2009 Seventh Annual Communication Networks and Services Research Conference

Energy Balancing Routing Schemes for Low-Power Wireless Networks

Eun-Sook Sung, Miodrag Potkonjak
Computer Science Department
University of California, Los Angeles, CA 90095
{eunsook, miodrag}@cs.ucla.edu

Abstract— **We address the problem of energy balancing in multi-hop wireless networks to optimize their operational lifetime. We first present a localized routing algorithm. Then, for a performance evaluation of the localized algorithm, we present a centralized routing algorithm because a centralized approach with global knowledge can closely approach the optimal solution. Unlike many previous energy-aware routing protocols, both of our presented algorithms consider total energy expenditure for guaranteed packet delivery. We use simulation to study their performance under a variety of network conditions. The simulation results show, firstly, that both schemes achieve significant improvement of energy balancing proportional to node density and, secondly, our proposed localized routing scheme achieves a performance comparable to the centralized scheme under dense networks.**

Keywords: low power networks, energy balancing

I. INTRODUCTION AND BACKGROUND

Wireless ad-hoc and sensor networks have received great attention during the last few decades due to their potential applications. Routing, which guides the way of data exchanges between nodes, is one of the most fundamental tasks of such networks and significantly affects the overall performance of the networks.

In this paper we design and evaluate an energy efficient routing algorithm that optimizes the operational lifetime. Addressing the lifetime of a network, although the lifetime of each node in the network is an important concern in some applications, overall network lifetime is more consequential in many applications. The definition of network lifetime may depend on the design purpose and application scenario, but we follow the most general definition: the time to the first node's death [1]. The rationale for this definition is, firstly, we want to optimize the worst-case to avoid any gaps from developing. Secondly, we want to evaluate the performance of routing methods without any prior knowledge of network scenarios or topology. Other definitions, such as the time to the failure of a specific functionality or the time to the first network partitioning, require detailed information of the network.

Recently, several studies in the wireless ad-hoc community have addressed different aspects of wireless link properties. Studies have shown that wireless links are almost always lossy and asymmetric as a result of energy constraints and/or the low power nature of radios used [2][3][5]. Specifically, the communication range of wireless networks exhibits high variability, as opposed to conventional networks, and the wireless links have different costs although the neighbors are within the same communication range. Therefore, in order to achieve reliable communication for guaranteed packet delivery, retransmission may be required to recover from link errors, and thus, the estimated energy for packet delivery must include this potential retransmission cost. Also, because links aren't symmetric, the estimated energy costs of forward and backward transmissions have to be calculated separately. Accordingly, an accurate link cost estimation is one of the most important building blocks for the development of an energy-aware routing protocol. We present localized and centralized routing algorithms that consider the requirements imposed by lossy wireless links. The proposed algorithms take great advantage of the simple and accurate link cost estimation. Specifically, the algorithms predict the expected energy cost based on our novel link cost estimation technique via the expected number of transmissions along routing paths.

II. SYSTEM MODELS AND ASSUMPTIONS

Not to simplify the routing problem, but to enable us to evaluate the effectiveness of the route learning schemes, our work first assumes immobile nodes. Second, nodes are equipped with the same features and same battery capacity. Third, the proposed algorithms are location driven, and thus, nodes know the location information of the node to which they want to send their packets, using a location service. Lastly, we assume a node has a localization system that makes possible to know its position and its one-hop neighbor's information.

Communication model: In the communication to be conducted in our work, when a node sends a packet, it expects the acknowledgement from the receiver in order to ensure that the packet is correctly received. Because bandwidth requirements and transmission rates are low in most sensor networks, we assume that every transmission is completely scheduled and far apart, which results in the absence of interfering transmissions. To realize this assumption, the waiting/idle period is a few times longer than the sum of the time required for the receiver to receive the packet and to transmit an acknowledgement back. After the waiting period interval, if the sender does not receive an acknowledgement from the receiver, the sender retransmits the packet. In addition, when a packet is forwarded through the network, all nodes except destination nodes receive packets and immediately forward them. We examine three types of traffic scenarios in our simulations: peer-to-peer,

978-0-7695-3649-1/09 $25.00 © 2009 IEEE
DOI 10.1109/CNSR.2009.70

data collection, and Zipf's, and unless specified, the peer-to-peer traffic model is considered.

Link model: We describe how to estimate the link costs for transmitting a packet between two nodes. To correctly estimate required energy amounts, the link model has to be built from lossy wireless link data. The link data for our link model is from a set of low power wireless link data [3]. We first estimate the reception rate of a link. Links between a pair of nodes that are within the communication range are assigned lossy link qualities. The link qualities are based on the cumulative density function (CDF) [2]. Specifically, a geometric distance between two nodes determines the relationship between link quality and CDF, and then, considering wireless link variability, a random number is used to decide the percentage of links that will have lower reception rate than the random number, by using inverse CDF. As a next step, we estimate the expected number of forward and backward transmissions between two nodes with a lossy link. For a detailed explanation, we use RR_{AB} (the rate of node S_A's successfully sending a packet to node S_B) and RR_{BA} (the rate of node S_A's successfully receiving a packet from node S_B.) The expected forward retransmission number can be denoted by $1/(RR_{AB} \times RR_{BA}$.) Since the communication we adopt expects an acknowledgement from the receiving node back to the sender for guaranteed delivery during these retransmission sessions, node S_B will send acknowledgement packets back to node S_A, and the expected backward retransmission number of packets is computed by $1/(RR_{BA})$. In summary, the expected total required number of retransmissions is the sum of the forwarding and backward transmission numbers, which is $1/(RR_{AB} \times RR_{BA}) + 1/(RR_{BA})$ [7]. Finally, the link costs assigned for the calculation of the energy expenditure become the function of the expected total required number of packets and the energy consumption model below.

Energy consumption model: The components that consume power on a node include the radio, the microprocessor, and the attached modules. We assume that the energy spent in a node mainly depends on the communication between nodes. We estimate energy consumption using the measurements proposed in [4]. In this formulation, the power spent in transmission is referred to as tx and set to 1.9W, the power spent in reception is referred to as rx and set to 1.5W, and the power spent in idle time is referred to as idle and set to 0.75W. We have implemented a simple energy model in which every time a packet is transmitted, the total energy of the node decreases by the value tx multiplied by the transmission time. The same formula is applied to determine the decrease in battery upon reception (using rx), or when idle (using idle.)

Energy path costs using statistical modeling: In our localized scheme, the estimation of an energy cost to the destination is needed to find the best next-hop. In general, the energy cost from the current node to the destination can be divided into two values: the value from the current node to a neighbor node and the value from the neighbor to the destination. The first value denotes the expected energy cost from the current node to its neighbor node, which is available to the current node according to the method described in our link model. To obtain the second value, which is the expected energy cost to transmit a packet from a neighbor to the destination, we use the Monte Carlo simulation and linear fits. Using the distance between its neighbor and the destination, the required energy cost to reach the destination from its neighbor is statistically estimated. Specifically, to predict the second value, we estimated the energy cost from the minimum energy path routing for that portion of routing, using the geometric distance between the one-hop neighbor and the destination. Data is obtained for large number of cases by using the Monte Carlo simulation, and the execution of linear fits to the data results in the linear models. Fig.1 shows the linear fit of expected energy costs versus geometric distance for the case of density 0.1 networks. The x-axis indicates the geometric distance from a neighbor to the destination as the predictor, and the y-axis indicates the estimated energy expenditure of the optimal path.

The summary of the linear fits for different node densities, which will be used in calculating expected energy costs from a neighbor to a destination, is presented in Table 1. The density of the first column is defined as the number of nodes per square meter in two-dimensional networks. As reported in Table 1, the R squared value is approximate to 1 when the density becomes larger. The summary of linear fits indicates that the approximation is statistically good.

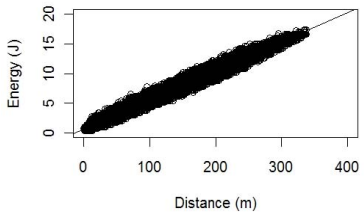

Figure 1. Energy expenditure (J) vs. Distance (m) for density 0.1: y=0.0492 * x+0.59

TABLE I. LINEAR REGRESSION FITS FOR DIFFERENT DENSITIES

Density	Intercept	Slope	R^2
0.1	0.59	0.0492	0.9374
0.2	0.4488	0.04001	0.9418
0.3	0.4054	0.03709	0.9623
0.4	0.4019	0.03489	0.9672
0.5	0.4037	0.03347	0.967
0.6	0.3774	0.03277	0.9704
0.7	0.3498	0.02815	0.9774
1.0	0.4031	0.0244	0.9846

III. LOCALIZED ENERGY BALANCING

We propose a localized energy-balancing algorithm, called L-LIMAX. L-LIMAX consists of two policies: a greedy forwarding eNergy-Aware aDvance (NAD) policy and an energy-centric routing eNergy-Aware Prospect (NAP) policy. We introduce our novel link preference strategy that only assumes one-hop neighborhood information.

A. Link Preference

We denote S_i a neighbor of S_C, where S_C is the current node holding a packet to transmit, and S_D the destination of the packet. Above, we decomposed the energy cost from the current node to the destination into two components. In most greedy approaches, when a packet is forwarded, the neighbor closest to the destination is selected as the next hop to achieve the minimization of the cost to the destination, without considering the cost to a neighbor. However, we believe that the cost from the current node to a neighbor also has to be considered importantly. In [3], the authors identify bad quality links where due to high error probability nodes cannot exchange packets in most cases. As a result, to successfully forward a packet to neighbors with very low quality links, a lot of energy is needed. We want to avoid neighbors with bad quality links and thus, decide to consider the first component with equal importance as the second component. In summary, when choosing next hops, we attempt to exclude neighbors with low quality links, and simultaneously we try to gain as much reach as possible for efficient delivery. To achieve both of the goals, we weigh the two components equally and select a neighbor whose sum of the two components is the smallest among its neighbors. This link preference scheme is called NAD, and $NAD(S_i)$ denotes the estimated energy cost from the current node to the destination via neighbor S_i.

The next link preference scheme is called NAP. Our observation about how to prolong network lifetime is that a node that has suffered from low energy avoids spending its energy any more. NAP takes advantage of nodes' energy information. $NAP(S_C)$ is equal to ***good*** if S_C decides its energy level is not in trouble, when locally comparing to its neighbors. On the other hand, ***bad*** indicates a node has used up much more energy. A node determines its energy level by comparing the amount of energy consumption to those of its neighbors. Specifically, the way of computing $NAP(S_C)$ of the current node, S_C, is as follows. $NAP(S_C)$ is ***bad*** if S_C, has consumed over a certain percentage more than the arithmetic mean of its neighbors' energy consumption, because we need a certain tolerance constant. For example, we used 10% in our work, based on the simulation results.

B. L-LIMAX: Route Discovery

L-LIMAX is composed of two main policies: a NAD-based policy and a NAP-based forwarding policy. The NAD based forwarding considers not only the cost from the neighbors to the destination but also the cost from the forwarding node to the neighbors. Furthermore, in order to protect the nodes with low energy, the NAP based forwarding is provided. We give the simple formulation of L-LIMAX algorithm.

Algorithm 1. L-LIMAX

L-LIMAX(S_C, S_D):

Repeat

A. Compute $NAP(S_C)$

B. if $NAP(S_C)$ is ***good,***
Choose S_i whose NAD is minimum among S_C's neighbors and whose NAP is good, $S_C = S_i$

C. else
Choose S_i to which the energy cost is the minimum among S_C's neighbors, $S_C = S_i$

Until $S_i = = S_D$

A) To determine whether the energy level of the current node is ***good*** or ***bad***, we use the method described in the Link preference section. A node computes its NAP value. B) If $NAP(S_C)$ is ***good***, the current node S_C has energy relatively enough to find the most energy efficient route. Thus, it selects a neighbor whose NAD value is the minimum among its neighbors. Importantly, the node excludes the neighbors in trouble from the candidates for a next hop since the neighbors whose NAPs are ***good*** are preferred to relay the packet. To realize this, NAP maintains the neighbors that have relatively enough energy. Excluding a node that is in trouble is similar to a node with low-energy switching its state to sleep mode implicitly, to avoid a relaying task, and protecting its energy level.

C) On the other hand, if the NAP of a node is ***bad***, and the node holds a packet to transmit, the node might want to spend as little energy as possible. Hence, the node chooses a neighbor with the least link cost, no matter how low the energy level of the neighbor is. In other words, a ***bad*** node transmits a packet to the neighbor that requires the least amount of energy, to protect its energy level. As a result, the node that is in trouble prioritizes its own energy level over those of its neighbors. In brief, NAP directly (by choosing the node with the least cost link) and indirectly (by excluding bad neighbors) protects the nodes that have suffered from energy. In conclusion, L-LIMAX has the following beneficial properties:

Energy balancing: L-LIMAX does not overload any node because the likelihood of becoming a next hop for a relaying task falls as a node uses up its energy.

Loop free: L-LIMAX cannot get stuck in a loop because the nodes in the loop will reduce their energy after some period of time if they keep exchanging the packet. Finally, the packet will take another route towards a different node that has more energy.

Simple and fast cost estimation: L-LIMAX takes advantage of the simple but effective link cost estimation.

IV. CENTRALIZED ENERGY BALANCING

Mainly for a performance evaluation of the localized approach, we present a centralized routing algorithm for the purpose of energy balancing, called C-LIMAX. C-LIMAX conducts the routing task at a global vantage point, where the energy, location, and link cost measurements of all nodes are available. Note that we considered the total required energy expenditure for guaranteed delivery as an edge weight, while most other energy-aware routing protocols considered only transmission costs or did not take into account the retransmission costs. Admittedly, this centralized version is of limited practical interest; however the centralized algorithm with global knowledge can produce a potentially near optimal solution. Specifically, C-LIMAX can provide us performance benchmarks, which are useful for a performance evaluation of the proposed L-LIMAX.

As for notation, the critical node is denoted as the node with the least remaining energy in a network. Our centralized algorithm finds paths between all nodes in a network, using a two-step algorithm. The first procedure finds the optimal routing paths for any pair of source and destination nodes. After the total energy cost to send a packet over the link is assigned to the edge weight, the Dijkstra's shortest path algorithm [6] is used. Consequently, the first procedure is the minimum total energy routing (MTER). After the MTER is completed, it is apparent which node is the critical node that determines the lifetime of the network and its dissipated energy amount. This critical node information is recorded for the next step. The next procedure, called Re_route(), is to make the MTER paths to be iteratively re-routed by avoiding the critical node. The Re_route() procedure is just iterations of the update_route(), and the update_route() examines paths that include the critical node and aims to re-route those paths. Also, the iteration is continued until the energy consumption of the critical node approximates the arithmetic mean of all nodes' dissipated energy, as long as the network is sufficiently dense. The following is the simple formulation of the proposed centralized algorithm.

Algorithm 2. C-LIMAX

1) MTER()
 & record the information of the critical node
2) Re_route()
Repeat
 update_route() for any pair of source and destination nodes
 & examine whether the critical node has been changed; if changed, update the path and changed critical node information.
Until (the energy level of the critical node does not become lower anymore, or almost approximates the mean energy value of nodes in the network).

The task of the update_route() procedure is, firstly, to collect all the other possible paths with the same source and destination that do not include the critical node. To find those paths, the optimal paths for any pair of source and destination nodes from the MTER are used. The update_route(), then, calculates the energy consumption value of the critical node along each possible route in order to select only the paths in which the energy consumption at the new candidate for the critical node becomes less than the value of the current critical node. Third, the update_route() procedure takes the minimum energy path among all those candidate paths, in order to minimize the total energy consumption along the path, and at last, updates the path from the MTER or discovered earlier to this new path. Finally, it records the critical node and its energy information if the critical node information is re-identified. The paths from the update_route() are changed from the MTER paths to new paths that include only the nodes with less drain-out, by continuously avoiding the nodes with the most energy drain-out. In other words, the algorithm starts with the initial critical node information from the MTER and repeatedly modifies the critical node information to arrive at a better solution, where a better solution is quantified as the reduction of the consumed energy by the new critical node.

Our scheme achieves the property of energy balancing since the newly chosen paths repeatedly protect the energy levels of the critical node. The energy balancing becomes greater since the energy consumption of the re-identified critical node after new candidate paths are discovered becomes less and less than the energy value of the original critical node after the MTER.

C-LIMAX takes advantage of both of the energy information of the critical node, which determines the lifetime of a network, and the realistically estimated link quality for guaranteed delivery. In summary, C-LIMAX accomplishes its goal of maximizing the lifetime of a network by distributing energy consumption iteratively as many as possible over nodes, to protect a few critical nodes from running out of energy.

V. PERFORMANCE EVALUATION

In this section we present results from experiments using the proposed LIMAX methods in different traffic models and node densities. After completing the assigned tasks, the amount of energy dissipated by the critical node from the centralized, localized, and MTER algorithms is recorded each time. Also, we propose two novel metrics for comparison: upper and lower bound values.

Lower bound: the arithmetical average of energy dissipated by all nodes in a network when the minimum energy routing is applied. In the MTER, each node only spends the minimum energy value to accomplish the assigned task. Also, energy balancing is maximized when the energy consumption of the critical node approaches the arithmetic mean of energy values of the nodes in the network. Combining both facts, we argue that the lower bound of the dissipated energy of the critical node is the arithmetic mean of the energy dissipation of all nodes in the network.

Upper bound: the energy dissipated by the critical node when the MTER is applied. This minimum energy routing does not consider the energy level of the critical node and seeks the optimally minimum energy paths; therefore, the dissipated energy value by the critical node can be an upper bound from the point of the critical node's energy consumption.

We denote the network lifetime efficiency ratio (NLER) as the ratio of the Upper bound value to the dissipated energy amount of the critical node derived from the proposed algorithms. We also denote the fairness ratio (FR) as the ratio of the Lower bound value to the dissipated energy amount of the critical node derived from the proposed algorithms.

Each simulation is repeated more than 50 times in different random networks, and the average is recorded.

Choosing parameters: Two parameters are used in our proposed algorithms. The first one is the tolerance constant for L-LIMAX. Recall that a node compares its energy level with this constant multiplied by the arithmetic mean of the energy dissipated by its neighbors. We investigated the relationship between the different threshold values, varying from 0.01 to 0.5, versus the corresponding NLER, and the highest NLER was found at 0.1(omitted due to space). Thus, we decided to use 0.1 for the tolerance constant. The other parameter is for C-LIMAX. Since the goal of C-LIMAX is to achieve as much load sharing as possible, the iterations of the procedure update_route are continued until the energy value of the critical node approaches the average of the energy expenditure across the network, or as many iterations as possible. Accordingly, we have to get an idea of the number of iterations for C-LIMAX. Our result showed that up to 500 iterations produce increased NLER (omitted due to space); however, more than 500 iterations do not show any prominent performance improvement in the case of density 0.5 networks with 200 nodes.

A. Dimension (1D, 2D, 3D) & Density (How much network density affects)

The first set of simulation is aimed at grasping any relationship of the proposed algorithms with dimensions. We define density as the number of nodes in a network divided by area R, where $R = [0, l]^d$, l is the size of a network and d denotes the dimension. We simulated one-dimensional, two-dimensional, and three-dimensional networks.

For each dimension, we first experimentally measured the minimum density to be maintained in order to obtain better performance against the MTER. Simulation results show how the minimal density in which the proposed localized algorithm performs better than the MTER varies on dimensions. In one-dimensional networks, the minimal density for achieving better performance against the MTER is 1.00, while the value becomes 0.21 in two- dimensional networks. Further, in three-dimensional networks, 7% improvement is shown even with density 0.03. Clearly, the minimal density of higher dimensional networks is lower. This happens because in higher dimensional networks, finding neighbors to relay packets is easier; there are a more number of neighbors surrounding a node. Hence, in lower dimensional networks, more nodes should be distributed to achieve the desired performance, which indicates higher density is required from the dimensional density perspective.

Fig. 2 and 3 illustrate the average performance of the study over a range of different node densities in each dimensional network, where the network size is fixed as 10m. The y-axis shows the dissipated energy of the critical node, and the x-axis indicates different density values. Larger density in the fixed network size lets more nodes be deployed and accomplish the assigned peer-to-peer routing task, which incurs much larger dissipated energy.

According to our definitions of the Lower and Upper bound values, if the energy values from the algorithms are closer to the Lower bound, the algorithms perform well in terms of the dissipated energy of the critical node that determines the energy balancing. Fig. 2 and 3 show that the average energy expenditure lines of C-LIMAX and L-LIMAX still approach the Lower bound line closely, as the density increases. This indicates the applicability of the proposed algorithms in any dimensional networks as long as the minimal density requirement is satisfied. In Fig.3, as expected, C-LIMAX with global knowledge consumes lower energy compared to that consumed with L-LIMAX in many cases.

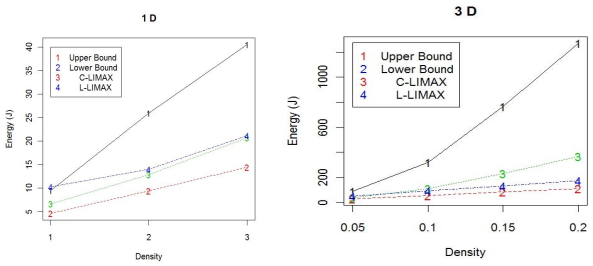

Figure 2. Avg. energy expenditure (J) vs. Density in one-dimensional (Left) and three-dimensional (Right) networks with size 10m

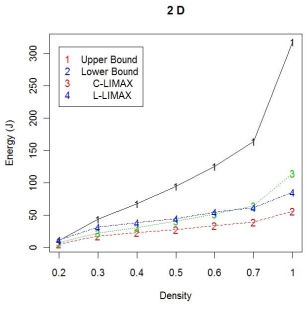

Figure 3. Avg. energy expenditure (J) vs. Density in two-dimensional networks with size 10m

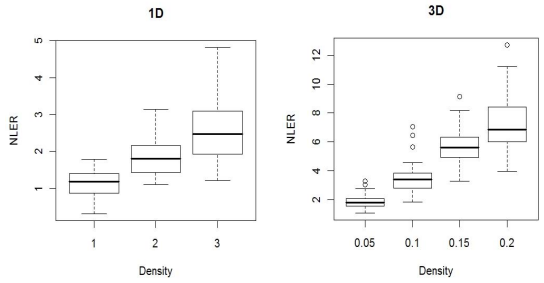

Figure 4. NLER of one- (Left) and three-(Right) dimensional networks

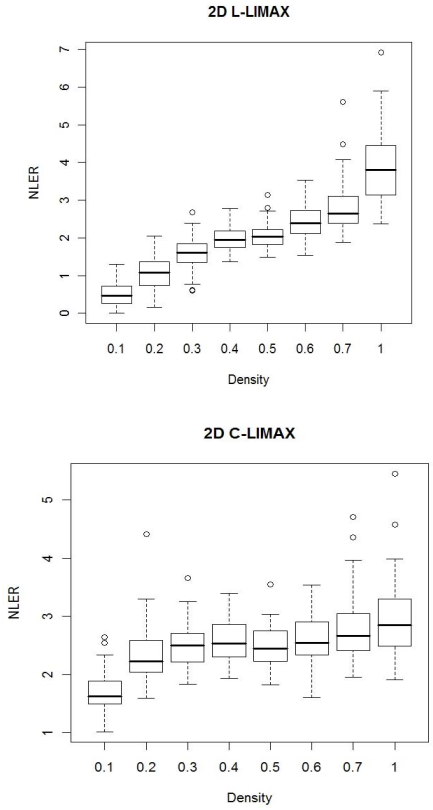

Figure 5. NLER vs. density of L-LIMAX(upper) and C-LIMAX(lower)

To further examine how much energy balancing can be achieved in L-LIMAX against MTER in each dimension, the NLER box-plots of one-dimensional and three-dimensional networks are provided in Fig.4. In one- and three-dimensional networks, the energy balancing can be achieved more than 2.5 and 6.5 times in higher density networks, respectively.

For the results of two-dimensional networks, the NLER box-plots of the localized and centralized algorithms are shown in Fig.5. The figures quantitatively illustrate how the energy-balancing extent changes with different node densities, as the node densities increase beyond the minimum level discovered earlier. In brief, the energy balancing can be achieved more than 3.5 (C-LIMAX) and 2.5 (L-LIMAX) times in higher density networks, respectively.

In particular, as node density increases, our localized algorithm takes advantages of more candidate neighbors for forwarding packets, and higher chance of load balancing can be achieved since a next hop is selected from the good neighbors. Also, because our centralized algorithm is based on the minimum energy routing, with an increase of the number of nodes in a given network size, the energy expenditure on each link is reduced. Ultimately, the NLER increases as the node density increases. In summary, Fig. 2, 3, 4 and 5 demonstrate the indication of the effectiveness of the proposed schemes in any dimensional networks as long as the minimal density requirement is satisfied.

B. Network size and traffic

We first evaluate how the proposed algorithms perform as the network size gets larger, in two-dimensional networks. The experiments are accomplished in the medium density networks (the density is fixed at 0.5.) In the medium density networks, a slightly positive effect of an increased network size on energy balancing is observed in both L-LIMAX and C-LIMAX (omitted due to space).

Fig. 6 reports more detailed experimental results at the different number of nodes to examine if the proposed schemes scale as the network size (the number of nodes) increases. From Fig. 6, the positive effects of an increased number of nodes on load (energy) balancing, in both localized and centralized algorithms, are observed in the medium density networks. As expected, C-LIMAX outperforms L-LIMAX.

Network traffic: We also examine how the proposed schemes perform in other traffic patterns, while in the peer-to-peer traffic model conducted in the previous sections each run is repeated until every node sends the packets to every other node. In data gathering traffic model, we assume that the data sink node is positioned at the origin in a two dimensional plane. Each node transmits a packet to the sink node in a round-robin fashion, one transmitter at a time; the rest of the nodes are in the listening mode and may receive packets for the purpose of relaying. In this traffic pattern, the nodes closet to the sink node will naturally drain their energy resources earlier, since these nodes participate much more in message forwarding on behalf of other nodes to the sink node. Thus, we decided to place more nodes near the sink node than the area further away from the sink node. In order to realize this placement, we use a polar network, where the position of a node is determined by a randomly chosen angle and radius values. In these polar networks, more nodes are naturally placed near the sink node.

The performance variation of the data gathering traffic model among various network densities are shown in Fig.7 (Left). It demonstrates a positive effect of increased density and at a density of 1.0, there is a median NLER of 7.6, 7.6 times energy balancing.

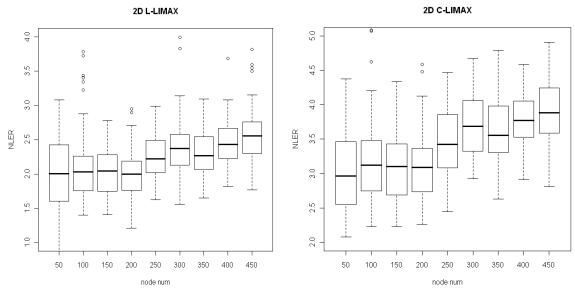

Figure 6. NLER vs. number of nodes for L-LIMAX (Left) and C-LIMAX (Right)

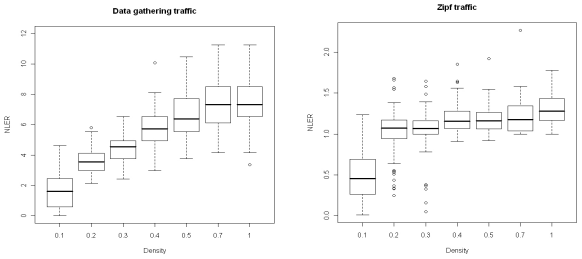

Figure 7. NLER vs. density for data gathering traffic in polar networks (Left) and for Zipf traffic (Right)

The next traffic model is based on the Zipf's distribution in randomly deployed networks. The results of the Zipf' traffic model are reported in Fig.7 (Right), which indicate that there is a limited positive effect of increased density. In fact, as expected, since there are only a few popular nodes in the Zipf's model, our scheme is not able to take advantage of the increased number of neighbors in denser networks.

C. Performance comparison between localized and centralized algorithms

In developing routing schemes only using local information, it is informative to analyze how well the localized scheme can achieve capability comparable to centralized schemes with global knowledge. This is important because centralized approaches do not scale in terms of computation and communication overheads. We observe that global knowledge does provide better performance in many cases; however, we also notice that in high-density networks, the localized algorithm can perform better in terms of the dissipated energy of the critical node. To further examine this finding in two-dimensional networks, Fig. 8, 9 and 10 are provided. In Fig.8, we experiment on the impact of density in the performance of the proposed algorithms with using the different density networks with the fixed number of nodes; we used 100 nodes.

Fig. 8 shows that in higher density networks (density 1.0), the localized algorithm starts outperforming the centralized algorithm, in terms of the energy expenditure of the critical node. Fig. 9 shows the average behavior of L-LIMAX and C-LIMAX in density 0.5 and density 1.0 networks. The white boxes indicate the results of the centralized algorithm, and the grey boxes show the results of the localized algorithm. The results of Fig.9 (upper) indicate that the centralized algorithm has overall better performance in different number of nodes with density 0.5 networks; however in the lower figure, it is apparent that the localized algorithm starts outperforming the centralized algorithm in dense networks. In Fig.10, the FR gaps between C-LIMAX and L-LIMAX are significant at density 0.5 networks (upper), but they become insignificant as the number of nodes increases at density 1.0 networks (lower).

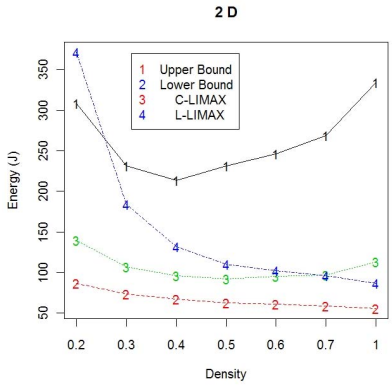

Figure 8. Avg. energy expenditure (J) vs. Density in two-dimensional networks with 100 nodes

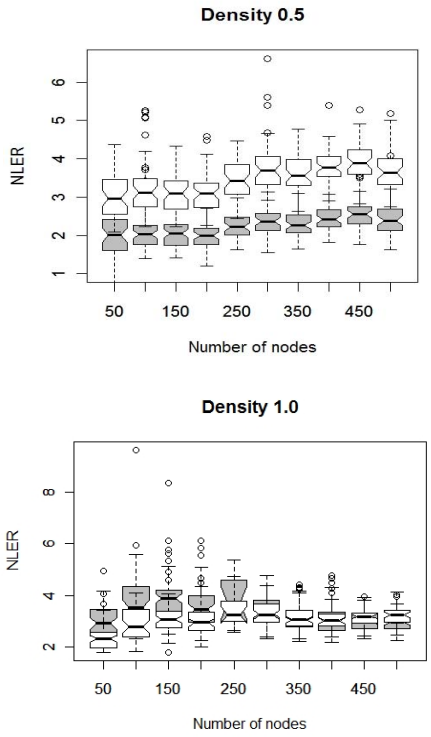

Figure 9. NLER vs. # of nodes for Density 0.5 (upper) and Density 1.0 (lower) with white boxes for centralized, grey boxes for localized

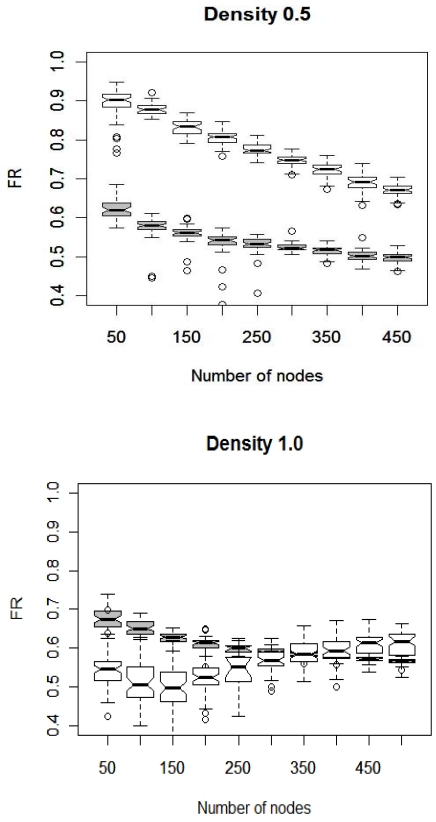

Figure 10. FR vs. the number of nodes for density 0.5 (upper) and density 1.0 (lower) with white boxes for centralized, grey boxes for localized

Recall that our centralized algorithm is executed at a single point with the availability of complete information about all measurements, and in the localized algorithm, a node uses only the information provided by its one-hop neighbors. The results from Fig. 8, 9 and 10 might indicate that local information with partial topology within one-hop neighborhood can amount to global knowledge in dense and large-scale networks. We believe that this finding is very important in developing future localized schemes and improving their performance.

VI. CONCLUSION

In this paper we have described two algorithms for the purpose of energy balancing: one for a localized algorithm and the other for a centralized algorithm. The localized scheme achieves energy balancing by exploiting the effective load sharing NAP and energy-efficient routing NAD methods. Our main contribution of the localized scheme is the integration of geographic information into energy awareness; it produces a simple but beneficial technique in load sharing/energy balancing areas. The results from our extensive simulations conducted on diverse networks and traffic models verify the effectiveness of our approach. Higher energy balancing is beneficial because the difference in energy consumption between nodes is reduced, which typically results in longer network life. Alternatively, the centralized algorithm approaches its objective using an iterative improvement method and repeatedly modifies current solutions. The results of the centralized algorithm are useful for the performance evaluation in developing the localized scheme. In comparison to this centralized approach, the proposed localized algorithm achieves considerable amounts of overall energy balancing.

Above all, one of the important findings is that the localized algorithm can compete with the centralized algorithm with global knowledge. This might indicate that local information with partial topology can amount to global knowledge in dense and large-scale networks.

Although many schemes have been proposed to achieve the energy savings at the link layer, by integrating link quality information into the routing layer our proposed schemes contribute to saving energy. Through cooperation with the link layer, our scheme will effectively perform the routing process. The simulation results indicate that incorporating our scheme provides a useful technique in energy balancing areas.

The main purpose of this work is to analyze the performance of the proposed scheme individually; several aspects for a complete localized routing protocol will be improved in the future. We also plan to implement the current schemes on test-beds to evaluate the performance in practice.

REFERENCES

[1] J. Chang and L. Tassiulas. “Energy conserving routing in wireless ad hoc networks”. In Proc. IEEE INFOCOM 2000, pages 22-31, 2000.

[2] A. Cerpa, J. L. Wong, L. Kuang, M. Potkonjak and D. Estrin “Statistical Model of Lossy Links in Wireless Sensor Networks”. IPSN’05

[3] A. Cerpa, J. L. Wong, M. Potkonjak and D. Estrin. “Temporal Properties of Low Power Wireless Links: Modeling and Implications on Multi-Hop Routing”. CENS Technical Report 0044, January 2005

[4] Oliver Kasten. Energy consumption Model. ETH-Zurich, Swiss Federal Institute of Technology

[5] D. S. J. De Couto, D. Aguayo, J. Bicket and R. Morris. “A High-Throughput Path Metric for Multi-Hop Wireless Routing”. ACM MobiCom, September 2003.

[6] Comen, Leiserson, and Rivest “Introduction to Algorithms” The MIT Press;2nd edition (September 2001)

[7] Discussed during CS259 class, instructed by Professor Potkonjak, winter 06