World Applied Sciences Journal 14 (2): 356-362, 2011
ISSN 1818-4952

Energy-Efficient and Real Time Routing Protocol for Wireless Sensor Networks

[1]*Ali Ghaffari*, [2]*Hanieh Rahbari Bannaeian* and [3]*Davood Keykhosravi*

[1,2]Department of Computer Engineering, Tabriz Branch, Islamic Azad University, Tabriz, Iran
[3]Department of Computer Engineering, Osku Branch, Islamic Azad University, Osku, Iran

Abstract: Real time and energy efficient data forwarding are the two important requirements for critical-mission applications in wireless sensor networks. Data forwarding with delay is not acceptable for mission-critical application (automatic battlefield monitoring, forest fire detection and nuclear plant monitoring) in wireless sensor networks. In this paper, we propose a new energy-efficient and real time (ERR) routing protocol for wireless sensor networks, called ERR. We use a new cost function for selecting the next optimum neighbor nodes. ERR considers the node residual energy, link energy cost and node buffer occupancy levels to dynamically select the best neighbor node from the set of available one-hop neighbor nodes. The proposed protocol provides a differentiated routing service by assigning priorities to time-sensitive packet according to their criticality by using three level priority based queues in intermediate nodes. Simulation results show that the proposed method can achieve real time data forwarding with energy-efficiency, low missed-deadline data delivery, low end-to-end latency and extremely low control signal overhead.

Key words: Wireless sensor networks · Multi-level queue · Geographical routing · Energy-efficent · Priority level · Timeliness

INTRODUCTION

A wireless sensor networks (WSNs) consists of large number of small sensor nodes that run on battery power, have limited memory and processing power [1]. End-to-end real-time communications are important for WSNs to achieve the collaborative sensing task with specific timing constraints and prone to failure sensor nodes. Consequently, applications such as battlefield surveillance, industrial production control, real-time target tracking, environmental monitoring and emergency response require certain levels of real time for on time event reporting.

According to the event importance and urgency, the data packets associated with different events can be assigned different end-to-end deadline requirements. Only the packets that are delivered to the base station node before expiration of the deadline are useful [2]. On the other hand, energy consumption is one of the most important concerns in routing protocol for prolonging the network lifetime. In this paper, a new packet delivery method for real time data with localized geographic forwarding that uses one-hop neighbor information (ERR) is proposed. ERR is scalable and self-adaptable to network dynamics and decreases the overhead of route discovery and maintenance. Our goal is to provide end-to-end real time provisioning while prolonging the lifetime of networks via load balancing and local decisions at each relay node without end-to-end route discovery and route saving in each intermediate node. In real-time domains, each node assigns proper priority levels to the data packets depending on the information urgency and packet remaining deadline. At each relay node, a three-level queue scheduler is used to provide the service to the packets, according to their priority levels. ERR computes optimal forwarding nodes based on parameters such as residual energy, occupied buffer and required energy for packet transmission.

The rest of the paper is organized thus: section 2 will present related work. A detailed description of the proposed routing algorithm including the one-hop neighbor table creating and updating are explained in section 3. In section 4, the detailed operation of packet dropping policy and packet deadline updating in intermediate nodes are provided. Section 5 describes the multi-level queuing model of sensor nodes. The simulation model and the comparative performance evaluation of the proposed schemes are presented in section 6. Section 7 concludes the paper.

Corresponding Author: Ali Ghaffari, Department of Computer Engineering, Tabriz Branch, Islamic Azad University, Tabriz, Iran. Tel: +98-411-3317068; Fax: +98-411-3328560.

Related Works: RAP [3] uses geographical forwarding and provides service differentiation in the timelines domain by a new packet scheduling policy called velocity-monotonic scheduling of packets. RAP schedules packets on the basis of their required transmission speed. Velocity-aware scheduling ensures that packets meet their deadlines by giving higher priority to packets with higher requested velocity. However, for path calculating, RAP fails to consider energy of nodes. Real-time power-aware routing (RPAR) protocol [4] fulfilled the end-to-end real-time requirements with low energy consumption by dynamically adapting transmission power to the workload.

SPEED [5] is a stateless protocol designed to provide soft end-to-end deadline guarantees for real-time packets in WSNs. Using the delay and distance, each node computes the packet progress speed of each neighboring node and forwards a packet to a neighborhood node with a progressive speed that is higher than the pre-specified lower-bound speed. When a node cannot find any neighbor node with a speed that is higher than set-speed, it probabilistically drops packets to save energy consumption and regulate the workload so that at least one neighbor node with a speed higher than set-speed exists at all times. At the same time, the node sends a backpressure packet to the previous nodes to keep them from forwarding any more packets through the congested area. One of the drawbacks of SPEED is that it does not have a packet prioritization method.

The multi-path and multi-speed routing protocol called MMSPEED [6] is a novel packet delivery mechanism that takes into accounts both timeliness and reliability for path selection. For timeliness domain, multiple QoS (Quality of Service) levels are supported by providing multiple packet delivery speed guarantees. Intermediate nodes can boost a packet's transmission speed if they notice that the packet may not meet its delay deadline at the current speed. However, MMSPEED fails to consider energy issues.

Akkaya and Younis [7] proposed an energy-aware routing protocol for real-time traffic generated by WSNs. The path cost is a function of node residual energy, energy transmission, distance between nodes and error rates. All traffic is divided into real-time and non-real-time classes. Real-time Routing Protocol with Load Distribution (RTLD) [8] considers a real-time routing protocol. RTLD take into account link quality, packet delay and remaining power of the next hop nodes. This protocol uses geo-directional forwarding. In [13], the authors proposed a new QoS-based routing protocol for wireless sensor networks that improves the reliability and timeliness in packet forwarding and prolongs the network lifetime. In [14], a three-dimensional sensor placement as well as non-uniform distribution is used in order to have a more realistic simulation of real life deployment environment. To study the interconnection effect between sensors, two methodologies were used. The first is the classical uniform generation of the adjacency matrix as being used by previous researchers on this problem. The second, the authors introduce the generation of adjacency matrix based on Partial Order Set (POSET).

Proposed Protocol

Network Model and Assumption: We assume that nodes are stationary or have low mobility, with communication range Rc. Sensor nodes are scattered randomly, in a uniform distribution .We assume nodes are aware of their positions, either through a global positioning system (GPS) [9] or any other localization protocols [10, 11]. Thus, the Euclidean distance between two neighbor nodes (i and j) can be calculated by:

$$d(i,j)=\sqrt{(x_i-x_j)^2+(y_i-y_j)^2} \qquad (1)$$

The neighbor set of node i is the set of nodes within the radio range of node i and can be defined as:

$$NS_i=\{j \mid d(i,j)\le R_c\} \qquad (2)$$

Where the R_c is node radio range and $d(i,j)$ is Euclidean distance between two neighbor nodes (i, j). The forward area of node i is the area in the transmission range of node i that provides progress toward the sink node (Figure 1).

To obtain enough information to construct and maintain one-hop neighbor table, each sensor node broadcasts a HELLO message to its one-hop neighboring nodes from a forward area node set.

Neighbor Table Construction and Updating: In ERR, as in all geographic routing protocols, each sensor nodes knows its one-hop neighboring nodes and their current parameters. Neighboring nodes use HELLO packets to create a neighbor table (NT) that assigns an entry for each one-hop neighboring node, which includes its position, node identification (node ID), energy cost, used buffer level and residual energy updating. One-hope neighbor nodes receive HELLO message and, by comparing their geographical position with the source node, decide to reply to the HELLO message response if the neighboring node is closer to the sink node than it is. In this paper, we have an initial condition; we choose a threshold for residual energy and every node having more energy than this threshold will be permitted to add in the neighbor table. The format of HELLO packet is shown in Figure 2.

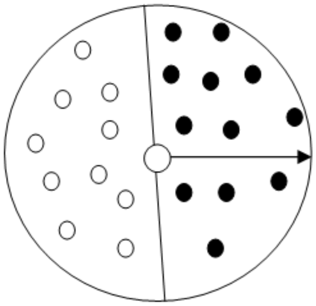

Fig. 1: Candidate Nodes for Data Forwarding

NodeId	Position	Residual Energy	Buffer Used	Energ y cost

Fig. 2: Packet Structure of HELLO Packet Response

Upon receiving the HELLO packet response, each node keeps a neighbor table to store information of neighbor nodes passed by the HELLO packets and satisfying the forwarding condition.

To achieve soft real-time guarantees, we can calculate the desired velocity for data packets by dividing the advance in distance from the next node *j* by the average delay:

$$Speed_i^j = \frac{progress\ distance\ to\ sink}{Avg-delay(i,j)} = \frac{dist(i,sink)-dist(j,sink)}{(RTT/2)} \qquad (3)$$

Where RTT is packet round-trip time. One of the most important phases in ERR is maintenance and neighbor table updating due to node mobility and link failure. High periods of HELLO packets exchange consume network resources. Consequently, this period should be carefully selected to maintain proper trade-offs between resource consumption and information freshness. When we select some nodes to send packets to, we consider a timer in each node. These timers decide to send acknowledge packets to the upstream node or not. By broadcasting this control packet to all one hop neighbors, if the sender node exists in their table, all routing tables will update. One of the most important pieces of information that can obtain from an acknowledgment signal is residual energy. When the energy level of a node reaches less than our energy threshold, the node will be deleted rapidly from the node routing table.

Cost Function: To select the next node among the alternative neighbors providing positive advancement towards the sink, we use the lowest cost nodes from the neighbor table. The cost function is computed as:

$$\mathrm{Cost}_i^j = \alpha(\mathrm{B_{used}}(j)) - \beta(\mathrm{E_{cost}}(i,j)) - \gamma(\mathrm{E_{res}}(j)) \qquad (4)$$

Where $E_{cost}(i, j)$ is required energy for sending a packet from node i to node j directly. $B_{used}(j)$ is the used buffer level of j node. $LQ(i\ j)$ is the link quality between (i, j), or packet reception rate, is considered to improve the packet delivery ratio. E_{res} is residual energy of the next node. In (3) α, β, γ and δ are weight factors and α,+ β,+ γ. Based on these values, ERR selects a path that satisfies packet required reliability and that requires low energy and low delay. Link energy cost can be obtained as follows:

$$E_{total}(k) = E_{TX}(k,d) + E_{RX}(k) = \begin{cases} k(2E_{elec} + \varepsilon_{Fs}d^2), & d < d_0 \\ k(2E_{elec} + \varepsilon_{mp}d^4), & d \geq d_0 \end{cases} \qquad (5)$$

In a real-time domain, we need a hop count from source node to sink. As mentioned before, all sensor nodes are location-aware, so we can estimate hop count from source node to the sink as follows:

$$h_i^{Sink} = \frac{dist(i,sink)}{avg[dist(i,j)]} \qquad (6)$$

Where h_i^{sink} is hop count estimation from node i to the sink and *dist* (*i, sink*) is the distance between node i and sink. Thus, with localized information, we can calculate the required delay for event reporting packet:

$$Deadline_i^{Sink}(x) = h_i^{Sink} \times Max[Delay_i^j] \qquad (7)$$

We attach this deadline to the event reporting packet.

ERR selects the most desired velocity for a packet x based on the distance to sink and end-to-end deadline as follows:

$$Desired\ Speed(x) = \frac{dist(i,sink)}{deadline(x)} \qquad (8)$$

We must choose a neighbor node from a real-time table, the speed of which is equal to or higher than the desired speed.

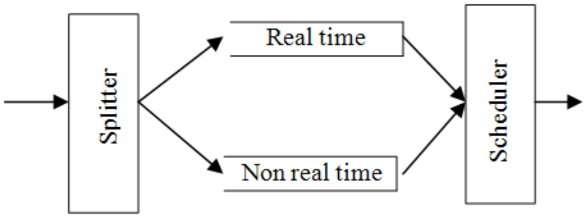

Fig. 3: Multi-level Queuing Model for ERRg4

Packet Deadline Updating: By transferring a packet through the network, we can update its deadline. While reaching a packet to a intermediate node, the media access control (MAC) layer adds a timestamp to the packet. Then the received packet is forwarded to the chosen next hop node via the MAC layer and waiting queue. So we can compute elapsed time by subtracting the transmission time from the arrival time. For real-time packets, each intermediate node can update the remaining time to deadline as follows:

$$Deadline_x^{new} = Deadline_x^{old} - t_{elapsed} \tag{9}$$

Where $Deadline_x^{new}$ is the new remaining time to deadline of packet x and $T_{elapsed}$ is the sum of the delay from the source node to the current intermediate node. $T_{elapsed}$ is calculated as:

$$T_{elapsed} = t_{MAC} + t_{proc} + t_{prop} + t_{tran} + t_{queue} \tag{10}$$

Where t_{MAC} is the time it takes for the sensor node to obtain the channel. t_{proc} is the processing time of the packet in the intermediate node before forwarding it to the next neighbor node. t_{prop} is the propagation time between two neighbor nodes that can be calculated by the distance between them and the signal propagation speed. t_{tran} is the time to transmit the packet from the source to the next hop node that is determined by the packet length and wireless channel bandwidth. t_{queue} is the queuing delay and its determination is elaborated upon in the following section.

Multi Level Queueing Delay Analysis: Incoming packets from previous sensor nodes are classified into appropriate M/G/1[12] non-preemptive priority queues according to their priority levels as shown in Figure 3. In this paper, we have three priority classes (n=3) in each node: high, medium and low. Thus, the packets with higher deadline priorities can be scheduled to access the channel earlier.

This packet classifying strategy prevents a highest priority packet from being delayed by lower priority packets in the same node. In M/G/1 non-preemptive priority queues, the packets in each priority level are served in first-in-first-out (FIFO) discipline. The packet experiences a delay time at the intermediate node as it waits to be transmitted to the next node. In route, selection of the queue size (occupied buffer) of the next node is considered. The following will be used in the analysis [12]. The expected queuing waiting time of k^{th} queue (for $\forall K \in \{1, 2, 3\}$) is computed as:

$$E[W_k] = E[R] + \sum_{i=1}^{k} E[Q_i]\overline{X}_i + \sum_{i=1}^{k-1} \lambda_i E[W_k]\overline{X}_{k-1} \tag{11}$$

The server performance for packets of priority *i* is:

$$\rho_i = \frac{\lambda_i}{\mu_i} = \lambda_i \overline{X}_i \tag{12}$$

$$E[W_k] = \frac{E[R]}{(1-\sum_{j=1}^{k-1}\rho_j)(1-\sum_{j=1}^{k}\rho_j)} \tag{13}$$

Mean residual service time E[R] is:

$$E[R] = \frac{1}{2}\sum_{i=1}^{n}\lambda_i \overline{X_i^2} \tag{14}$$

Finally, using Eq. (13), we derive the final expression of $E[W_k]$:

$$E[W_k] = \frac{\sum_{i=1}^{n}\lambda_i \overline{X_i}}{2(1-\sum_{j=1}^{k-1}\rho_j)(1-\sum_{j=1}^{k}\rho_j)} \tag{15}$$

The average time a class k packet spends in the system can be calculated as:

$$E[t_{queue}(k)] = \frac{1}{\mu_k} + E[W_k] \quad , \quad \forall k = 1,2,3 \tag{16}$$

We can calculate the average time delay per packet using Little's Law:

$$E\left[t_{queue}\right] = \frac{\sum_{i=1}^{n}\lambda_i t_{queue(i)}}{\sum_{i=1}^{n}\lambda_i} = \text{n} = 3 \tag{17}$$

Table 1: Queuing Parameters

Parameter	Description
λ_i	Poisson process arrival rate for class i packets
$\overline{X_i} = 1/\mu_i$	Mean service time for class i packets
$\overline{X_i^2}$	Second moment
W_i	Average queuing time of class i packets
$\rho_i = \lambda_i / \mu_i$	Utilization factor for class i
R	Residual service time of packet
Q_i	Number of packets waiting in queue upon arrival of packet i

Table 2: Simulation Environment Settings

Parameter	**Value**
Bandwidth	200Kbps
Area	300m X 300m
Sensors	80
Packet size	60B
Initial energy	5j
Transmission power	0.8mw
Energy threshold	0.15j

According to the above formula, a packet in the high priority level queue is delivered first to the best next-hop candidate, so the end-to-end latency can be minimized. The parameters of M/G/1 queue are described in Table 1.

Performance Evaluation: We evaluate the performance of EER by comparing it with two location-based routing protocols, RTLD and MMSPEED protocols with C++.

In our evaluation, we show the following results: a) average end-to-end delay under different packet congestion levels, b) control packet overhead and c) per-packet energy consumption. All experiments are repeated ten times with different random node deployments. Table 2 shows the simulation parameters used to simulate ERR. We use some phenomena generator in our simulation environment that creates real-time traffic such as intrusion detection or fire detection reporting, with approximately 10% of total traffic.

Average End-To-End Delay: The time needed to send data packets from the source node to the sink node is the average end-to-end delay. Figure 4 shows that ERR provides a short average delay compared to the other two protocols. By changing the packet rates, we evaluate the average end-to-end delay. ERR considers required speed, which minimize the average end-to-end delay. When we increase the packet rates to ten packets per second, ERR significantly decreases the average end-to-end delay. Because ERR uses a three-level priority queue, real-time packets are in high priority classes and control packets are in low priority classes. Thus, this packet classifying strategy prevents a real-time packet from being delayed by non-real-time and control packets. Therefore, average end-to-end delay decreases.

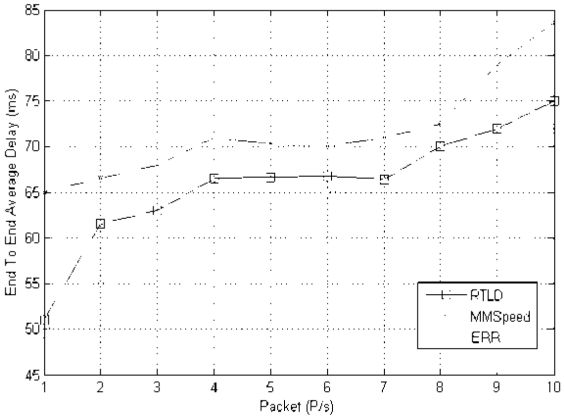

Fig. 4: End-to-end Average Delay

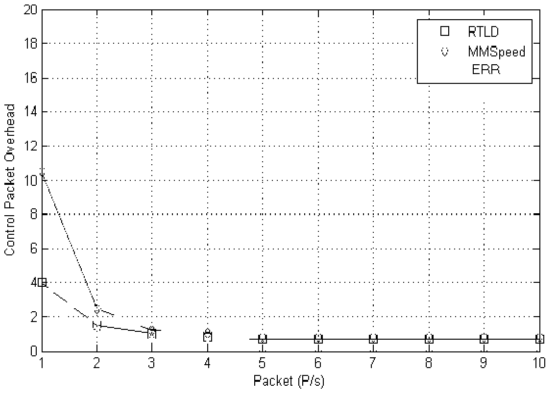

Fig. 5: Control Packet Overhead Figure 6. Energy Consumption per Packet

Control Packet Overhead: The control packet overhead indicates the ratio of control packets, used for optimum route discovery and route maintenance, to the total data packets delivered via sink node. Figure 5 shows that ERR has less control packet. This is due to its neighbor discovery, which does not allow the one-hop neighbor node to respond if it is not in the direction of the sink node or has lower energy than our threshold energy. Therefore, the probability of packet collision is reduced and control packet overhead is minimized. The reason for high overhead at the starting point is that extra control packets are used to create the neighbor table.

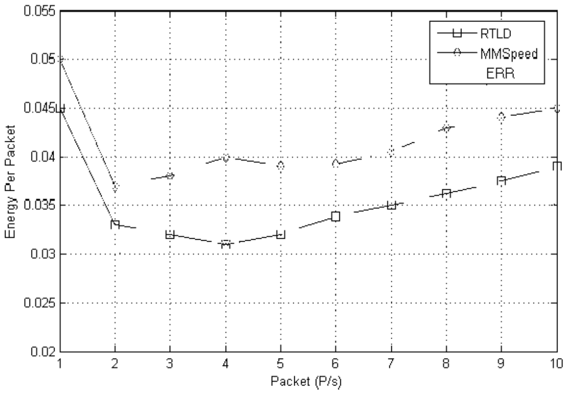

Fig. 6: Energy Consumption per Packet

Energy Consumption: Energy consumption measures the average amount of energy consumed by the source and relay nodes to get the data packet to the sink within the given simulation time interval. There is always a tradeoff between delay and energy consumption, because the average delay increases by saving more energy with assigning lower costs to neighbors that have lower energy requirements. Considering node residual energy, ERR effectively distributes energy consumption among nodes and can increase the lifetime of networks. Figure 6 demonstrates that ERR consumes less power compared to the other two protocols, because the packet overhead in ERR is less than in the other two protocols. The reduced energy consumption in ERR results from distributing the load throughout the neighboring nodes.

CONCLUSION

Energy efficient and real time data forwarding are most important parameters in wireless sensor networks. Delay sensitive data packets must forward to destination timely. In this paper, we propose an energy-efficient and real time routing scheme (ERR) for wireless sensor networks. ERR uses local decisions for selecting the next node without end-to-end path discovery and maintenance. For prolonging network life time we select next single-hop neighbor sensor nodes with maximum residual energy. We use a cost function with residual energy and buffer occupancy levels parameters for dynamically determining and selecting the next hop among the forwarding candidate nodes set. Simulation results show that, ERR provides timeliness and energy efficient requirements for WSNs, balances the energy consumption in nodes and is adaptive to the dynamics and scalability of WSNs. As a result, this proposed method can significantly improve on-time packet delivery rate and, most importantly, extended network lifetimes.

REFERENCES

1. Ian, F.A., Su, S. Weilian, S. Yogesh and C. Erdal, 2002. A Survey on sensor networks. IEEE Commun. Mag., 14: 102-114.
2. Yuyan, X., R. Byrav and V. Mehmet, 2008. A service-Differentiated Real-time Communication Scheme for Wireless Sensor Networks. In Proceedings of the 33rd IEEE Conf. on Local Computer Networks, 2008. pp: 748-755.
3. Chenyang, L., B. Brian, A. Tarek, S. John and H. Tian, 2002. RAP: A Real-time Communication Architecture for Large-scale Wireless Sensor Networks. In Proceedings of the Real-Time and Embedded Technology and Applications Symposium, IEEE Conference, pp: 55-66.
4. Octav, C., H. Zhimin, X. Guoliang, C. Qin, W. Xiaorui, L. Chenyang, John and S. Tarek, 2006. A.Real-time power-aware Routing in Sensor Network. In Proceedings of IWQoS, pp: 83-92.
5. Yian, H., A. SJohn S. Chenyang and L. Tarek, 2003. A. SPEED: A Stateless Protocol for Real-time Communication in Sensor Networks. In Proceedings of the 23rd International Conference on Distributed Computing Sys., pp: 46-55.
6. Emad, F., F. Chang-Gun and L. Eylem, 2006. E. MMSPEED: Multipath Multi-speed Protocol for QoS Guarantee of Reliability and Timeliness in Wireless Sensor Network. IEEE Trans. Mobile Computing. 5(6): 738-754.
7. Kemal, A. and Y. Mohamed, 2005. Energy-Aware QoS Routing in Wireless Sensor Networks" Cluster Computing, 8(2-3): 179-188.
8. Adel Ali, A. and F. Norsheila, 2008. A Real-time Routing Protocol with Load Distribution in Wireless Sensor Networks. Computer Communications, pp: 3190-3203
9. Karp, B. and H. Kung, 2000. Greedy Perimeter Stateless Routing for Wireless Networks. In proceedings of IEEE/ACM Int'l Conf. Mobile Computing and Networking, pp: 243-254.
10. He, T., C. Huang, B. Blum, J. Stankovic and T. Abdelzaher, 2003. Range-Free Localization Schemes for Large Scale Sensor Networks, Proc. Mobicom Conf.,

11. Doherty, L., K.S.J. Pister and L.E. Ghaoui, 2001. Convex Estimation Position in Wireless Sensor Networks. Proc. Infocom Conf.,
12. Leon-Garcia, A., A. Probability and Random Process for Electrical Engineering. 2nd ed.Wesley, 1994.
13. Ghaffari, A., A.M. Rahmani and A. Khademzadeh, 2011. Energy-efficient and QoS-aware geographic routing protocol for wireless sensor networks. IEICE Electron. Express, 8(8): 582-588.
14. Saleh. H., A. Al-Sharaeh, A. Sharieh, A.L. Abu Dalhoum, R. Hosny and F. Mohammed, 2008. Multi-Dimensional Poisson Distribution Heuristic for Maximum Lifetime Routing in Wireless Sensor Network. World Applied Science J., 5(2): 119-131.