CHAPTER 4
ENERGY AND DELAY IN WEIGHTED FAIR QUEUING ROUTING

Networking unattended wireless sensors expected to have a significant impact on the efficiency of many militaries and civil applications such as combat field surveillance, security and disaster management. For this instance, disaster management setup networking these sensors can assist rescue operations by locating survivors, identifying risky areas and making the rescue crew more aware of the overall situation. On the military side, applications of the sensor network are numerous. For example, the use of the networked set of sensor nodes has limited the need for personnel involvement in the usually dangerous reconnaissance missions. Wireless Sensor nodes used in unattended setups have usually miniaturized and constrained in energy supply. Such constraint has necessitated energy-awareness at most layers of networking protocol stack including the network layer. Also many applications require the deployment of a large number of sensor nodes making it impractical to build a global addressing scheme.

Energy-efficient routing mechanism to ensure bounded delay (known delay) for the data delivery in the sensor network is discussed. Moreover, the approach sets up energy-aware multi-hop data paths considering sensor’s transmission power and sensor energy reserve and imposes end-to-end delay as a constraint. End-to-end delay bound is achieved through the use of a Weighted Fair Queuing (WFQ) based packet scheduling technique in each sensor node. WFQ consider queue for each incoming flow and has been shown to provide, in the statistical term, an upper bound on delay for a leaky bucket constrained flow. Moreover, also regulate the incoming traffic from the sources by using the
leaky-bucket traffic regulation mechanism and separate the real-time traffic from non-constrained traffic with the usage of two different queues in each node.

4.1 Network Topology

A set of sensor node spreads throughout an area of interest to detect when an event occurs in a particular area. The sensor nodes are always battery-operated. The mission for these sensor nodes has been dynamically changing to serve the need of command center. A sensor gateway node, which is significantly less energy-constrained than the sensors, is deployed in the physical proximity of sensors. The gateway is assumed to know the geographical location of deployed sensors. The gateway is responsible for organizing the activities at sensor nodes contingent in achieving a mission, fusing data collected by sensor nodes, coordinating communication among nodes and interacting with command node. The gateway node sends to the command node, reports generated through the fusion of sensor readings, e.g., events of detected events. The command node presents these reports to the user and performs the system-level fusion of the collected reports for overall situation awareness.

4.1.1 Weighted Fair Queue method

In a battle environment, it is crucial to locate, detect and identify an event. Target identification and monitoring have been performed through the deployment of imaging sensors. After locating and detecting the events using acoustic sensors, one can turn on imaging sensors to capture images of the target and periodically send them to the controller to take the proper actions. In case, it should deal with real-time data, which requires a minimum level of bandwidth and the bounded delay. Therefore, a service differentiation mechanism is needed to guarantee the reliable delivery of such data.
In such an application have been finding an optimal path to the gateway regarding energy consumption and performance rate while meeting the end-to-end delay requirement. End-to-end delay requirement is associated only with the real-time data. The Quality of service routing problem is very similar to typical path constrained path optimization problems. Moreover, then trying to find a least-cost path, which meets the end-to-end delay path constraint. This approach is based on associating a cost function for each link and using a $K$ least cost path algorithm to find a set of candidate routes. Such routes are checked against the end-to-end delay constraints and the one that meets the requirements is picked, to achieve such delay constraints for real-time traffic and employ a packet scheduling technique, namely Weighted Fair Queuing (WFQ), at each sensor node to differentiate between delay-constrained and non-constrained packets.

Table 4.1 Generalized Processor Sharing

<table>
<thead>
<tr>
<th>Symbols</th>
<th>Definitions</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\sigma_l$</td>
<td>Maximum burst size for leaky bucket on flow $i$</td>
</tr>
<tr>
<td>$\rho_l$</td>
<td>Average data rate of the flow $i$</td>
</tr>
<tr>
<td>$D(i)$</td>
<td>End-to-end delay for flow $i$</td>
</tr>
<tr>
<td>$C$</td>
<td>Link bandwidth $i$</td>
</tr>
<tr>
<td>$P_{\text{max}}(i)$</td>
<td>Maximum packet size for flow $i$</td>
</tr>
<tr>
<td>$P_{\text{max}}$</td>
<td>Maximum packet size allowed in the network</td>
</tr>
<tr>
<td>$g_{l,m}$</td>
<td>Service rate on node $m$ for flow $i$</td>
</tr>
<tr>
<td>$g(i)$</td>
<td>Minimum of all service rates for flow $i$</td>
</tr>
<tr>
<td>$M$</td>
<td>The number of nodes on path of flow $i$</td>
</tr>
</tbody>
</table>

In such a model, each sensor node employs scheduling discipline that approximates the Generalized Processor Sharing (GPS). GPS achieves exact weighted max-min fairness by dedicating a separate FIFO queue for each session and serving an infinitely small amount of data from each queue in a weighted
round robin fashion. GPS works in details, introduce the following notations as shown in the Table 4.1.

A-GPS server $m$ handles $n$ sessions on a link by giving each session a share of the link based on $n$ positive real numbers, $s_1^m, s_2^m, \ldots, s_n^m$. These numbers denote the relative amount of service to each flow on the server $m$. The GPS server ensures that backlogged connections share the remaining bandwidth in proportion to the assigned weights. Analyzed in each backlogged connection ‘$i$’ receives a service rate as shown in the Equation 4.1.

$$g_i^m = \frac{\phi_i^m}{\sum_{j=1}^{n} \phi_j^m} C$$  \hspace{1cm} (4.1)

However, GPS is not implementable in practice due to its ideal fluid model. Therefore, packet approximation algorithms of GPS are proposed. The Weight Fair Queuing (WFQ) and the Packetized Generalized Processor Sharing (PGPS) are two identical disciplines developed independently and do not require GPS’s infinitely small service assumption. They serve the incoming packets according to their service times under GPS. Therefore, for each flow of the packet with the earlier service time have served first. Throughout this chapter, using the term WFQ for the packet-based version of GPS. WFQ has two important features: First, it provides a fair allocation of bandwidth among all backlogged sessions as long as the total service rate of all sessions is less than the link bandwidth. Secondly, when combined with traffic regulation, it has been shown to provide an upper bound for the end-to-end delay. Such regulation has been done through the use of a leaky bucket mechanism at the sources to ensure a constant data rate and to restrict the burst size for the traffic reaching the next relay sensor nodes. Assuming flow $i$ is constrained by a leaky bucket with parameters $(\sigma_i, \rho_i)$, the maximum end-to-end delay, introduced by packet queuing and handling till full transmission, for a packet of flow $i$ under WFQ is given as shown in the Equation 4.2,
\[ D(i) \leq \frac{\sigma_i}{g(i)} + \sum_{m=1}^{M-1} \frac{P_{\text{max}}(i)}{g_{m}^n} + \sum_{m=1}^{M} \frac{P_{\text{max}}}{c} \] (4.2)

### 4.1.2 Weighted Fair Queue model

The WFQ queue model is specifically designed for the case of coexistence of real-time and non-real-time traffic in each sensor node. This model employed from inspired class-based queuing. Weighted Fair Queuing (WFQ) is a packet scheduling technique allowing guaranteed bandwidth services. Weighted Fair Queuing, the link’s bandwidth is distributed among competing flows proportionally to their weights.

Since in WFQ each flow has its own queue, considering each imaging sensor node as a source of different real-time-flow, however, only one real-time queue is used in the node to serve the data coming from these multiple flows. These mechanisms have an approximation to flow-based WFQ approach and are used due to two reasons: First, having a different queue for each real-time flow have been inefficient regarding the storage capacity of a sensor node. Second, the real-time flows are generated dynamically depending on the number of active imaging sensors. Since the number of such flows can change during the sensing activity, having one queue will reduce the maintenance overhead. Moreover also, dedicate another separate queue to serve non real-time data coming from different sources. The model is depicted in Figure 4.1. In this model, the service ration \( r_m \) for the real-time queue on a node will be the summation of link shares of all real-time flows passing through that particular sensor node. While delay-constrained delivery of real-time packets is achieved through WFQ, approach addresses energy awareness through the usage of a cost function that considers relevant energy factors at sensor nodes.
The advantage of WFQ is that it gives each flow different weight to have different bandwidth percentage in a way that preventing monopolization of the bandwidth by some flows providing fair scheduling for the different flows supporting variable length packets, WFQ provides protection for each service class by ensuring a minimum level of output port bandwidth independent of the behavior of other service classes. When combined with traffic conditioning at the edges of a network, WFQ guarantees a weighted fair share of output port bandwidth to each service class with a bounded delay.

The idea behind the algorithm is that for each packet, WFQ computes the time at which service to the packet would be finished, deploying a GPS scheduler. Then the WFQ scheduler services the packets in the increasing order of their finish times. In other words, WFQ simulates GPS on one side and uses the results of this simulation to determine the packets’ service order. It has been prove that WFQ have an excellent use in creating firewalls between classes but at the same time it does punishing flows or classes, in the case of DiffServ for using un-contended bandwidth which making it not suitable for DiffServ architecture. There are some enhancements have been done for WFQ although
all those algorithms are fair in the worst-case sense and tend to have low delay, they were not designed to provide service differentiation among classes in the context of DiffServ networks. It assumes that the server can serve all backlogged flows simultaneously and that the traffic is infinitely divisible. Most of the wireless sensor networks transmit the data in packet level only. Therefore, many packet fair queuing (PFQ) algorithms have been proposed to approximate GPS. Weight fair queuing (WFQ) is a representative scheme to approximate GPS. In WFQ, each packet is associated with a start tag $S_i(\cdot)$ and finish tag $F_i(\cdot)$, which correspond to the virtual times at which the first and the last bits of the packet, respectively, are served in GPS. At the time when the $k^{th}$ ($k \geq 1$) packet of flow $i$, denoted by $p_k^i$, arrives at the queue. Where $F_i(p_k^i = 0)$, $A(p_k^i)$ is the physical arrival time of packet $p_k^i$.

### 4.1.3 Optimization of Energy Efficiency

The responsibilities of the gateway node include sensor node organization and management of the network topology. Sensor node organization refers to the appropriate selection of the required sensors that need to be operational to achieve the application objective. The gateway sets up paths for collecting the sensor data. The gateway informs each sensor with its role, namely probing the environment, relaying data of some of these sensor nodes require dynamic adjustment of the network topology. The gateway broadcast the routing table to all sensors before starting or resuming data transmission. The gateway has use model-based energy consumption for the data processor, radio transmitter and receiver on board the sensor to track the life time of the sensor node’s battery. This model have been used in the link cost function. The gateway updates the sensor node energy model with each packet received by changing the remaining battery capacity for the nodes along the path from the source sensor node to the gateway.
The new routing approach has based on a two-step strategy incorporating both link-based costs and end-to-end delay constraints. First, designate some candidate paths without considering the end-to-end delay. Upon obtaining these candidate paths further check them to pick the least cost path that meets the end-to-end delay requirements and for which a feasible $r$-value for each sensor node have been found. Account for the limitation of sensor’s energy supply, formulates a link cost function that captures the effect of required transmission power and available energy reserve at the nodes. In addition to energy consideration and also account for the transmission error due noise and packet drop. The end-to-end delay requirement, as explained in the next subsection, is imposed on the whole path as a constraint. It has defined the following cost function for a link between nodes $i$ and $j$, as shown in the Equation 4.3.

$$
\text{cost}_{ij} = \sum_{k=0}^{2} CF_k = c_o X (\text{dist}_{ij})^t + c_1 X f(\text{energy}_j) + c_2 X f(e_{ij}) \quad (4.3)
$$

Where,

- $\text{Dist}_{ij}$ is the distance between the nodes $i$ and $j$,
- $f(\text{energy}_j)$ is the function of the current residual energy of node $j$,
- $f(e_{ij})$ is the function of the factors that affect the error rate on the link between $i$ and $j$.

Real time data route:-The real-time data entering to the gateway through QoS path, first obtain a set of energy-efficient paths. Then, we further check them to identify the one that can meet the end-to-end delay requirement by trying to find an $r$-value for each node on that path. Therefore, for each flow, the necessary service rate at each node should be estimated. Let $T_{\text{required}}$ be the required end-to-end delay for the application. Moreover, thus, find a set of values for the nodes on the path so that $D(i) \leq T_{\text{required}}$, using the Equation 4.2.
\[
\frac{\sigma_i}{g(i)} + \sum_{m=1}^{M-1} \frac{P_{\text{max}}(i)}{g_i^m} + \sum_{m=1}^{M} \frac{P_{\text{max}}}{C} \leq T_{\text{required}} \quad (4.4)
\]

To find \(g_i^m\) from the above equation, we assume that the service rate is same for all the nodes on the path of a particular flow, i.e. \(g(i) = g_i^m\) once \(g_i^m\) have calculated, then the link share can be calculated directly from the Equation 4.1.

\[
\Phi_i^m = \frac{g_i^m \sum_{j=1}^{N} \Phi_j^m}{C} \quad (4.5)
\]

The service ratio for real-time data on a sensor node, have the link shares of the summation of all flows passing through that sensor node, i.e., \(r_m = \sum_{j=1}^{N} \Phi_j^m\) where \(N\) is the number of real-time data flows passing through node \(m\) and \(\Phi_j^m\) is the link share for a real-time flow \(j\) on node \(m\). The gateway calculates this \(r\)-value and sends it to the corresponding node as shown in the Equation 4.4 to 4.5.

Considering the above calculations, the Algorithm 4.1, to find a least-cost path that meets the constraints for real-time data. The algorithm calculates the cost for each link, line 0 of Algorithm 4.1, for each flow from imaging sensors, the least cost path to the gateway is found by running a K-shortest path algorithm in line 3. Here, the least cost path is taken K alternatives by setting \(k\) to 1 in line 1. Lines 4-5, appropriate \(r\)-values are calculated for each real-time data flow. The line 6 and 7, the overall bandwidth spilled ration of each node on the path is checked while considering all flows through the node. If that the value is not between 0 and 1. Alternative paths with bigger costs tried by increasing the \(k\) value (line 2 and 3). Moreover, proper \(r\) - value is found, the loop exits (line 11). There is no such \(r\)-value, the connection request of the node to the gateway is rejected (line 12-13). Find the \(K\) least-cost paths. Since the algorithm finds a set of paths with similar nodes and links, modified the algorithm so that each time a new path is searched for a particular node, only node-disjoint paths are considered during the process. This ensures simplicity and helps in finding a proper \(r\)-value more easily since that node-disjoin path
will not inherit the congestion of path whose costs are less and for which the end-to-end delay as shown in the algorithm 4.1.

Algorithm 4.1 Pseudo code for setting $r$- values

0 calculate cost$_{ij}$, $\forall i, j \in V$
1 $k \leftarrow 1$
2 repeat
3 find $k^{th}$ least cost path for each node
4 for each flow from imaging sensor $i$ do
   begin
5 compute $\emptyset_i$ for all the nodes from $D(i) \leq T_{\text{required}}$
6 for each node $m$ (on the path of flow $i$) do
7 $r^m \leftarrow r^m + \emptyset_i$
8 if ($r^m$ is not in range $[0,1)$) then
9   break;
end
10 $k \leftarrow k + 1$
11 until $> K$ or a proper $r^m$ for each node is found
12 if no proper $r^m$ is found.
13 discard the connection

4.2 Mobility Gateway

Based on the applications the sensor gateway node to change its location dynamically. For instance, in a disaster management application, the gateway has the mobile or a moving emergency vehicle equipped with computing and communication devices. The gateway is mobile, ensuring uninterruptible data traffic and timely delivery of quality of service data to the gateway becomes a challenge. The main issue with a mobile gateway is the need for dynamic maintenance of the multi-hop routes to provide the desired service for the data as if the gateway is stationary. The mechanism to relay the data packets of that
node to the gateway area to the network, when the gateway goes out of the transmission range of sensor node that is within one hop distance to the gateway. Setting up totally new routes based on the gateway’s new area when needed setting up totally new routes have require frequent acknowledgment of the gateway’s to the network. Moreover, then cause significant overhead due to the increase in the volume of their radio on and off. Such overhead have eventually increased the energy consumption of the sensor nodes and can significantly drain out the sensor node’s battery life time.

Moreover, providing on-time delivery of real-time data in the case of a mobile gateway is much harder. Since real-time data is associated with end-to-end delay constraints, maintaining enough resources for real-time traffic while the gateway is on the move is crucial. Based on the new locations of the gateway, either new relay nodes with enough resources are found to keep the end-to-end delay within the acceptable range or the routes are set up from scratch. Again introduce extra traffic and hence increase the energy consumption of the sensor nodes.

Based on the observations, define the problem as follows: Interested in extending WFQ-based routing approach for efficient handling of gateway mobility. Given the area, speed of the gateway and end-to-end delay constraints for real-time data. While the gateway is on moving such that the number of lost packet drops is minimized and the hit ratio for real-time data is not diminished. In this chapter, described a new algorithm to extend the WFQ- based routing approach to handle gateway mobility.

4.3 Analytical Evaluation

An energy-aware approach for routing delay-constrained data pursues multi-hop packet relaying to minimize transmission energy and employs Weighted Fair Queuing (WFQ) packet scheduling methodology along with leaky bucket constrained data sources in order to provide soft real-time guarantees for
data delivery. The main idea is to watch for gateway reachability to sensor nodes and dynamically adjust the network topology in order to ensure timely data delivery. The energy efficient multi-hop routes set at the initial gateway location for both real-time and non-real-time traffic are used as long as the gateway is within the transmission range of last hop sensors i.e., sensors that are within one hop distance away from the gateway. When the gateway goes out of range of last hop sensors, it designates new sensor nodes as forwarders and therefore continual delivery of messages is maintained. The effectiveness of new energy-aware delay-constrained routing approach under stationary and mobile gateway architectures validated through ns-2 simulation and MATLAB.

4.3.1 Sensor network operation

The network operational model for evaluating approach. The gateway assumes responsibility for sensor node organization based on missions that are assigned to the sensor network. Mission-oriented organization of the sensor network enable the appropriate selection of only a subset of the sensors to be turned on and thus avoids wasting the energy of sensors that should not have to be involved. Thus, the gateway has controlling the configuration of the data processing circuitry of each sensor node.

The sensor nodes have one of four main states: sensing only, relaying only, sensing-relaying and inactive. In the sensing state, the node sensing circuitry is on and it sends data to the gateway at a constant rate. In the relaying state, the sensor node should not probe the environment but its communications circuitry is on to relay the data from other active nodes. It is considered in the sensing-relaying state when the node is both sensing the target and relaying messages from other nodes. The sensor node has considered inactive and switch to a low power sleep mode. The decision for determining the node’s state done at the gateway based on the current sensor organization, node battery levels and desired network performance measures.
The typical operation of the sensor network consists of two alternating cycles: data cycle and routing cycle. During the data cycle, the nodes, which are sensing the environment sends their data to the gateway. In 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. Rerouting is performed after a specified number of data cycles, which refer as reroute the data. Rerouting is triggered by an application-related event requiring a different set of sensors to probe the environment including the activation of new sensors based on the gateway movement or the depletion of the battery of an active node. A Time Division Multiple Access (TDMA) based MAC protocol is used. The gateway manages slot assignment based on the network topology. The gateway informs each node about slots on which it should listen to other nodes transmission and slots, which the node can use for its own transmission.

4.3.2 Simulation setup

In the simulations, the wireless sensor network consists of 200 randomly placed sensor nodes deployed in a 1000x1000 m$^2$ area. The gateway initial position is determined randomly within the region bounders. The free space propagation channel model is assumed with the capacity set to 4 Mbps. Packet lengths are 20 Kbit for data packets and 4 Kbit for routing packets. Each node is assumed to have an initial energy of 10 joules. A sensor node has considered non-functional if its energy level reaches zero. The maximum transmission range for a sensor node is assumed to be 100 meters. For a sensor node in the sensing state, packets are generated at a constant rate of 2 packet/sec. Assumed to be leaky bucket constrained with the maximum burst parameter $\sigma$ of 20 packets of the source generating data. Each data packet is time-stamped when it is generated to allow the calculation to calculate the average energy per packet since our cost function defined for each link is using remaining energy as part of the cost.
Moreover, assume that the wireless sensor network has tasked with a target-tracking mission in the analytical and the gateway is moving based on the Random Waypoint Model with a maximum speed of 7 m/s unless otherwise specified. The initial set of sensing nodes is chosen to be the nodes on the convex hull of sensors in the deployment area. Changing the set of sensing nodes as the moving of target. The sensing circuitry of all boundary nodes always turned on since targets assumed to come from outside the area. The sensing circuitry of the other nodes is usually turned off but can be turned on according to the target or gateway movement. Then assume that each sensor node is capable of taking the image of a target to the target or gateway movement. Moreover, also assume that each sensor node is capable of taking the image of a target to identify it clearly and can turn on its imaging capability on demand. During simulation, small subsets of currently active nodes, which are the closest nodes to the target, are selected to turn on their imaging capabilities. Therefore, the imaging sensor set may change with the movement of the event.

The packet generation rate for imaging sensor nodes is bigger than the normal sensors. Packets, generated by imaging sensors, are labeled as a real-time type and treated differently at the relaying nodes. The $r$ value is initially assumed to be zero, but it is recalculated as imaging sensors get activated. The default end-to-end delay requirement for real-time data is taken to be 0.06 sec. Events assumed to start at a random position outside the convex hull. Characterized by the events have the constant speed chosen uniformly from the range 5 m/s to 6 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. It is assumed that only one target is active at a time. This event remains active until it leaves the deployment region area triggering the generation of a new event.
4.4 Results and Discussions

In this section, presents performance results obtained through ns-2 simulation and an analytical Equation 4.1 and Equation 4.2 are verified by MATLAB tool. Finally the empirical values are taken on running simulation as a baseline approach, comparing these two results, used the same cost function with same routing algorithm without doing any service differentiation. That is, only one queue in each sensor node for all kinds of packets. Therefore, no bandwidth allocation is done through adjusting of $r$ values.

Comparing the average delay per real-time packet achieved through WFQ based model with the average delay per packet obtained in the single queue model and observed that the service differentiation employed using WFQ based approach achieves substantially better average delay for both mobile and stationary gateway models as shown in the Figure 4.2. There is an increase in the average delay per real-time packet when the real-time data rate is boosted. This is due to the increase in the queuing delay that the real-time packets incur. Such
increment in the average delay per real-time packet is very small when using the WFQ-based approach. Although the average delay per real-time packet from distant nodes increases, nodes close to the gateway, which have less average delay, also generate more real-time packets, causing the overall average delay for real-time packets to stabilize. In the case of a mobile gateway, the results in Figure 4.2 shown that the new approach for handling mobility slightly increases the average delay per real-time packet since the routes are adjusted by adding new hops introducing an extra delay of each packet.

WFQ approach achieves almost 90% on-time delivery with the stationary gateway and 80% with mobile gateway when there is no congestion in the network, applying the baseline approach causes most of the packets to miss their deadlines. It is interesting to note that the amount of growth in average delay due to gateway mobility is stable under different loads in comparison to a stationary gateway. Also, the effect of gateway mobility becomes marginal under heavy load.

![Figure 4.3 Average energy per packet](image-url)
Energy consumption represents in the Figure 4.3 depicts the impact of the generation rate of real-time data on energy usage, measured regarding average energy per packet. Moreover, observed that for moderate real-time data rates, routes set using our algorithm lead to the consumption of almost the same amount of energy compared to the baseline approach. As the real-time data grows, the gap between the quality of service routing and baseline curves widens. This is due to not being able to consistently use the least cost path while meeting end-to-end delay constraints.

The average energy consumed per packet in the case of a mobile gateway is higher than that for a stationary gateway as shown in the Figure 4.3. Because the transmission power of last hop sensors gets increased during the gateway motion to sustain the reach ability to the gateway at a further distance. The average energy is 37.5% at the moderate data 7 between stationary and mobile gateway. However, the mobile gateway and baseline are almost same, nearly 3.5% variation. Moreover, any extra hop added when the gateway is out of range will introduce new transmission and reception energy for a packet. Again the increment in energy usage to handle gateway mobility is very stable and becomes relatively marginal at high data rates.

To investigate the effect of reroute frequency and have varied the reroute period, i.e., the number of data phases to pass before the next reroute. And also studied how such effect relates to the number of sensors. Such effect is very much expected. Moreover, observed from Figure 4.4 that when the number of the sensor increases significantly, the overhead of rerouting becomes dominant in total energy consumption. The effect of reroute on the energy consumption for a mobile gateway based setup is presented in Figure 4.5. The observed results indicate that the average energy per packet grows, in comparison to the case of a stationary gateway, when the duration between successive rerouting is increased. Considered the effect of the different reroute periods on the delay and timeliness metrics.
As indicated by Figure 4.6, the average delay per real-time packet grows when the frequency of rerouting get lower. Such observation is expected since a longer reroute period means more forwards are engaged increasing the number
of hops for the routes. Such increases hop adds extra delay for the packets, especially when the forwarder is not dedicated to one path. In such case of a stationary gateway, the average delay per real-time packets missing their deadline increases in the case of a mobile gateway when the reroute period increases.

Figure 4.6 Average delay per real-time packet

Figure 4.7 Comparison results of average delay
The results of proposed and existing work as shown in the Figure 4.7 of average delay of network, proposed work is lesser delay than existing work. Which considers a different queue for each incoming flow and has been shown to provide, in statistical term, an upper bound on path delay for a leaky bucket constrained flow? It can regulate the incoming traffic from the sources by using the leaky-bucket traffic-regulation mechanism and separate the real-time (delay-constrained) traffic from non-constrained traffic with the usage of two different queues in each node. Once such separation is achieved, the service rate for the real-time data queue is estimated so that most deadlines are met. Further extend the proposed routing mechanism to provide uninterrupted message delivery for a mobile gateway. The main idea is to watch for gateway’s reachability to sensor nodes and dynamically adjust the network topology in order to ensure timely data delivery. As shown in the Figure 4.8, the energy efficient multi-hop routes set at the initial gateway location for both real-time and non-real-time traffic are used as long as the gateway is within the transmission range of last hop sensors i.e., sensors that are within one hop distance away from the gateway. The proposed works is lesser energy consumption than existing work and as well as
the number of nodes is higher than the existing work. When the gateway goes out of range of last hop sensors, it designates new sensor nodes as forwarders and therefore continual delivery of messages is maintained. Such process continuous until establishing new routes is deemed necessary. Rerouting is triggered when the end-to-end delay is not acceptable or current routes become instable due to the exhaustion of the battery of some nodes on the data paths. From the Figure 4.7 and Figure 4.8 approaches balance the energy and timeless goals of the network of proposed work and prevent the potential excessive topology management overhead.

4.5 Summary

This chapter concludes that the energy-aware delay-constrained routing approach for sensor networks. This approach finds energy-efficient paths for real-time and non-real-time flows. It has employed to support both best effort and delay-constrained traffic. Since WFQ provides per flow upper bounds on end-to-end delay, each sensor node generating real-time data is considered as a different flow. However, rather than employing a different queues and link share for each real time flow, a single queue is used to accommodate all real-time data from different flows. Service rate estimation mechanism is used along with a leaky bucket constrained packet generation model so that end-to-end delay bounds are met.

Based on the observations from figures discussed in this subsection, Have been concluding that a routing protocol that handles the mobility of the gateway should consider minimizing the number of reroutes i.e., increasing the number of reroute period without negatively affecting the delivery of real-time packets on time. Moreover, fast moving gateways have negatively affected the network efficiency and even make it impossible to get on-time data delivery. WFQ approach for providing energy-aware delay-constrained routing with a mobile gateway has adjusted through tuning the reroute period to give the best
performance regarding energy usage. Finding new sensor nodes within the communication range of the gateway to relay the packets and the routes are updated by adding those new hops. In this approach is further extended to support a mobile gateway moves out of the range. If the gateway moves close to the source nodes, it overhears, it overhears on-going traffic to limit the end-to-end delay and save unnecessary relaying.