# Performance Evaluation of an Energy-Aware Routing Protocol for Sensor Networks

Moustafa A. Youssef*
Department of Computer Science
University of Maryland at College Park
A.V. William's Building
College Park, MD 20742

Mohamed F. Younis*
Dept. of Comp. Sc. and Elect. Eng.
University of Maryland Baltimore County
1000 Hilltop Circle
Baltimore, MD 21250

Khaled A. Arisha
Honeywell Intl. Inc.
Advanced Systems Tech. Group
7000 Columbia Gateway Drive
Columbia, MD 21046

***Abstract-*** *Networking unattended sensors is expected to have significant impact on the efficiency of many military and civil applications, such as combat field surveillance, security and disaster management. These systems process data gathered from multiple sensors to monitor events in an area of interest. Sensors in such systems are typically disposable and expected to last until their energy drains. Therefore, energy is a very scarce resource for such sensor systems and has to be managed wisely in order to extend the life of the sensors. In this paper, we study the performance of a novel energy-aware routing algorithm under different load conditions. The new algorithm has three main features: centralized routing decisions, using a multi-objective cost function, and using an accurate energy model for the nodes. We compare the new algorithm with traditional routing algorithms, which try to maximize the throughput or minimize the end-to-end delay, and with other energy-aware algorithms in a simulated environment for target-tracking applications. Network traffic is controlled by changing the target arrival rate, and thus used to study the performance of the algorithms under heavy load. The results show that the new algorithm performs well under different performance metrics and has the best predictability in terms of network lifetime under different load conditions.*

## 1 Introduction

Recent advances in miniaturization and low-power design have led to active research in large-scale, highly distributed systems of small-size, wireless unattended sensors. Each sensor is capable of detecting ambient conditions such as temperature, sound, or the presence of certain objects. Over the last few years, the design of sensor networks has gained increasing importance due to their potential for some civil and military applications. A network of sensors can be used to gather meteorological variables such as temperature and pressure. These measurements can be used in preparing forecasts or detecting natural phenomena. In disaster management situations such as fires, sensor networks can be used to selectively map the affected regions directing the nearest emergency response unit to the fire. In military situations, sensor networks can be used in surveillance missions and can be used to detect moving targets, chemical gases, or presence of micro-agents. Sensors in such environments are energy constrained and their batteries can not be recharged. Therefore, designing energy-aware algorithms becomes an important factor for extending the lifetime of sensors.

In wired networks, the emphasis of routing protocols has traditionally been on maximizing throughput and minimizing end-to-end delay. While wireless networks inherited such design metrics from the wired counterparts, the energy constraint has become a central issue [1],[2]. A number of energy-aware routing protocols have been introduced, for example [1]-[5], to minimize the energy consumption and increase the network lifetime. However, some applications are delay sensitive and an algorithm that tries to optimize the energy consumption alone may lead to unacceptable end-to-end delay. In this paper we report the performance of a new energy-aware and context-aware routing algorithm that tries to balance the requirements of minimizing the energy consumption and other performance metrics such as throughput and delay. We study the performance of this routing algorithm in a sensor network operating in a simulated target-tracking environment. We also compare the new algorithm with traditional routing algorithms, which try to maximize the throughput or minimize the end-to-end delay, and with other energy-aware algorithms. By increasing the target arrival rate, we study the performance of the algorithms under heavy load.

The system architecture for the sensor network is depicted in Fig. 1. In the architecture sensor nodes are grouped into clusters controlled by a single command node. Sensors are only capable of radio-based short-haul communication and are responsible for probing the environment to detect a target/event. Every cluster has a gateway node that manages sensors in the cluster. Clusters can be formed based on many criteria such as communication range, number and type of sensors and geographical location [12][13]. In this paper, we assume that sensor and gateway nodes are stationary and the gateway node is located within the communication range of all the sensors of its cluster. Clustering the sensor network is

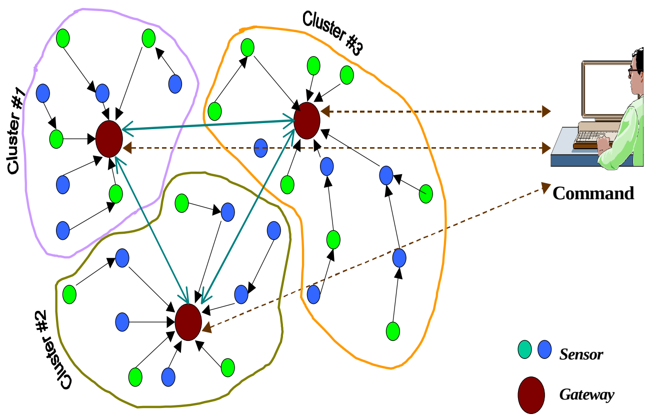


Fig. 1: Multi-gateway clustered sensor network

beyond the scope of this paper. While the gateway can be a single point of failure within the cluster, we plan to extend our approach to support recovery from a gateway failure.

Sensors receive commands from and send readings to its gateway node, which processes these readings. Gateways can track events or targets using readings from sensors in any clusters as deemed by the command node. However, sensors that belong to a particular cluster are only accessible via the gateway of that cluster. Therefore, a gateway should be able to route sensor data to other gateways. Our focus in this paper is the management of the sensor network within a cluster while inter-gateway communication is not addressed.

Gateway nodes, which are significantly less energy-constrained than the sensors, interface the command node with the sensor network via long-haul communication links. The gateway node sends to the command node reports generated through fusion of sensor readings, e.g. tracks of detected targets. The command node presents these reports to the user and performs system-level fusion of the collected reports for an overall situation awareness.

The sensor is assumed to be capable of operating in an active mode or a low-power stand-by mode. The sensing and processing circuits can be powered on and off. In addition, both the radio transmitter and receiver can be independently turned on and off and the transmission power can be programmed based on the required range. It is also assumed that the sensor can act as a relay to forward data from another sensor. The on-board clocks of both the sensors and gateways are assumed to be synchronized, e.g. via the use of GPS[1]. It is worth noting that most of these capabilities are available on some of the advanced sensors, e.g. the Acoustic Ballistic Module from SenTech Inc. [6].

The next section briefly describes the new energy-aware routing approach and the MAC layer protocol. Section 3 discusses the simulation environment and the performance analysis. Finally, section 4 concludes the paper and gives directions for future work.

## 2 Routing and MAC Layer Protocols

In this section, we summarize the new routing protocol and the MAC layer protocol. Detailed description and analysis of the algorithm can be found in [16].

### 2.1 Routing Protocol

Our routing protocol is based on three main features: centralized routing decisions, using a multi-objective cost function, and using an accurate energy model for the sensor nodes. The gateway periodically runs a centralized single-sink routing algorithm to determine the routes from sensor nodes to itself. Using a centralized routing algorithm fits the nature of the sensor networks: Since the sensor nodes are energy-constrained, it is more advantageous to move the routing decision from the sensor nodes to the less energy-constrained gateway. Moreover, the gateway has a global view of the cluster, which enables it to make more optimal routing decisions. Scalability concerns of the centralized routing algorithm are addressed by clustering. In addition, the gateway of the cluster will take charge of sensor organization and network management based on the mission and available energy in each sensor. Knowing which sensors need to be active in signal processing, the gateway can dynamically adapt the network topology within the cluster to minimize the energy consumed for communication, thus extending the life of the network while ensuring QoS for data transmission.

The nodes in a cluster can be in one of four main states: sensing only, relaying only, sensing-relaying, and inactive. Transitions among the different states are triggered by the gateway based on the current sensor organization, node battery levels, and desired network performance measures. based on the best organization for the application and on the most efficient network topology within the cluster. In the sensing state, the node's sensing circuitry is on and it sends data to the gateway with a constant rate. In the relaying state, the node does not sense the environment but its communications circuitry is on to relay the data from other active nodes. When a node is both sensing the environment and relaying messages from other nodes, it is considered in the sensing-relaying state. Otherwise, the node is considered inactive and can turn off its sensing and communication circuitry. It should be noted that our approach is transparent to the method for selecting the nodes that should sense the environment. Fig. 2 shows a typical cluster tasked with a target-tracking mission.

The typical operation of the network consists of two alternating cycles: data cycle and routing cycle. During the data cycle, the nodes sensing the target sends their data to the gateway, using other nodes as relays. During the routing cycle, the state of each node in the network is determined by the gateway and the nodes are then informed about their newly assigned states and how to route the data. These cycles are explained in the MAC protocol discussion (section 2.2).

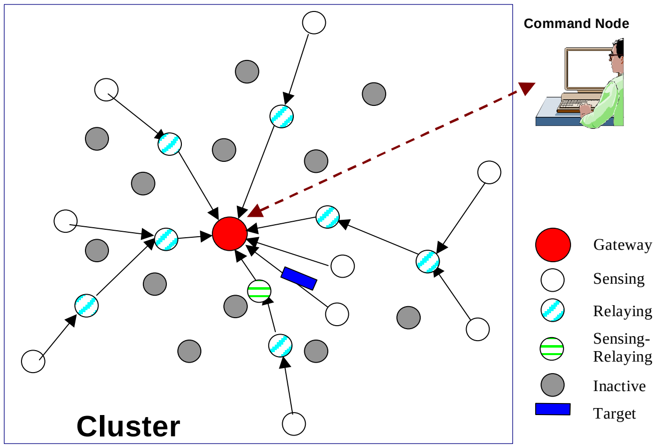


Fig. 2: A Typical Cluster in a Sensor Network

---

[1] While the GPS consumes significant energy, it has to be turned on for a very short duration during cluster formation. We use time-based approach for media access control that enables the maintenance of clock synchronization afterward.

### 2.1.1 Multi-Objective Cost Function

The cost function used in the routing algorithm takes into account the transmission energy, remaining battery level in each sensor using an accurate energy model, propagation delay, queuing delay, and cost for changing the state of a node. The routing problem is defined as minimization of the following cost function:

$$\sum_{k=0}^{7} CF_k = c_0 \times (distance_{ij})^l + c_1 \times f(energy_j) + c_2 \times (MaxTime - T_j) + c_3 + c_4 + c_5 + c_6 \times distance_{ij} + c_7 \times overall\ load$$

Where $distance_{ij}$ is the distance between nodes $i$ and $j$, $energy_j$ is the current energy of node $j$, and $CF_k$ are cost factors defined as follows:

- $CF_0$: Communication cost = $c_0 \times (distance_{ij})^l$, $c_0$ is some constant. This factor reflects the cost of the wireless transmission power, which is directly proportional to the distance raised to some power $l$. Value of the parameter $l$ depends on the nature of the environment. The closer a node to the destination, the less its cost factor $CF_0$, the more attractive it is for routing.
- $CF_1$: Energy amount = $c_1 \times f(energy_j)$ for node $j$. This cost factor favors the nodes with more energy. The nodes with abundant energy are expected to last long and, thus increase the reliability to the paths selected for routing. The function '*f*' is chosen to reflect a battery's remaining lifetime.
- $CF_2$: Energy consumption rate = $c_2 \times (MaxTime - T_j)$, where $T_j$ is the expected time for the current consumption rate till its energy amount hits the minimum acceptable threshold for a routing node $j$. The $CF_2$ increases when the node is close to die. It makes the over-used nodes less attractive, even if they have a lot of energy, still being overloaded will make them die soon enough to make the paths going through them less reliable and having shorter lifetime.
- $CF_3$: Relaying enabling cost = $c_3$, the overhead required to enable a node to be a relay from being inactive. The cost is added to all nodes at the beginning and subtracted when a node is enabled as a relay. This factor allows balancing the desire to offload current relay nodes by utilizing inactive nodes and the conserving energy on the cluster level.
- $CF_4$: Sensing enabling cost = $c_4$, the amount of overhead added when activating the signal processing circuit on the sensor. This factor favors less the sensing-enabled nodes to be used as relays, since these nodes already consume some energy in their sensing circuits. We prefer not to overload these nodes with relaying since they are critical for the functionality of the sensor network, therefore it is better to keep these sensing enabled nodes live as long as possible.
- $CF_5$: Maximum connections per router: once this threshold is reached, we add an extra cost $c_5$ to avoid more paths to exceed limit of connections per relay. This factor helps to favor less the overloaded relay-enabled nodes in order to extend their lifetimes, since they are already critical by being on more than one path.
- $CF_6$: = $c_6 \times distance_{ij}$, $c_6$ is the speed of wireless transmission. The farther the node from the gateway, the more delay it takes the wireless communication to transmit a message from it. So, this factor favors closer nodes in distance.
- $CF_7$: Queuing Cost = $c_7 \times \lambda / \mu$, where $\lambda = \Sigma \lambda_i$, where $\lambda_i$ is the packet arrival rate for each sensor node $i$ whose route passes through the node $j$, and $\mu$ is the service rate (mainly store-and-forward process). The expression $\lambda / \mu$ is the average queue length for the M/M/1 queuing model. This factor can be mathematically simplified to be the overall load to the relay node, where the overall load is the total number of sensing-enabled nodes whose data messages are sent via routes through this node to the gateway. Assuming equal service rate $\mu$ for each relay as well as equal data sensing rate for each sensing-enabled node, the constant $\mu$ can be reduced inside $c_7$, and $\lambda$ can be reduced to the overall load times the constant data sensing rate for each sensor. Thus $CF_7 = c_7 \times overall\ load$. This factor favors less the relays with longer queue-lengths to avoid dropping or delaying data packets.

It's clear that many of these factors are contradicting. Choosing a specific value for each of the parameters depends on the nature of the sensor network mission and on the required performance parameters. The above model is a typical path-optimization routing problem, which can be solved using a shortest path (least-cost) algorithm [14]. Our routing problem can be considered as the transpose of the one-to-some shortest path. We use Dijkstra's algorithm, which is shown to suite one-to-some shortest path problems [15].

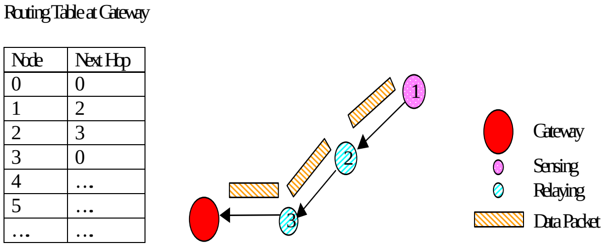
### 2.1.2 Energy Model for the Sensor Nodes

A model-based energy consumption for the data processor, radio transmitter and receiver to track the life of the sensor battery is used in the routing cost function. For the data processor data model, we used the leakage current model[5][7][8][9]. In our communication energy consumption model, the key energy parameters for communication are the energy/bit consumed by the transmitter electronics ($\alpha_{11}$), energy dissipated in the transmit op-amp ($\alpha_2$), and energy/bit consumed by the receiver electronics ($\alpha_{12}$). Assuming a $1/d^n$ path loss, the energy consumed thus becomes:

$$E_{tx} = (\alpha_{11} + \alpha_2 d^n) * r \qquad and \qquad E_{rx} = \alpha_{12} * r$$

Where $E_{tx}$ is the energy consumed to send '*r*' bits and $E_{rx}$ is the energy consumed to receive '*r*' bits [3] [5][9].

The gateway updates the sensor energy model with each packet received by changing the remaining battery capacity for the nodes along the path from the initiating sensor node to the gateway. Fig. 3


Routing Table at Gateway

| Node | Next Hop |
|---|---|
| 0 | 0 |
| 1 | 2 |
| 2 | 3 |
| 3 | 0 |
| 4 | … |
| 5 | … |
| … | … |

Fig. 3: When the gateway receives a packet from node1, it uses the routing table to update the energy model of nodes 1, 2, and 3, which are on the path from node1 to the gateway

shows an example.

The energy model can deviate from the actual node battery level due to inaccuracy in the model, packet drop due to communication error, or packet drop due to buffer overflow at a node. This deviation may lead to inaccuracy in the routing decisions, which may affect the network performance. To compensate for this deviation, the nodes refresh the gateway energy model periodically at a low frequency. All nodes, including inactive nodes, send their refresh packets at a pre-specified time and then turn their receivers on at a predetermined time in order to hear the gateway's routing decision. If a node's refresh packet is dropped due to communication error, the gateway assumes that the node is nonfunctioning during the next cycle, which leads to turning this node off. The next section summarizes our MAC layer protocol.

### 2.2 MAC Layer Protocol

Although our routing protocol is independent of the MAC layer protocol, we choose to implement a time division multiple access (TDMA) based MAC layer whose slot assignment is managed by the gateway. The gateway informs each node about the slots in which it should listen to other nodes' transmission and the slots that the node can use for its own transmission. The TDMA MAC layer provides two features that are advantageous to our approach. First, clock synchronization is built in the TDMA protocol. Second, collision among the nodes can be avoided with assigning non-overlapping time slots.

To set the routes, the gateway sends to each sensing node the transmission range to cover so that data can reach the next relay node on the route. In addition, the gateway sends a forwarding table to relay nodes. The forwarding table consists of ordered tuples of the form: (time slot, data-originating node, transmission range). The "time slot" entry specifies when to turn on the receiver in order to listen for an incoming packet. The "source node" is the sensor node that originated this data packet, and the "transmission range" specifies the transmission power to use in sending the data. This transmission power should be enough to reach the next relay on the path from the originating node to the gateway. The transmission range ensures that the next relay node, which is also told to forward that data packet, can clearly receive the data packet.

The protocol consists of four main phases: data transfer, refresh, event-triggered rerouting, and refresh-based rerouting phase. Fig. 4 shows an example of a typical sequence of phases. In the data transfer phase, sensors send their data in the time slots allocated to them. Relays use their forwarding tables to forward this data towards the gateway. Inactive sensor nodes remain off until the time for sending a status update or to receive route broadcast messages. The refresh phase is designated for updating the sensor model at the gateway. This phase is periodic and occurs after multiple data transfer phases.

Two phases are designated for rerouting and scheduled at different frequencies. The first phase is called event-based rerouting and allows the gateway to react to changes in the sensor organization and to drops in the available energy of one of the relay sensors below a preset acceptance level. The second rerouting phase occurs immediately after the refresh phase terminates. During both phases, the gateway runs the routing algorithm and sends new routes to each node in its pre-assigned slot number and informs each sensor about its new state and slot numbers.

## 3 Performance Evaluation

We present here performance evaluation for the routing algorithm in a simulated target-tracking environment, with emphasis for heavy load conditions. This section presents the performance metrics, the simulation environment, and performance evaluation results of the new algorithm compared with other routing algorithms.

### 3.1 Performance Metrics Phase

We use the following metrics to capture the performance of our routing approach and to compare it with other algorithms:

- *Time to network partition*: When the first node runs out of energy, the network is said to be partitioned [1].
- *Time for last node to die*: This metric, along with the time to network partition metric, gives an indication of network lifetime.
- *Average and standard deviation of nodes' lifetime*: This also gives a good measure of the network lifetime. A routing algorithm, which minimizes the standard deviation of node lifetime, is predictable and thus desirable.
- *Average delay per packet*: Defined as the average time that a packet sent by a node in the sensing state takes to reach the gateway. Most energy aware routing algorithms try to minimize the consumed energy. However, some sensor network missions are delay sensitive, so this metric is important.
- *Average energy consumed per packet*: This metric represents the average energy consumed in transmitting, and receiving a data packet. A routing algorithm that minimizes the energy consumed per packet will, in general, yields better energy savings and increased network lifetime.
- *Network Throughput*: Defined as the rate of data packets received at the gateway.

### 3.2 Environment Setup

In the experiments the cluster consists of 100 randomly placed

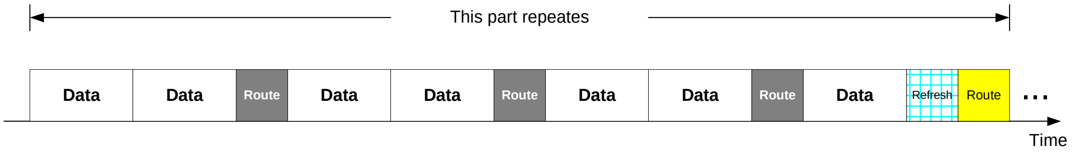


Fig. 4: MAC Protocol Time-Based Phases

nodes in a 1000×1000 meter square area. The gateway position is determined randomly within the cluster boundaries. A free space propagation channel model [11] is assumed with data rate set to 2Mb/s. Packet lengths are 10 kbit for data packets and 2 kbit for routing and refresh packets. The buffer size at each node is 15 packets. Each node has an initial energy of 5 joules. A node is considered non-functional if its energy level reaches 0.

For a node in the sensing state, packets are generated at a constant rate of 1 packet/sec. This value is consistent with the specifications of the Acoustic Ballistic Module from SenTech Inc.[6]. Each data packet is time-stamped when it is generated to allow the calculation of average delay per packet. In addition, each packet has an energy field that is updated during the packet transmission to calculate the average energy consumed per packet. A packet drop probability is taken equal to 0.01. This is used to make the simulator more realistic and to simulate the deviation of the gateway energy model from the actual energy model of nodes.

We assume that the cluster is tasked with a target-tracking mission in the experiment. The initial set of sensing nodes is chosen to be the nodes on the convex hull of the sensors of the cluster. The set of sensing nodes change as targets move. Since targets are assumed to come from outside the cluster, the sensing circuitry of all boundary nodes is always turned on. The sensing circuitry of other nodes are usually turned off but can be turned on according to targets movement.

Targets are assumed to start at a random position outside the convex hull. We experimented with different types of targets but for this paper we choose the linearly moving targets. These targets are characterized by having a constant speed chosen uniformly from the range four m/s to six m/s and a constant direction chosen uniformly depending on the initial target position in order for the target to cross the convex hull region.

Targets arrive in the deployment area according to a Poisson arrival process. The average inter-arrival time is chosen such that the average number of targets per unit time ranges from 1 to 16. Each target remains active until it leaves the deployment region area.

### *3.3 Performance Results*

In this section we present some results obtained by simulation. For the purpose of our simulation experiments the values for the parameters $\{c_i\}$ are initially picked based on sub-optimal heuristics for best possible performance. The performance of the new algorithm is compared with the following routing algorithms:

- *Direct routing algorithm*: In this algorithm, each node sends its data directly to the gateway [3].
- *Minimum transmission energy routing algorithm*: This algorithm chooses the intermediate nodes such that the transmit amplifier energy is minimized. The chosen cost function tries to minimize the sum of the distance squared between a node and gateway [3].
- *Linear battery*: This routing algorithm chooses the paths such that nodes with depleted energy reserves do not lie on many paths. In this routing algorithm, the node remaining lifetime is taken to be a linear function of its remaining energy, which is the normal behavior of some alkaline batteries [1].

Figures 5 through 11 summarize the comparative results. We can see from the figures that some algorithms, such as the minimum transmission energy routing algorithm fails to work at high targets arrival rates as the number of time slots becomes inadequate. This can be explained by noticing that the minimum transmission energy routing algorithm tries to minimize the transmission energy by taking short distances leading to more hops and thus more relays. Each relay requires a number of time slots for transmitting its own data. As the number of targets increases, the number of slots required becomes more than the number of available slots and thus the algorithm fails. It is worth mentioning here that the minimum transmission energy routing algorithm may still work under a different MAC layer protocol. However, choosing a contention-based MAC layer protocols may consume more energy due to contention and collisions. The linear battery routing algorithm ensures that the shortest-hop routing will be used when the network starts operation but as the network nodes approach the end of their lifetimes, the packets are routed so that no node dies before the others. This explains the similarity in performance between the direct routing algorithm and the linear battery routing algorithm especially in figures 5, 7, and 11.

Figure 5 shows that regardless of the minimum transmission energy routing algorithm, which fails at high target arrival rate, the new algorithm gives the best time for network partitioning. This is expected, as the new algorithm is the only algorithm of the remaining algorithms that takes energy consumption into consideration. At high load, the new algorithm gives an order of magnitude enhancement over the other algorithms.

Figures 6 and 7 show the time for the last node to die and the average node lifetime respectively. The curves show that the new algorithm performs well under low and high target arrival rates. However these curves alone may be misleading without looking at Fig. 8, which shows the standard deviation of the nodes lifetime. The figure shows that the new algorithm gives the best standard deviation after the minimum transmission energy algorithm, which an indication of the good predictability of the performance of the new algorithm. Under high load, the new algorithm is the most predictable with 11% enhancement over the other algorithms.

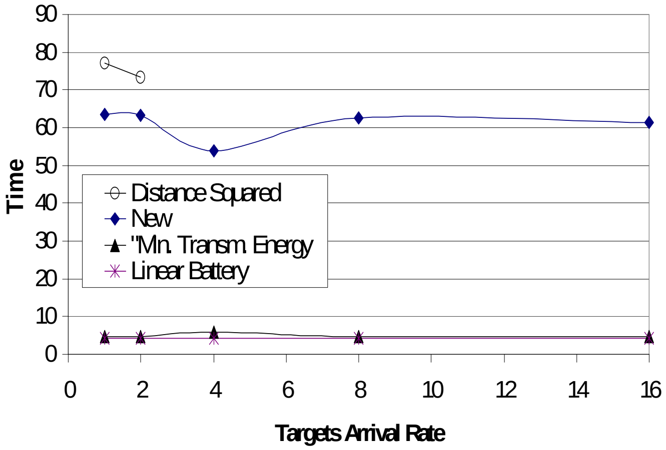


Fig. 5: Comparison between different routing algorithms (Time to network partition)

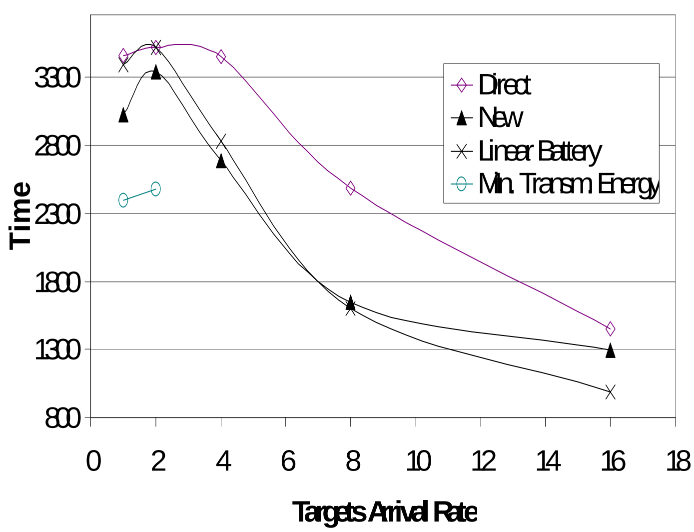


Fig. 6: Comparison between different routing algorithms (Time for last node to die)

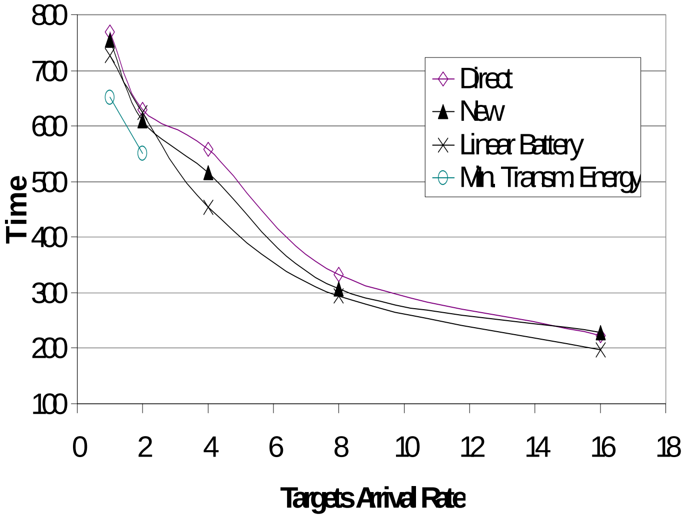


Fig. 7: Comparison between different routing algorithms (Average node lifetime)

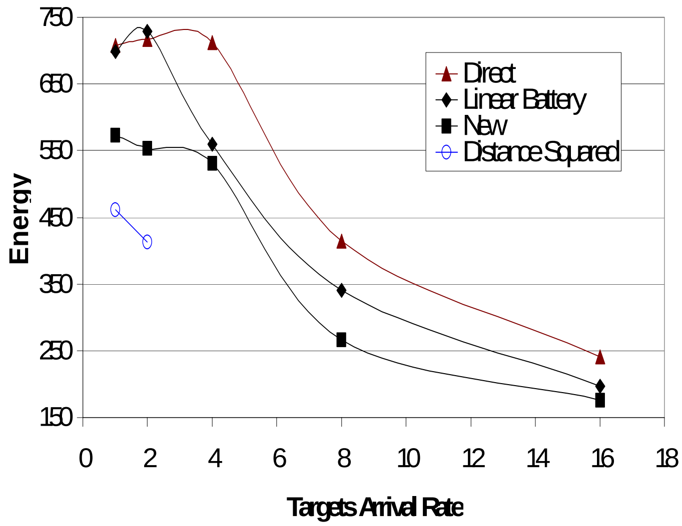


Fig. 8: Comparison between different routing algorithms (Standard deviation of nodes lifetime)

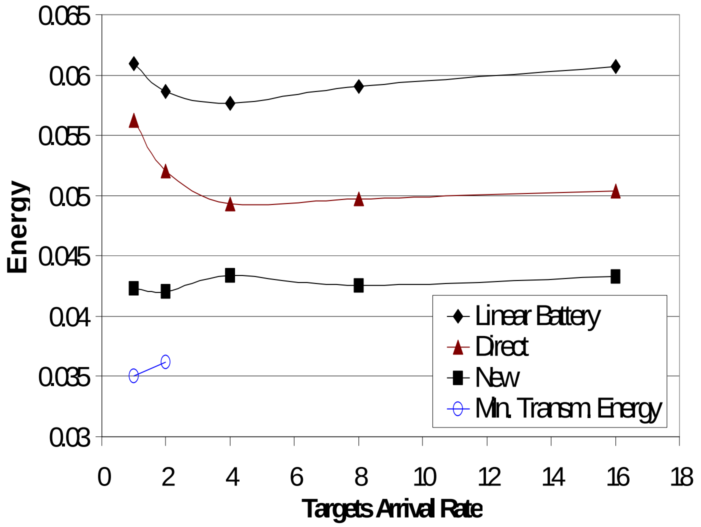


Fig. 9: Comparison between different routing algorithms (Avg. energy consumed per packet)

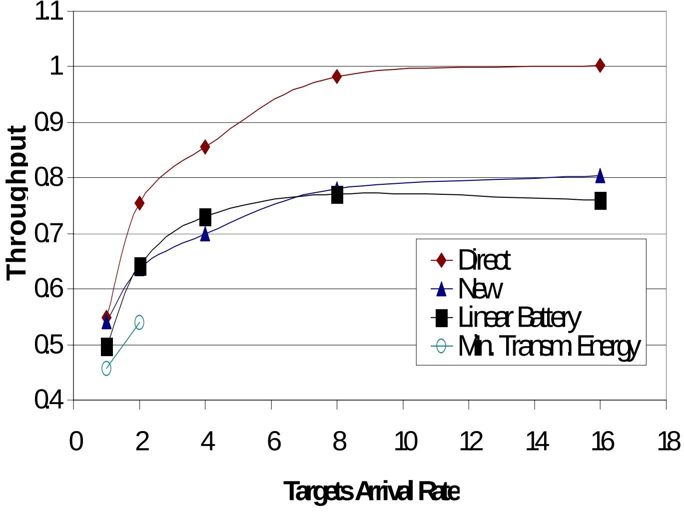


Fig. 10: Comparison between different routing algorithms (Network Throughput)

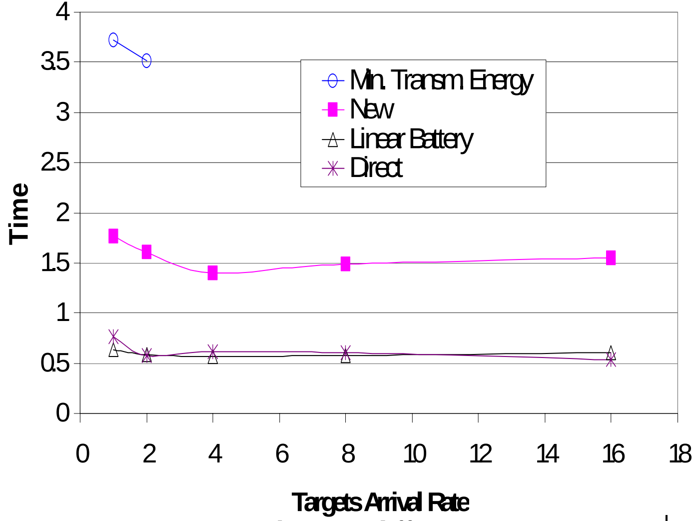


Fig. 11: Comparison between different routing algorithms (Average delay per packet)

Figure 9 shows the average energy consumed per packet. The figure shows that the new algorithm's performance is consistent under different target arrival rates. Moreover, under heavy load, the new algorithm gives the best average energy consumed per packet with a 14% enhancement. This is expected as the new algorithm tries to minimize the energy consumption while other algorithms either fail to work or do not take energy consumption into consideration in the routing decision. Although the linear battery routing algorithm tries to conserve each node's battery, it does not try to reduce the energy consumed per packet. For example, the linear battery routing algorithm may choose a far away node with a large remaining battery level over a near node with moderate energy level leading to a large amount of transmission energy per packet.

The network throughput is shown in Fig. 10. The new algorithm's performance is accepted under different target arrival rates. The best throughput is achieved using the direct routing algorithm as it gives the minimum average delay per packet as shown in Fig. 11. However the nodes do not stay long under direct routing even under light load, as previously concluded from figures 5-7.

Figure 11 shows the average delay per packet for the different routing algorithms. The figure shows that the new algorithm performance is also consistent under different target arrival rates. The best average delay per packet is achieved by using the direct routing algorithm while the worst average delay per packet is achieved when the minimum transmission energy routing algorithm is used. Again, the minimum transmission energy routing algorithm tries to minimize the transmission power by taking short distances and larger number of hops leading to increased delay. The opposite reasoning is applied to the energy consumed per packet shown in Fig. 9.

The above results show that the new algorithm gives a relatively good performance for all the metrics. Other algorithms may slightly outperform our algorithm in some metrics. However, the same algorithms perform poorly on other metrics. Moreover, under heavy load, the new algorithm gives the best values in terms of time to network partitioning (with an order of magnitude enhancement), predictability (with 11% enhancement), and average energy consumed per packet (with 14% enhancement).

# 4 Conclusions and Future Work

In this paper, we studied the performance of a new energy-aware routing protocol under heavy load conditions. The new algorithm tries to balance different QoS requirements by using a multi-objective cost function weighting different energy and delay requirements. The results show that the new algorithm gives a relatively good performance for all the performance metrics. Under heavy load, the new algorithm gives an order of magnitude enhancement in time to network partitioning, 11% enhancement in predictability (standard deviation of node's lifetime), and 14% enhancement in average energy consumed per packet whereas other routing algorithm may completely fail to work.

We are currently investigating efficient algorithms for dynamic slot assignment in our TDMA-based MAC protocol. Our future plan includes extending the routing model to handle QoS constraints and to allow for node mobility. We would like also to study network clustering approaches, inter-cluster interaction and operations, the management of resources at the cluster level and the handling of sensor or gateway failure.

# REFERENCES

[1] S. Singh, M. Woo and C. S. Raghavendra, "Power-aware routing in mobile ad hoc networks", ACM Mobicom'98, Dallas, Texas, Oct. 1998.

[2] C-K. Toh, "Maximum battery life routing to support ubiquitous mobile computing in wireless ad hoc networks," IEEE Comm. Mag., Jun. 2001.

[3] W. Heinzelman, A. Chandrakasan, and H. Balakrishnan, "Energy-Efficient Communication Protocols for Wireless Microsensor Networks," Hawaii International Conference on System Sciences (HICSS '00), January 2000.

[4] J.-H. Chang, L. Tassiulas,“ Energy conserving routing in wireless ad-hoc networks”, IEEE Infocom'00, 2000.

[5] W. Heinzelman, A. Sinha, A. Wang, and A. Chandrakasan, "Energy-Scalable Algorithms and Protocols for Wireless Microsensor Networks," Proc. International Conference on Acoustics, Speech, and Signal Processing (ICASSP '00), June 2000.

[6] "Data sheet for the Acoustic Ballistic Module", SenTech Inc., http://www.sentech-acoustic.com/

[7] R. Min, M. Bhardwaj, S. Cho, A. Sinha, E. Shih, A. Wang, and A. Chandrakasan, "An Architecture for a Power-Aware Distributed Microsensor Node", IEEE Workshop on Signal Processing Systems (SiPS '00), October 2000.

[8] A. Sinha and A. Chandrakasan, “Energy Aware Software”, Proceedings of the 13th International Conference on VLSI Design, pp. 50-55, Calcutta, India. January 2000.

[9] A. Sinha and A. Chandrakasan, "JouleTrack - A Web Based Tool For Software Energy Profiling", Proceedings of the 38th Design Automation Conference, June 2001.

[10] M. Bhardwaj, T. Garnett, and A. P. Chandrakasan, "Upper bounds on the lifetime of sensor networks", Proc. ICC 2001, Jun. 2001.

[11] J.B. Andresen, T.S. Rappaport, and S. Yoshida, “Propagation measurements and models for wireless communications channels,” IEEE Comm. Mag., vol. 33, no. 1, Jan. 1995.

[12] A. Buczak and V. Jamalabad, "Self-organization of a Heterogeneous Sensor Network by Genetic Algorithms," Intelligent Engineering Systems Through Artificial Neural Networks, C.H. Dagli, et. a.. (eds.), Vol. 8, pp. 259-264, ASME Press, New York, 1998.

[13] Chunhung Richard Lin and Mario Gerla, "Adaptive Clustering for Mobile Wireless Networks," IEEE Journal on Selected Areas in Communications, Vol. 15, No. 7, September 1997.

[14] M. H. Hung and J. J. Divoky, “A Computational Study of Efficient Shortest Path Algorithms,” Computers & Operations Research, Vol. 15, pp. 567-576, 1988.

[15] F. B. Zhan, and C. E. Noon, “Shortest Path Algorithms: An Evaluation Using Real Road Networks,” Transportation Science, 1996.

[16] M. Younis, M. Youssef, K. Arisha, “Energy-Aware Routing in Cluster-Based Sensor Networks”, under submission.