On Performance Evaluation of Reliable Topology Control Algorithms in Mobile Ad Hoc Networks (Invited Paper)

Ngo Duc Thuan[1,*], Hiroki Nishiyama[1], Nirwan Ansari[2], and Nei Kato[1]
[1]Graduate School of Information Sciences, Tohoku University, Japan
[2]Advanced Networking Lab., ECE Department, New Jersey Institute of Technology, USA
*thuan@it.ecei.tohoku.ac.jp

***Abstract*— Energy consumption and network connectivity are two of the important research issues that are yet to be resolved in mobile ad hoc networks (MANETs). As taken advantage of in static networks, reliable topology control algorithms are also considered to be a good approach for mobile networks. However, a more adequate evaluation of these algorithms regarding mobility is still needed. In this paper, we evaluate the performance of some well-known topology control algorithms with various scenarios and measurements. The results show that Local Tree-based Reliable Topology (LTRT), a recently proposed algorithm, is the most scalable method and provides the most benefit in terms of redundant connectivity.**

I. INTRODUCTION

The Mobile Ad hoc Networks (MANETs) area has received much attention not only in research, but also in many real applications. However, a number of research issues are yet to be resolved. In addition to minimizing energy consumption, mobile networks also focus on guaranteeing network connectivity as an important requirement.

Topology control has been used in static networks as an efficient method for saving energy. Each node will be able to decide its appropriate transmission power instead of assigning homogeneous value to the whole network by using topology control algorithms. Topology control can be made more cost effective by a localized version, which no longer requires a central node to collect information of the whole network. Each node can acknowledge its neighbors based on the information of its local graph. Thus, localized topology control algorithms have been applied to minimize energy consumption in static networks. Accordingly, they are also tailored for mobile networks.

However, some studies [1] [2] reveal that using such algorithms for mobile networks may lead to the loss of network connectivity because of moving nodes. Each node may not be able to decide its transmission power correctly because of the outdated information received from its neighbors. Therefore, our work focuses on evaluating the effect of moving nodes on network connectivity. Our study takes into account some well-known localized algorithms, namely Local Minimum Spanning Tree (LMST) [3], Relative Neighborhood Graph (RNG) [4], Local Shortest Path Tree (LSPT) [5], Fault-tolerant Local Spanning Subgraph (FLSS) [6], and the recently proposed Local Tree-based Reliable Topology (LTRT) [1] which is capable of generating reliable topology for static networks.

LMST, RNG, and LSPT are considered as typical algorithms that construct connected topologies, minimize the network redundancy, and have low complexity. On the other hand, in addition to minimizing the transmission power, FLSS and LTRT aim at preserving k-edge network connectivity, which makes the network more reliable. Among the algorithms that guarantee k-edge connectivity, FLSS is proven as a min-max optimal algorithm and LTRT is near-optimal. However, the computational cost of LTRT is much lower than that of FLSS, and thus, is more practical.

This work attempts to evaluate the five algorithms in MANETs to figure out the factors that affect the network connectivity. To simulate the movement of network nodes, we use a typical mobility model called random waypoint mobility model, which has been widely used to evaluate routing protocols in MANETs [7] [8]. The experiment results show that the moving speeds of nodes and the interval of refreshing network information have considerable effect on network connectivity. Furthermore, the changes of network connectivity vary among the above algorithms. We then accomplish a further evaluation of FLSS and LTRT that show the superiority to the others. Various values of k in the term "k-edge connectivity" are evaluated to measure the trade-off between network redundancy and connectivity. Evaluation results are used to discuss the improvements, specifically for mobile networks to fulfill the two requirements, namely energy consumption and network connectivity.

The remainder of this paper is organized as follows: Section II surveys some related works on localized topology control algorithms. Section III shows the effects of mobility on network connectivity and the need of reliable topology control algorithms. The performance evaluation of the algorithms in MANETs is presented in Section IV. Finally, Section V concludes the paper.

II. RELATED WORKS

LMST is a Minimum Spanning Tree (MST) based localized topology control algorithm proposed by Li *et al.* [1]. The common idea of MST-based algorithms is to find the minimum spanning tree of a graph that connects all vertices and has the minimum total link weight. In LMST, each node uses information from its 1-hop neighbors to construct a local minimum spanning tree. Thus, LMST can preserve the connectivity with

978-1-4244-3574-6/10/$25.00

TABLE I

COMPUTATION TIME OF THE CONSIDERED ALGORITHMS.

LMST	$O(m + n \log n)$
RNG	$O(n^2)$
LSPT	$O(m + n \log n)$
FLSS	$O(m(m + n))$
LTRT	$O(k(m + n \log n))$

a rather small average node degree. The node degree of a node is the number of edges that connect to it. By using Prim's algorithm, the time complexity of LMST is computed to be $O(n^2)$ when using simple searching and $O(m+n \log n)$ when using Fibonacci heap. Here, m is the number of edges and n denotes the number of vertices. LMST also has the lowest complexity among the five algorithms that are evaluated in this work, as shown in Table I.

Toussaint [4] proposed Relative Neighborhood Graph (RNG) that attempts to remove redundant edges while maintaining the connectivity. An edge (u, v) is redundant if there is a node w satisfying $d(w, u) < d(u, v)$ and $d(w, v) < d(u, v)$, where $d(u, v)$ is the distance between u and v. The constructed topology is the remained graph after removing all redundant edges. RNG is also carried out in the local graph of each node to become a localized topology control algorithm. Cartigny *et al.* [9] prove that with the same graph, the topology generated by LTRT is a sub-graph of RNG's topology. Therefore, the average node degree and total transmission power of RNG are higher than those of LMST. RNG can be implemented with a computational cost of $O(n^2)$.

By using the simple idea that an edge will be removed if there is a 2-hop path between the two nodes consuming a lower transmission power than the directed way, Rodoplu and Meng [10] proposed another topology control algorithm. Li and Halpern [5] then extended the algorithm by using k-hop paths instead of 2-hop paths. Accordingly, the algorithm can be explained as carrying out Dijkstra's algorithm [11] for a graph with the weight function $E = d^\alpha$ where E and d are the weight and length of the edge, respectively. Localized version of this algorithm is called Local Shortest Path Tree (LSPT) and can run in $O(m + n \log n)$ time when using Fibonacci heap.

Although LMST, RNG, and LSPT can preserve a connected topology, the network is still 1-edge connected. Consequently, the connection can easily be dropped when one link is broken. In order to come up with a fault-tolerant solution, Li and Hou [6] proposed Fault-tolerant Local Spanning Subgraph (FLSS) algorithm that guarantees k-edge connectivity if the original network is k-edge connected. In terms of maximum transmission power, FLSS is proven to be a min-max optimal algorithm. However, the complexity of FLSS is $O(m(m+n))$, which is not applicable for dense networks.

Recently, Miyao *et al.* [1] proposed Local Tree-based Reliable Topology (LTRT), a reliable topology control algorithm that can preserve k-edge connectivity. While FLSS is a min-max optimal algorithm, LTRT is a near-optimal one, but it has much lower complexity, $O(k(m + n \log n))$. The initial proposal of tree-based reliable topology (TRT) was made by Ansari *et al.* [12] and LTRT can be considered as a localized version of TRT.

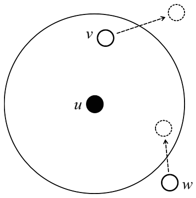

Fig. 1. The local view of node u at time t and $(t + \Delta t)$.

III. MOBILITY EFFECTS ON TOPOLOGY CONTROL

A. *Mobility effects*

In static networks, to reduce energy consumption, topology control is used to help each node decide its own transmission power while also guaranteeing the network connectivity. After constructing the topology, every node only needs to transfer information in its range. However, in MANETs, the connectivity might be lost as a consequence of node movement. Fig. 1 is a circumstance of local graph in MANETs. The figure demonstrates a local view of node u and its local graph after an interval Δt. From the local view of u, there are two types of neighbor movements that can cause the loss of connectivity: moving out of the local area, like v, and moving into the local area, like w. If node u does not have the updated information of its neighbors, it will lose the connection with v and will not be able to use w to calculate the new topology. The two situations imply the two methods of improvement, namely frequently updating the topology and using redundancy of edges. These two fault-tolerant strategies are described next.

B. *Fault-tolerant strategies*

The first strategy that can be applied to maintain the network connectivity is to frequently update the topology. This is obvious because whenever the nodes know the updated information of their neighbor, they can decide the appropriate transmission power to keep their connections. However, the size of the refreshing interval needs to be considered. If the movement is slow while the interval is too short, the cost of topology construction will be needlessly high.

The second strategy is preserving k-edge connectivity. As the previous example, a connection between two nodes can be easily broken in mobile networks. As a result, the algorithms that only guarantee 1-edge connectivity, such as LMST, LSPT and RNG, may not be suitable for MANETs because the loss of only one edge may lead to the loss of the whole network connectivity. Although there is a trade-off between maintaining connectivity and minimizing energy consumption, FLSS and LTRT should be more suitable for MANETs. Because they can generate k-edge connected topologies, the network will be more fault-tolerant. Moreover, these two algorithms are also more scalable than the others because the value of k can easily be changed. We can achieve more fault-tolerant topologies by using bigger values of k. However, the cost of topology construction and the total transmission power will be

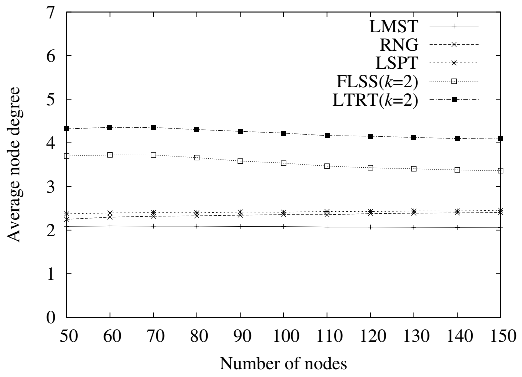

Fig. 2. Average node degree.

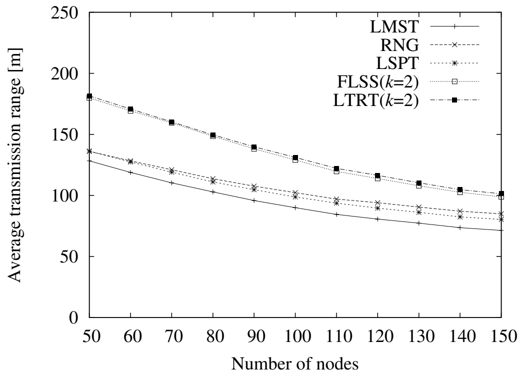

Fig. 3. Average transmission range.

higher in such a case. The trade-off between connectivity and energy consumption can be seen clearly by evaluating LTRT and FLSS for different values of k.

IV. PERFORMANCE EVALUATION

A. Simulation setups

We evaluate the performance of the five algorithms, i.e., LMST, RNG, LSPT, FLSS, and LTRT, by using Network Simulator version 2 (NS2). To simulate the algorithms in MANETs, we use random waypoint mobility model that was first proposed by Johnson and Maltz [13] and has been considered as a typical mobility model to evaluate routing protocols in MANETs [7] [8]. The model gives mobile nodes pause times between changes of moving speed or direction. In each process, after a pause time in current position, a mobile node moves to a random destination with speed in specified range. Until arriving to the chosen destination, the node keeps its direction and speed. In our simulation, the scenarios are generated on an area fixed to 1000m×1000m with the maximum transmission range set to 250m for each node. The number of nodes, moving speeds and topology update intervals are changed in conducted experiments. Corresponding to each different number of nodes or moving speed, one hundred scenarios are generated to find the average results for each experiment. With FLSS and LTRT, the value of k is also individually assigned in each experiment.

We introduce a metric named *connectivity ratio* to evaluate the performance of the considered algorithms in MANETs. Connectivity ratio measures the network connectivity at a certain snapshot in time. Every node always keeps its neighbor list obtained by the latest topology construction. However, at an arbitrary moment, the positions of all nodes may change and a node may be too far from another to keep the link. Connectivity ratio demonstrates the connectivity status of the whole graph constructed by all nodes and existing links. It is the percentage of connected node pairs out of all possible node pairs in the network. A node pair (u, v) is called connected if there is a path from u to v and vice versa. If n is the number of nodes and V is the set of nodes, connectivity ratio, which is indicated by C, can be calculated by the following equation.

$$C = \frac{\sum\limits_{u,v \in V} c_{uv}}{n(n-1)} \tag{1}$$

where

$$c_{uv} = \begin{cases} 1, & \text{if } u \neq v \text{ and } (u,v) \text{ is connected,} \\ 0, & \text{otherwise.} \end{cases}$$

Connectivity ratio is calculated without data transmitting. In each simulation, a snapshot is taken right before an arbitrary topology update. For example, if t is the time of a topology update, the time of taking snapshot will be $(t - \varepsilon)$, where ε is the assigned error.

B. Simulation results

1) Node degree and transmission range: We first make a comparison between the five algorithms in terms of average node degree and transmission range. The algorithms are evaluated with different scenarios with the number of nodes varying from 50 to 150. In this evaluation, LTRT and FLSS are executed with $k = 2$. Because this evaluation can be done without mobility, the scenarios are generated with moving speed set to zero.

Fig. 2 demonstrates average node degree of the topologies constructed by the five algorithms, respectively. As topology control algorithms attempt to preserve network connectivity with minimal energy consumption, they tend to find the fewest and nearest 1-hop neighbors for each node. Therefore, the average node degree is not affected by the number of nodes in the network.

However, the network density has a strong relationship with the needed transmission power. In sparse networks, because every node is far from each other, a node may have to use higher transmission power to connect enough neighbors. Fig. 3 also demonstrates that the topologies generated by all algorithms require a higher average transmission range when the number of nodes is smaller. As demonstrated in the two figures, LTRT needs significantly higher node degree than FLSS, but LTRT needs slightly higher transmission range. LMST, RNG, and LSPT preserve only 1-edge connectivity, and thus, evidently need lower node degree and transmission range.

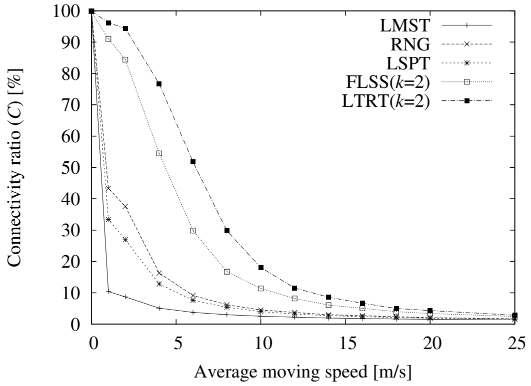

Fig. 4. Effect of moving speed on network connectivity.

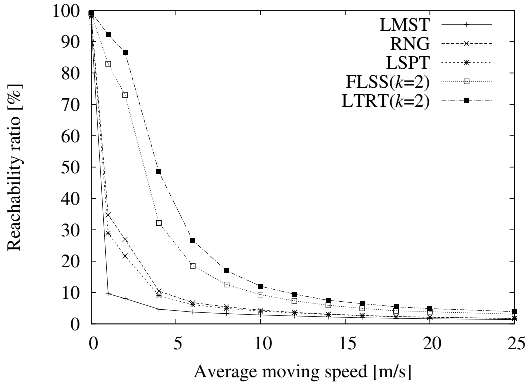

Fig. 5. Reachability ratio of the packets being transmitted.

2) Effects of moving speed on network connectivity: We then evaluate the performance of all algorithms in case of mobile nodes. In the same area as the previous experiment, the number of nodes is fixed to 100. In order to evaluate the effect of moving speed on network connectivity, the scenarios are generated with an average moving speed, which varies from 1m/s to 25m/s. LTRT and FLSS are evaluated with $k = 2$. In the experiment measuring connectivity ratio, the topology update interval is set to five seconds (i.e., $\delta t = 5$s).

Fig. 4 shows that the moving speeds have a strong effect on connectivity ratio. All the evaluated algorithms, especially LMST, RNG, and LSPT, are considerably sensitive to mobility. When the average moving speed is larger than 15m/s, no algorithm can generate a topology that has connectivity ratio higher than 10%.

As mentioned before, connectivity ratio is calculated without data transmitting. Therefore, we attempt to conduct another experiment to measure the success of transmitting data packets under the effect of moving speed. In this experiment, $\delta t = 10$s and data packets are randomly transmitted every second by a flooding method. We calculate the *reachability ratio* denoting the percentage of nodes, which received at least one packet, out of the all nodes. Because this experiment is conducted with $\delta t = 10$s and data packets are randomly transmitted during simulation time, the calculated reachability ratio proximately implies the state of network connectivity at $\delta t/2$ seconds after a topology update (i.e., five seconds). Therefore, it can be compared with connectivity ratio that is calculated with $\delta t = 5$s in the previous experiment. Note that all the snapshots used to calculate connectivity ratio are taken right before the next topology update.

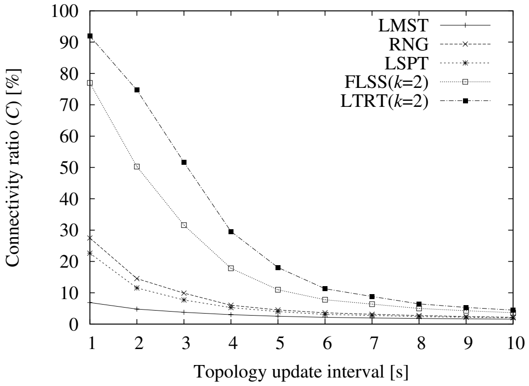

Fig. 6. Effect of topology update interval on network connectivity.

Fig. 5 demonstrates that the moving speeds also have a considerable effect on reachability ratio. LTRT and FLSS send data packets more successfully than others when $k = 2$. On the other hand, the calculated reachability ratio is lower than connectivity ratio corresponding to each moving speed. This is because it depends on the type of routing protocol used in the simulation. In our simulation, we only evaluate reachability ratio by using the Ad hoc On-Demand Distance Vector (AODV) routing protocol. Further evaluation with other routing protocols will be addressed in our future work.

3) Effects of topology update interval on network connectivity: Besides moving speed, the effect of topology update interval, δt, on connectivity ratio is also evaluated by running simulations with different intervals, from one to ten seconds. In this experiment, the average moving speed is fixed to 10m/s and the number of nodes in all scenarios is set to 100.

As evident from Fig. 6, frequent topology updating can help preserve network connectivity. The shorter the topology update interval is, the higher the connectivity ratio will be. The result also demonstrates that LTRT is more reliable than FLSS.

4) Comparison between LTRT and FLSS: We then evaluate the two reliable topology control algorithms, LTRT and FLSS, with different values of k which varies from one to seven. We conduct two experiments to compare LTRT and FLSS in terms of average transmission range and connectivity ratio. The number of nodes is fixed to 100. The experiment measuring average transmission range is conducted without mobility. On the other hand, in the experiment evaluating the effect of the change of k on connectivity ratio, the average moving speed is 20m/s and topology update interval, δt, is set to five or ten seconds.

Fig. 7 shows the average transmission range that is needed for the two algorithms corresponding to each value of k. The result shows that when k is bigger, not only the computation time becomes longer, but the needed transmission range also becomes higher. Although the topology control algorithms, certainly, have no meaning if all the nodes in the constructed topology still need a transmission range close to the maximum (i.e., 250m), as shown in the figure, LTRT needs a slightly

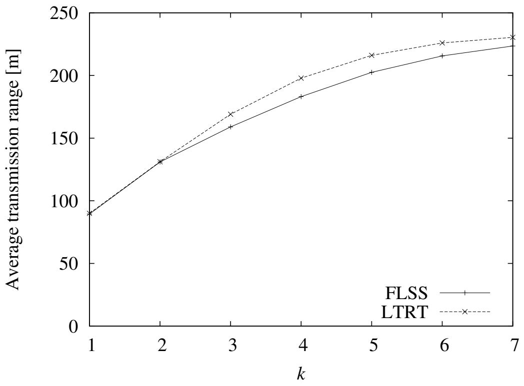

Fig. 7. Average transmission range with different values of k.

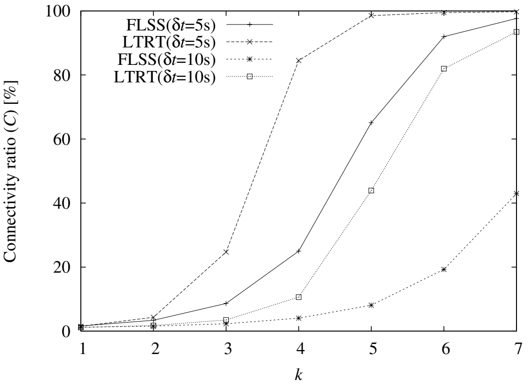

Fig. 8. Connectivity ratio with different values of k.

higher transmission range than that needed by FLSS. It can be acceptable because the complexity of LTRT is much lower than that of FLSS. Moreover, Fig. 8 shows that LTRT can achieve significantly better connectivity than FLSS when we use a bigger value of k. Therefore, LTRT is more suitable than FLSS in MANETs. Figs. 7 and 8 depict the trade-off between transmission power and network connectivity in case of the both algorithms.

Fig. 8 shows that decreasing the value of the topology update interval also can help improve network connectivity. For example, in the case where $k = 4$, LTRT can achieve connectivity ratio only about 10% when $\delta t = 10$s, but it is able to achieve more than 80% connectivity ratio when $\delta t = 5$s. Therefore, we may be able to choose an appropriate value of δt based on the value of k and the moving speed of the network nodes.

C. Discussion

The result of our conducted experiments shows that the well-known topology control algorithms, which maintain 1-edge connectivity and were successfully applied for static networks, are not suitable for MANETs. On the other hand, the algorithms, such as LTRT and FLSS that can construct k-edge connected topology, might be tailored for the mobile case. However, there is a trade-off between minimizing energy consumption and preserving network connectivity, and thus, the scalable algorithms are preferred.

When comparing with FLSS, LTRT is slightly more redundant, but it can generate a much more reliable topology. However, LTRT should be more applicable for MANETs, because of its lower complexity. Although LTRT shows it superiority to be a suitable method for MANETs, the cost of constructing topology still needs to be considered. The value of k cannot be too large. Therefore, some methods such as using network characteristics to decide the appropriate value of k for each local graph should be a good supplement.

V. CONCLUSION AND FUTURE WORK

We have accomplished the performance evaluation of LTRT, FLSS, and several other well-known topology control algorithms in MANETs. The results exhibit that both moving speeds of the nodes and topology update intervals have significant effects on network connectivity. Among the evaluated algorithms, LTRT and FLSS, which can preserve k-edge connectivity, are more reliable when they are applied in MANETs. With the low complexity, LTRT can be considered to be the most scalable and applicable method that can be tailored for MANETs. Our future work will include further evaluations of LTRT in various network scenarios to find the appropriate value of k and choose a suitable topology update interval for each type of network.

REFERENCES

[1] K. Miyao, H. Nakayama, N. Ansari, and N. Kato, "LTRT: An Efficient and Reliable Topology Control Algorithm for Ad-Hoc Networks," *IEEE Transactions on Wireless Communications*, vol. 8, no. 12, pp. 6050–6058, Dec. 2009.

[2] J. Wu and F. Dai, "Mobility-sensitive topology control in mobile ad hoc networks," *IEEE Transactions on Parallel and Distributed Systems*, vol. 17, no. 6, pp. 522–535, Jun. 2006.

[3] N. Li, J. C. Hou, and L. Sha, "Design and analysis of an MST-based topology control algorithm," in *Proc. IEEE INFOCOM*, San Francisco, USA, Apr. 2003.

[4] G. Toussaint, "The relative neighborhood graph of finite planar set," *Pattern Recognition*, vol. 12, no. 4, pp. 261–268, 1980.

[5] L. Li and J.Y. Halpern, "Minimum Energy Mobile Wireless Networks Revisited," in *Proc. IEEE International Conference on Communications*, St.-Petersburg, Russia, Jun. 2001.

[6] N. Li and J. C. Hou, "Localized fault-tolerant topology control in wireless ad hoc networks," *IEEE Transactions on Parallel and Distributed Systems*, vol. 17, no. 4, pp. 307–320, 2006.

[7] T. Camp, J. Boleng, and V. Davies, "A Survey of Mobility Models for Ad Hoc Network Research," *Wireless Communications and Mobile Computing*, vol. 2, no. 5, pp. 483–502, Sep. 2002.

[8] J. Broch, D. Maltz, D. Johnson, Y. Hu, and J. Jetcheva, "A performance comparison of multi-hop wireless ad hoc network routing protocols," in *Proc. International Conference on Mobile Computing and Networking*, Dallas, USA, Oct. 1998.

[9] J. Cartigny, D. Simplot, and I. Stojmenovic, "Localized minimum-energy broadcasting in ad-hoc networks," in *Proc. IEEE INFOCOM*, San Francisco, USA, Apr. 2003.

[10] V. Rodoplu and T.H. Meng, "Minimum Energy Mobile Wireless Networks," *IEEE Journal on Selected Areas in Communications*, vol. 17, no. 8, pp. 1333–1344, Aug. 1999.

[11] E. W. Dijkstra, "A note on two problems in connexion with graphs," *Numerische Mathematik*, vol. 1, no. 1, pp. 269–271, Dec. 1959.

[12] N. Ansari, G. Cheng, and R. Krishnan, "Efficient and reliable link state information dissemination," *IEEE Communication Letters*, vol. 8, no. 5, pp. 317–319, May 2004.

[13] D. Johnson and D. Maltz, "Dynamic source routing in ad hoc wireless networks," *Mobile Computing*, vol. 353, pp. 153–181, 1996.