Traffic-Differentiated Two-Hop Routing for Wireless Sensor Networks

Shiva Prakash T.¹, Chandrashekar H. S.¹, Raja K. B.¹, Suresh Babu K.¹, Venugopal K. R.¹, Iyengar, S. S.² and Patnaik L. M.³

¹University Visvesvaraya College of Engineering, Bangalore University, Bangalore, India.
²Florida International University, Miami, Florida, USA.
³Indian Institute of Science, Bangalore, India.

Abstract. This paper proposes a Traffic-Differentiated Two-Hop Routing protocol for Wireless Sensor Networks (WSNs). It targets WSN applications having different types of data traffic with several priorities. The protocol achieves to increase packet reception ratio and reduce end-to-end delay while considering multi-queue priority policy, two-hop neighborhood information, link reliability and power efficiency. The protocol is modular and utilizes memory and computational effective methods for estimating the link metrics. Numerical results show that the proposed protocol is a feasible solution to addresses QoS service differentiation for traffic with different priorities.

Keywords: end-to-end delay, packet reception ratio (PRR), quality-of-service (QoS), traffic-differentiation, two-hop neighbors, wireless sensor networks (WSNs).

1. Introduction

Wireless Sensor Networks (WSNs) form a framework to accumulate and analyze real time data in smart environment applications. WSNs are composed of inexpensive low-powered micro sensing devices called motes [1], having limited computational capability, memory size, radio transmission range and energy supply.

Emerging WSNs have a set of stringent QoS requirements that include timeliness, high reliability, availability and integrity. Various performance metrics that can be used to justify the quality of service include, packet reception ratio (PRR), defined as the probability of successful delivery should maximized. The end-to-end delay which is influenced by the queuing delay at the intermediate nodes and the number of hops traversed by the data flows of the session from the source to the receiver. Therefore, providing corresponding traffic differentiation QoS in such scenarios pose a great challenge. Our proposed protocol is motivated primarily by the deficiencies of the previous works (explained in the Section 2) and aims to provide better Quality of Service.

This paper explores the idea of incorporating QoS parameters in making routing decisions the protocol proposes the following features.
### Table 1. Our results and comparison with previous results for QoS routing in Wireless Sensor Networks.

<table>
<thead>
<tr>
<th>Related Work</th>
<th>Protocol Name</th>
<th>Metrics and Estimation</th>
<th>Traffic Differentiation</th>
<th>Duplication</th>
</tr>
</thead>
<tbody>
<tr>
<td>E. Felemban et al., [3]</td>
<td>MMSPEED (Multi-path and Multi-SPEED Routing Protocol)</td>
<td>one-hop delay, link reliability and residual energy. EWMA</td>
<td>Delay requirement</td>
<td>Towards the same sink</td>
</tr>
<tr>
<td>Y. Li et al., [7]</td>
<td>THVR (Two-Hop Velocity Based Routing Protocol)</td>
<td>two-hop delay and residual energy. EWMA, WMEWMA</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>D. Djenouri et al., [5]</td>
<td>LOCALMOR (Localized Multiobjectives Routing)</td>
<td>one-hop delay link reliability residual energy and transmission power. EWMA, WMEWMA</td>
<td>Regular, Reliability sensitive, Delay, sensitive and Critical Traffic</td>
<td>Towards different sinks</td>
</tr>
<tr>
<td>This paper</td>
<td>TDTHR (Traffic Differentiated Two-Hop Routing)</td>
<td>two-hop delay, link reliability residual energy and transmission power. EWMA, WMEWMA</td>
<td>Regular, Reliability Responsive, Delay, Responsive and Critical Traffic</td>
<td>Towards different sinks</td>
</tr>
</tbody>
</table>

1. Data Traffic is split into regular traffic with no specific QoS requirement, reliability-responsive traffic; which should be transmitted without loss but can tolerate some delay, delay-responsive traffic; which should be delivered within a deadline but may tolerate moderate packet loss and critical traffic; which has high significance and demanding both high reliability and short delay.
2. Link reliability is considered while choosing the next router, this selects paths which have higher probability of successful delivery.
3. Routing decision is based on two-hop neighborhood information and dynamic velocity that can be modified according to the required deadline, this results in significant reduction in end-to-end PRR.
4. Choosing nodes with higher residual energy and minimum transmission power, balances the load among nodes and results in prolonged lifetime of the network.

We test the performance of our proposed approaches by implementing our algorithms using ns-2 simulator. Our results demonstrates the performance and benefits of TDTHR over earlier algorithms.

The rest of the paper is organized as follows: Section 2 gives a review of related works. Section 3 and Section 4 detail the studied problem, network model, notations, assumptions and proposed routing algorithm. Section 5 is devoted to the simulation and evaluation of the algorithm. Conclusions are presented in Section 6.
2. Related Work

Stateless routing protocols which do not maintain per-route state is a favorable approach for WSNs. The idea of stateless routing is to use location information available to a node locally for routing, i.e., the location of its own and that of its one-hop neighbors without the knowledge about the entire network.

SPEED (Stateless Protocol for End-to-End Delay) [2] is based on geometric routing protocols such as greedy forwarding GPSR (Greedy Perimeter State Routing) [9,10]. Lu et al., [11] describe a packet scheduling policy, called Velocity Monotonic Scheduling, which inherently accounts for both time and distance constraints. Sequential Assignment Routing (SAR) [12] is the first routing protocol for sensor networks that creates multiple trees routed from one-hop neighbors of the sink by taking into consideration both energy resources, QoS metric on each path and priority level of each packet. However, the protocol suffers from the overhead of maintaining the tables and states at each sensor node especially when the number of nodes is large.

MMSPEED (Multi-path and Multi-SPEED Routing Protocol) [3] is an extension of SPEED that focuses on differentiated QoS options for real-time applications with multiple different deadlines. It provides differentiated QoS options both in timeliness domain and the reliability domain. For timeliness, multiple QoS levels are supported by providing multiple data delivery speed options. MMSPEED does not include energy metric during QoS route selection. Chipera et al., [6] (RPAR: Real-Time Power Aware Routing) have proposed another variant of SPEED. Where a node will change its transmission power by the progress towards destination and packet’s slack time in order to meet the required velocity; they have not considered residual energy and reliability.

DARA (Distributed Aggregate Routing Algorithm) [4] considers reliability, delay, and residual energy in the routing metric, and defines two kinds of packets: critical and noncritical packets. For delay estimation, the authors use queuing theory and suggest a method that, in practice, needs huge amount of sample storages. Mahapatra et al., [13] assign an urgency factor to every packet depending on the residual distance and time the packet needs to travel, and determines the distance the packet needs to be forwarded closer to the destination to meet its deadline. Multi-path routing is performed only at the source node for increasing reliability.

Sharif et al., [14] presented a new transport layer protocol that prioritizes sensed information based on its nature while simultaneously supporting the data reliability and congestion control features. Rusli et al., [15] propose an analytical framework model based on Markov Chain of OR and M/D/1/K queue to measure its performance in term of end-to-end delay and reliability in WSNs. Koulali et al., [16] propose a hybrid QoS routing protocol for WSNs based on a customized Distributed Genetic Algorithm (DGA) that accounts for delay and energy constraints. Yunbo Wang et al., [17] investigate the end-to-end delay distribution, they develop a comprehensive cross-layer analysis framework, which employs a stochastic queuing model in realistic channel environments. Ehsan et al., [18] propose energy and cross-layer aware routing schemes for multichannel access WSNs that account for radio, MAC contention, and network constraints.

All the above routing protocols are based on one-hop neighborhood information. However, it is expected that multi-hop information can lead to improved performance in many issues including message broadcasting and routing. Chen et al., [19] study the performance of 1-hop, 2-hop and 3-hop neighborhood information based routing and propose that gain from 2-hop to 3-hop is relatively minimal, while that from 1-hop to 2-hop based routing is significant. Li et al., [7] have proposed a
Two-Hop Velocity Based Routing Protocol (THVR). The routing choice is decided on the two-hop relay velocity and residual energy, an energy efficient packet drop control is included to enhance packet utilization efficiency while keeping low packet deadline miss ratio. Djenouri et al., [5] propose a new localized quality of service (QoS) routing protocol (LOCALMOR) it is based on differentiating QoS requirements according to the data type, which enables to provide several and customized QoS metrics for each traffic category. With each packet, the protocol attempts to fulfill the required data-related QoS metric(s) while considering power efficiency. The protocol proposed in this paper is different from LOCALMOR it considers two-hop transmission delay and queuing delay for selecting the next node.

3. Problem Definition
The topology of a wireless sensor network may be described by a graph $G = (N, L)$, where $N$ is the set of nodes and $L$ is the set of links. The objectives are to,

- Maximize the Packet Reception ratio (PRR).
- Reduce the end-to-end packet delay.
- Improve the energy efficiency (ECPP-Energy Consumed Per Packet) of the network.

3.1 Network model and assumptions
In our network model, we assume the following:

- The wireless sensor nodes consists of $N$ sensor nodes and a sink, the sensors are distributed randomly in a field. The nodes are aware of their positions through internal global positioning system (GPS).
- The $N$ sensor nodes are powered by a non renewable on board energy source. All nodes are supposed to be aware of their residual energy and have the same transmission power range.
- The sensors share the same wireless medium each packet is transmitted as a local broadcast in the neighborhood. We assume any MAC protocol, which ensures that among the neighbors in the local broadcast range, only the intended receiver keeps the packet and the other neighbors discard the packet.
- Like all localization techniques, [2,3,20] each node needs to be aware of its neighboring nodes current state (ID, position, link reliability, residual energy etc), this is done via HELLO messages.
- In addition, each node sends a second set of HELLO messages to all its neighbors informing them about its one-hop neighbors. Hence, each node is aware of its one-hop and two-hop neighbors and their current state.

4. Algorithm
TDTHR has the following components: a link reliability estimator, a queuing and transmission delay estimator, a queuing controller and a node forwarding metric incorporated with the two-hop dynamic velocity assignment policy. The proposed protocol LRTHR implements the modules for estimating queuing and transmission delay and packet delivery ratios using efficient methods. The packet delay is estimated at the node itself and the packet delivery ratio is estimated by the neighboring
Table 2. Notations used in Section 4.

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>$N$</td>
<td>Set of Nodes in the WSN</td>
</tr>
<tr>
<td>$D$</td>
<td>Destination Node</td>
</tr>
<tr>
<td>$S$</td>
<td>Source Node</td>
</tr>
<tr>
<td>dist$(x, y)$</td>
<td>Distance between a node pair $x, y$</td>
</tr>
<tr>
<td>$N_1(x)$</td>
<td>Set of one-hop Neighbors of node $x$</td>
</tr>
<tr>
<td>$N_2(x)$</td>
<td>Set of two-hop Neighbors of node $x$</td>
</tr>
<tr>
<td>$F^+_1(x)$</td>
<td>Set of node $x$’s one-hop favorable forwarders providing positive progress towards the destination $D$</td>
</tr>
<tr>
<td>$F^+_2(x, y)$</td>
<td>Set of node $x$’s two-hop favorable forwarders</td>
</tr>
<tr>
<td>$d_{txy}$</td>
<td>Estimated one-hop transmission delay between nodes $x$ and $y$</td>
</tr>
<tr>
<td>$d_{qy}$</td>
<td>Estimated queuing delay at node $y$</td>
</tr>
<tr>
<td>$t_{req}$</td>
<td>Time deadline to reach Destination $D$</td>
</tr>
<tr>
<td>$V_{req}$</td>
<td>Required end-to-end packet delivery Velocity for deadline $t_{req}$</td>
</tr>
<tr>
<td>$V_{xy}$</td>
<td>Velocity offered by $y \in F^+_1(x)$</td>
</tr>
<tr>
<td>$V_{xy, z}$</td>
<td>Velocity offered by $y \in F^+_2(x, y)$</td>
</tr>
<tr>
<td>$S_{req}$</td>
<td>Node pairs satisfying $V_{xy, z} \geq V_{req}$</td>
</tr>
<tr>
<td>$T_p(dist(x, y))$</td>
<td>Transmission power cost from node $x$ to $y$</td>
</tr>
<tr>
<td>$E_y$</td>
<td>Remaining energy of node $y$</td>
</tr>
<tr>
<td>$prr_{xy}$</td>
<td>Packet Reception Ratio of link relaying node $x$ to node $y$</td>
</tr>
<tr>
<td>$\beta$</td>
<td>Tunable weighting coefficient for $prr$ estimation</td>
</tr>
<tr>
<td>$\gamma$</td>
<td>Tunable weighting coefficient for queuing and transmission delay estimation</td>
</tr>
</tbody>
</table>

nodes. These parameters are updated on reception of a HELLO packet, the HELLO messages are periodically broadcast to update the estimation parameters. The overhead caused by the 1-hop and 2-hop updating are reduced by piggybacking the information in ACK, hence improving the energy efficiency. The notations used in this paper are given in Table 2. The protocol is based on the following parameters: (i) Link Reliability Estimation; (ii) Queuing and Transmission Delay Estimation; (iii) Node Forwarding Metric; and (iv) Queuing Controller.

4.1 Link reliability estimation

The Packet Reception Ratio (PRR) of the link relaying node $x$ to $y$ is denoted by $prr_{xy}$. It denotes the probability of successful delivery over the link. Window Mean Exponential Weighted Moving Average (WMEWMA) based link quality estimation is used for the proposed protocol. This parameter is updated by node $y$ at each window and inserted into the HELLO message packet for usage by node $x$ in the next window. Eqn. 1 shows the window mean exponential weighted moving average estimation of the link reliability, $r$ is the number of packets received, $m$ is the number of packets missed and $\beta \in [0, 1]$ is the history control factor, which controls the effect of the previously estimated value on the new one. $\frac{r}{r+m}$ is the newly measured PRR value.

\[
prr_{xy} = \beta \times prr_{xy} + (1 - \beta) \times \frac{r}{r + m} \tag{1}
\]
The PRR estimator is updated at the receiver side for each \( w \) (window size) received packets, the computation complexity of this estimator is \( O(1) \). The appropriate values for \( \beta \) and \( w \) for a stable window mean exponential weighted moving average are \( w = 30 \) and \( \beta = 0.6 \) [23].

### 4.2 Queuing and transmission delay estimation

The nodal delay indicates the time spent to send a packet from node \( x \) to its neighbor \( y \), it is comprised of the queuing delay (\( \text{delay}_Q \)), contention delay (\( \text{delay}_C \)) and the transmission delay (\( \text{delay}_T \)).

\[
\text{delay}_{node} = \text{delay}_Q + \text{delay}_C + \text{delay}_T
\]  

The queuing delay constitutes the time the packet is assigned to a queue for transmission and the time it starts being transmitted. During this time, the packet waits while other packets in the transmission queue are transmitted. Every node evaluates its queuing delay \( dq_x \) for the various classes of queues used, i.e., Critical-Queue, Delay-Responsive-Queue, Reliability-Responsive-Queue and Regular-Queue, each packet class has a different estimation of \( dq_x \) for the queuing delay, i.e., \( dq_x \ < \text{packet.class} > \). Equation 3 shows the EWMA (Exponential Weighted Moving Average) update for queuing delay estimation, \( dq \) is the current precise queue waiting time of the respective packet and \( \gamma \in [0, 1] \) is the tunable weighting coefficient.

\[
dq_x \ < \text{Packet.class} > = \gamma \times dq_x \ < \text{Packet.class} > + (1 - \gamma) \times dq \tag{3}
\]

The transmission delay represents the time that the first and last bits of the packet are transmitted. If \( t_s \) is the time the packet is ready for transmission and becomes head of transmission queue, \( t_{ack} \) the time of the reception of acknowledgment, \( BW \) the network bandwidth and size of the acknowledgment then, \( t_{ack} - \text{sizeof(ACK)}/BW - t_s \) is the recently estimated delay. Equation 4 shows the EWMA (Exponential Weighted Moving Average) update for transmission delay estimation, which has the advantage of being simple and less resource demanding.

\[
d_{txy} = \gamma \times d_{txy} + (1 - \gamma) \times (t_{ack} - \text{sizeof(ACK)}/BW - t_s) \tag{4}
\]

\( d_{txy} \) includes estimation of the time interval from the packet that becomes head of line of \( x \)’s transmission queue until its reception at node \( y \). This takes into account all delays due to contention, channel sensing, channel reservation (RTS/CTS) if any, depending on the medium access control (MAC) protocol, propagation, time slots etc. The computation complexity of both the estimators is \( O(1) \). The delay information is further exchanged among two-hop neighbors.

### 4.3 Node forwarding metric

In the wireless sensor network, described by a graph \( G = (N, L) \). If node \( x \) can transmit a message directly to node \( y \), the ordered pair is an element of \( L \). We define for each node \( x \) the set \( N_1(x) \), which contains the nodes in the network \( G \) that are one-hop i.e., direct neighbors of \( x \).

\[
N_1(x) = \{ y : (x; y) \in E \text{ and } y \neq x \} \tag{5}
\]

Likewise, the two-hop neighbors of \( x \) is the set \( N_2(x) \) i.e.,

\[
N_2(x) = \{ z : (y; z) \in E \text{ and } y \in N_1(x), z \neq x \} \tag{6}
\]
The Euclidean distance between a pair of nodes \( x \) and \( y \) is defined by \( \text{dist}(x, y) \). We define \( F_1^{+p}(x) \) as the set of \( x \)'s one-hop favorable forwarders providing positive progress towards the destination \( D \). It consists of nodes that are closer to the destination than \( x \), i.e.,
\[
F_1^{+p}(x) = \{ y \in N_1(x) : \text{dist}(x, D) - \text{dist}(y, D) > 0 \}
\]  
(7)

\( F_2^{+p}(x) \) is defined as the set of two-hop favorable forwarders i.e.,
\[
F_2^{+p}(x) = \{ y \in F_1^{+p}(x), z \in N_1(y) : \text{dist}(y, D) - \text{dist}(z, D) > 0 \}
\]  
(8)

We define two velocities; the required velocity \( V_{req} \) and the velocity offered by the two-hop favorable forwarding pairs. In SPEED, the velocity provided by each of the forwarding nodes in \( (F_1^{+p}(x)) \) is
\[
V_{xy} = \frac{\text{dist}(x, D) - \text{dist}(y, D)}{d_{txy}}
\]  
(9)

As in THVR, by two-hop knowledge, node \( x \) can calculate the velocity offered by each of the two-hop favorable forwarding pairs \( (F_1^{+p}(x), F_2^{+p}(x)) \) as shown in Eqn. 10. Furthermore, we include queuing delay at both the current \( d_{q_x} \) and the next hop \( d_{q_y} \) nodes, with the two-hop transmission delay \( d_{txy} \) and \( d_{tyz} \), this distinguishes the proposed protocol from LOCALMOR.
\[
V_{xy\rightarrow z} = \frac{\text{dist}(x, D) - \text{dist}(z, D)}{d_{q_x} <\text{packet.class}> + d_{txy} + d_{q_y} <\text{packet.class}> + d_{tyz}}
\]  
(10)
where, $y \in F_{1}^{+p}(x)$ and $z \in F_{2}^{+p}(x)$. The required velocity is relative to the progress made towards the destination [6] and the time remaining to the deadline, $lt$ (lag time). The lag time is the time remaining until the packet deadline expires. At each hop, the transmitter renews this parameter in the packet header i.e.,

$$lt = lt_{p} - (t_{tx} - t_{rx} + \text{sizeof}(\text{packet})/BW) \quad (11)$$

where $lt$ is the time remaining to the deadline ($t_{req}$), $lt_{p}$ is the previous value of $lt$, $(t_{tx} - t_{rx} + \text{sizeof}(\text{packet})/BW)$ accounts for the delay from reception of the packet until transmission. On reception of the packet the node $x$, uses $lt$ to calculate the required velocity $V_{req}$ for all nodes in $(F_{1}^{+p}(x), F_{2}^{+p}(x))$ as shown in Eqn. 12.

$$V_{req} = \frac{\text{dist}(x, D)}{lt} \quad (12)$$

The node pairs satisfying $V_{xy \rightarrow z} \geq V_{req}$ form the set of nodes $S_{req}$. If the packet class is $delay.responsive$ then the node with the maximum residual energy and minimum transmission power cost is chosen from the set $S_{req}$. However, if the packet class is $critical$ then the node with the highest packet reception ratio (PRR) is selected from $S_{req}$, but if more than one node has the same maximum PRR, a node with maximum power efficiency is picked.

The Traffic-Differentiation Based Two-Hop Routing is shown in Algorithm 1, the computation complexity of this algorithm is $O(F_{2}^{+p}(x))$. Our proposed protocol is different from LOCALMOR, as it considers two-hop neighborhood information that will provide enhanced foresight to the sender in identifying the node that can offer the required QoS and route the packets in real-time.

4.4 Queuing controller

The queuing controller helps accomplish low delay when routing critical and delay-responsive packets, higher precedence should be given to these packets in channel contention than the normal packets (regular and reliability-responsive packets). Additionally, critical packets need higher priority than delay-responsive packets. This can be accomplished by implemented the queuing controller.
controller module as detailed in Algorithm 2 [5]. Three queues are used to send packets from the highest priority queue to the lowest one. The highest priority queue, Critical-Queue, is used by critical packets, the second highest priority queue, Delay-Responsive-Queue, is used by delay-responsive packets, and the least priority queue, Reliability-Responsive-Queue, is used by regular and reliability-responsive packets. The number of critical and delay-responsive packets is usually small, and there would be instances where their corresponding queues are vacant. Otherwise, lower priority traffic may be forever blocked by higher priority traffic. In this case, a timer for each packet is employed to move it to the highest priority queue.

5. Performance Evaluation

To evaluate the proposed protocol, we carried out a simulation study using ns-2 [24]. In this study the proposed protocol (LRDTHR) is compared with LOCALMOR, DARA and MMSPEED. The simulation configuration consists of 900 nodes located in a 1800 m² area. Nodes are distributed following Poisson point process with a node density of 0.00027 node/m². The primary and secondary sink nodes are located in the region (0, 0) and (1800, 1800) while the source node is located in the center of the simulation area, equidistant from both the sinks. The source generated a CBR flow of 1 kB/second with a packet size of 150 bytes. Critical and regular packets are used in the simulation for comparing our protocol with LOCALMOR, DARA and MMSPEED, while delay-sensitive and reliability-sensitive packets are used for comparing with LOCALMOR only. The deadline requirement fixed in this simulation is 300 ms for all class of packets.

The MAC layer, link quality and energy consumption parameters are set as per TelosB³ (TPR2420) mote[25] with CC2420 radio as per LOCALMOR. Table 3 summarizes the simulation parameters. LOCALMOR, DARA and MMSPEED are QoS protocols and a comparison of PRR (Packet Reception Ratio), ECPP (Energy Consumed Per Packet i.e., the total energy expended divided by the number of packets effectively transmitted), packet average end-to-end delay (mean of packet delay) are obtained.
Table 3. Simulation Parameters.

<table>
<thead>
<tr>
<th>Simulation Parameters</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of nodes</td>
<td>900</td>
</tr>
<tr>
<td>Simulation Topology</td>
<td>1800 m × 1800 m</td>
</tr>
<tr>
<td>Traffic</td>
<td>CBR</td>
</tr>
<tr>
<td>Critical Packet Rate</td>
<td>From 0 to 1</td>
</tr>
<tr>
<td>Regular Packet Rate</td>
<td>1 - Critical Packet Rate</td>
</tr>
<tr>
<td>Payload Size</td>
<td>150 Bytes</td>
</tr>
<tr>
<td>Transmission Power Range</td>
<td>100 m</td>
</tr>
<tr>
<td>Initial Battery Energy</td>
<td>2.0 Joules</td>
</tr>
<tr>
<td>Energy Consumed during Transmit</td>
<td>0.0522 Joules</td>
</tr>
<tr>
<td>Energy Consumed during Receive</td>
<td>0.0591 Joules</td>
</tr>
<tr>
<td>Energy Consumed during Sleep</td>
<td>0.00006 Joules</td>
</tr>
<tr>
<td>Energy Consumed during Idle</td>
<td>0.000003 Joules</td>
</tr>
<tr>
<td>MAC Layer</td>
<td>802.11 with DCF</td>
</tr>
<tr>
<td>Propagation Model</td>
<td>Free Space</td>
</tr>
<tr>
<td>Hello Period</td>
<td>5 seconds</td>
</tr>
<tr>
<td>PRR – WMEWMA Window</td>
<td>30</td>
</tr>
<tr>
<td>PRR – WMEWMA Weight Factor (β)</td>
<td>0.6</td>
</tr>
<tr>
<td>Queuing/Delay – EWMA Weight Factor (γ)</td>
<td>0.5</td>
</tr>
</tbody>
</table>

In the first set of simulations the critical packet rate was varied from 0.1 to 1 and the remaining rate to 1 represents regular packet rate. Figure 1 illustrates the efficiency of the TDTHR algorithm in increasing the PRR with respect to critical packets, the performance of TDTHR is better than others in general. TDTHR, LOCALMOR and DARA linearly increase their performance as a function of critical packet rate, while performance of MMSPEED is relatively stable. The linear increase of the packet reception ratio for TDTHR, LOCALMOR and DARA with the increasing critical packet rate can be explained by the subsequent increase of duplications (applied only to critical packets). This means, the larger the number of critical packets we have, the more the packets are duplicated, which subsequently increases their reception ratio. In TDTHR the two-hop based routing and dynamic velocity of the TDTHR algorithm is able to aggressively route more packets to the sink node, hence it is observed that TDTHR has higher critical PRR than the others in general.

Figure 2 illustrate the packet end-to-end delay of critical packets respectively, performance of LRTHR is better that the other protocols. LOCALMOR and DATA consider one-hop transmission delay and queuing waiting time, while TDTHR considers dynamic velocity, two-hop transmission delay and queuing waiting time hence the selected paths from source to sink will be shorter and aid in reducing the end-to-end delay. MMSPEED also considers queuing and transmission delays, but on the other hand, the use of multipath single-sink transmissions causes congestion and thus results in several retransmission of packets before successful reception, which explains the relatively higher delay. In Figure 2 we notice a stable end-to-end delay for all protocols, which reflects the stability
of the routes selected for critical packets that are obviously not affected by the rise in rate of critical packets.

As depicted in Figure 3 the energy consumption per packet (ECPP) successfully transmitted, ascend as the critical packet rate increases. The energy consumption has similar tendency in both TDTHR and LOCALMOR but DARA has a higher energy utilization. LOCALMOR ensures a trade-off between traffic related QoS metrics and energy, its ECPP smoothly increases as rates become higher. LOCALMOR balances the load only among nodes estimated to ensure delivery within the deadline and having the highest reliability. DARA performs poorly in terms of energy, as it does neither use any traffic balancing technique nor any probabilistic selection.

In TDTHR the two-hop based routing will ensure shorter paths between source and sink, by selecting links providing higher PRR on the route to the sink, the energy consumption of the for-
warding nodes can be minimized, due to lower number of collisions and re-transmissions and help in traffic balancing. Furthermore, in the proposed protocol the link delay and packet delivery ratios are updated by piggybacking the information in ACK, this will help in reducing the number of feedback packets and hence reduce the total energy consumed.

6. Conclusions

In this paper, we propose a Traffic-Differentiation Based Two-Hop neighborhood based quality of service (QoS) routing protocol for WSN, it provides a differentiation routing using different quality of service metrics The consideration of two-hop neighborhood information and differentiation of both delay and reliability requirements distinguish the proposed protocol from other protocols. The protocol is able to augment real-time delivery by an able integration of multi queue priority policy, link reliability, two-hop information and dynamic velocity. The protocol is able to reduce the PRR, end-to-end delay and improve the energy efficiency throughout the network. This makes the protocol suitable for WSNs with varied traffic, such as medical and vehicular applications.

References


