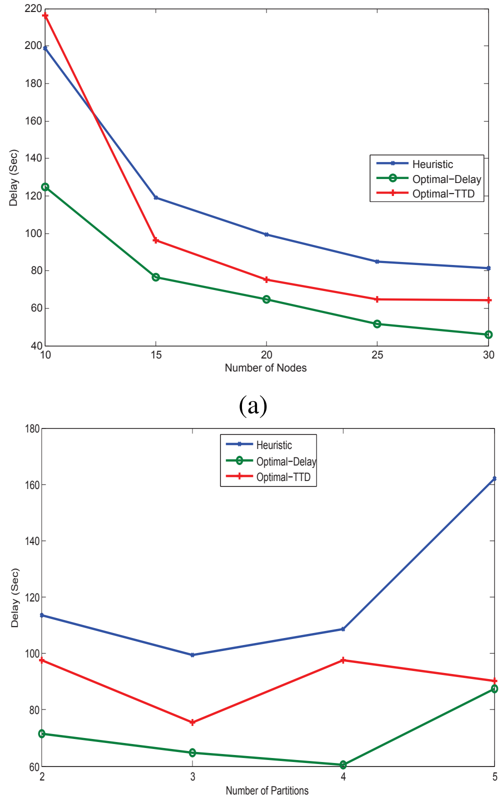
(a)

(b)

Fig. 4. Total travel distance for the three approaches.

Note that with the increasing number of partitions, the total travel distance somewhat increases for the all the solutions. This is expected since more nodes will need to move when the number of partitions is increased. For the increasing number of actuators, however, the total travel distance decreases for the heuristic and optimal solution. This can be attributed to the fact that more nodes will be in the same number of partitions and thus in average the partitions will be closer to each other reducing the travel distance of the actuators from these partitions.

2) Delay: The results of the experiments for the connectivity restoration delay are shown in Figures 5a and 5b. The optimal solution with the second objective provides the least delay. Our approach with the first objective also outperforms the heuristic while performing close to the optimal solution with the second objective.



(a)

(b)

Fig. 5. Total travel distance for the three approaches.

With the increased number of nodes, the time for the recovery is decreasing given that the partitions are becoming much closer to each other. However, with the increased number of partitions there is no regular behavior of the approaches since the number of partitions will not affect the maximum travel distance of all the actuators.

In summary, the optimal solution with the first objective provides the best total travel distance and performs better than the heuristic in terms of delay. If the delay is the sole concern, the second objective should be used. However, if both delay and energy are of concern, then the first objective will suit better. Obviously, depending on the application a weighted combination of both objectives can also be studied.

VI. CONCLUSION

In this paper, we have presented an optimal solution for connecting multiple partitions of a WSAN disconnected due to massive node failures. The solution is a mixed integer program depending on the flow-based idea. The experiment results indicated that optimal solution in terms of travel distance can be achieved with a reasonable delay in terms of recovery time.

For future work, we plan to work on two issues: First, the running time of the optimal solution increases with the increased number of nodes and partitions and thus more efficient solutions need to be investigated to speed up the process. Second, the obstacles and navigation problems will be considered in the movement.

REFERENCES

[1] J. Friedman, D. Lee, I. Tsigkogiannis, P. Aghera, A. Dixit, D. Levine, D. Chao, A. Kansal, W. Kaiser, and M. Srivastava, "Ragobot: A new hardware platform for research in wireless mobile sensor networks," in *Proceedings of International Conference on Distributed Computing in Sensor Systems*, 2005.

[2] G. T. Sibley and M. H. A. Rahimi, "Robomote: A tiny mobile robot platform for large-scale ad-hoc sensor networks," in *IEEE International Conference on Robotics and Automation*, 2002.

[3] M. B. McMickell, B. Goodwine, and L. A. Montestruque, "Micabot: a robotic platform for large-scale distributed robotics," in *ICRA*, 2003, pp. 1600–1605.

[4] I. F. Akyildiz and I. H. Kasimoglu, "Wireless sensor and actor networks: Research challenges," in *Elsevier Ad hoc Network Journal*, vol. 2, 2004, pp. 351–367.

[5] G. Wang, G. Cao, T. L. Porta, and W. Zhang, "Sensor relocation in mobile sensor networks," in *Proceedings of the 24th International Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM05)*, Miami, FL, March 2005.

[6] S. Zhou, M.-Y. Wu, and W. Shu, "Terrain-constrained mobile sensor networks," in *IEEE GLOBECOM '05*, December 2005.

[7] M. Younis and K. Akkaya, "Strategies and techniques for node placement in wireless sensor networks: A survey," in *Elsevier Ad-Hoc Network Journal*, vol. 6, no. 4, 2008, pp. 621–655.

[8] A. Abbasi, M. Younis, and K. Akkaya, "Movement-assisted connectivity restoration in wireless sensor and actor networks," in *IEEE Transactions on Parallel and Distributed Systems*, vol. 20, no. 9, Sept. 2009, pp. 1366–1379.

[9] K. Akkaya, F. Senel, A. Thimmapuram, and S. Uludag, "Distributed recovery from network partitioning in movable sensor/actor networks via controlled mobility," in *IEEE Transactions on Computers*, vol. 59, no. 2, Feb. 2010, pp. 258–271.

[10] "Gurobi optimizer," http://www.gurobi.com.

[11] A. Kansal, A. Somasundara, D., J. M. Srivastava, and D. Estrin, "Intelligent fluid infrastructure for embedded networks," in *MobiSys'04*, Boston, MA, June 2004.

[12] W. Wang, V. Srinivasan, and K.-C. Chu, "Using mobile relays to prolong the lifetime of wireless sensor networks," in *ACM Mobicom05*, Cologne, Germany, Aug. 2005.

[13] D. Goldenberg, J. Lin, A. S. Morse, B. Rosen, and Y. R. Yang, "Towards mobility as a network control primitive," in *MobiHoc'04*, Tokyo, Japan, 2004.

[14] K. Akkaya and M. Younis, "Coverage-aware and connectivity-constrained actor positioning in wireless sensor and actor networks," in *IEEE IPCCC'07*, April 2007, New Orleans, Louisiana.

[15] J. English, M. Wiacek, and M. Younis, "Core: Coordinated relocation of sink nodes in wireless sensor networks," in *Proceedings of the 23rd IEEE Biennial Symposium on Communications (QBSC06)*, Kingston, Ontario, Canada, May 2006.

[16] K. Akkaya and M. Younis, "Coverage and latency aware actor placement mechanisms in wireless sensor and actor networks," in *International Journal of Sensor Networks*, vol. 3, no. 3, 2008, pp. 152–164.

[17] P. Basu and J. Redi, "Movement control algorithms for realization of fault-tolerant ad hoc robot networks," in *IEEE Networks Journal*, vol. 18, no. 4, Aug. 2004, pp. 36–44.

[18] P. Sahoo, J.-P. Sheu, and W.-S. Lin, "Dynamic coverage and connectivity maintenance algorithms for wireless sensor networks," in *2nd International Conference on Communication Systems Software and Middleware (COMSWARE), 2007*, Bangalore, India, Jan. 2007.

[19] A. Abbasi, M. Younis, and U. Baroudi, "Restoring connectivity in wireless sensor-actor networks with minimal topology changes," in *2010 IEEE International Conference on Communications (ICC)*, Cape Town, South Africa, May 2010.

[20] F. Senel, K. Akkaya, and M. Younis, "An efficient mechanism for establishing connectivity in wireless sensor and actor networks," in *IEEE Global Telecommunications Conference (GLOBECOM), 2007*, Washington, DC, Nov. 2007.

[21] X. Chen, Z. Jiang, and J. Wu, "Mobility control schemes with quick convergence in wireless sensor networks," in *IEEE International Symposium on Parallel and Distributed Processing (IPDPS), 2008*, Miami, FL, Apr. 2008.

[22] L. Hu and D. Evans, "Localization for mobile sensor networks," in *MobiCom'04*, Philadelphia, PA, Sept. 2004.

[23] R. R. Meyer, "On the existence of optimal solutions to integer and mixed-integer programming problems," *Mathematical Programming*, vol. 7, pp. 223–235, 1974.

[24] G.-H. Lin and G. Xue, "Steiner tree problem with minimum number of steiner points and bounded edge-length," *Inf. Process. Lett.*, vol. 69, pp. 53–57, January 1999.

[25] P. K. Nagilla, "Sensor Coverage and Actors Relocation in Wireless Sensor and Actor Networks (WSAN): Optimization Models and Approximation Algorithms," Master's thesis, University of Missouri, Columbia, Missouri, 2010.

[26] R. Fourer, D. Gay, and B. Kernighan, *AMPL: A mathematical programming language*, 2nd ed. Brooks/Cole Publishing Company, 2002.