Stability And Delay Analysis For Multi-Hop Single-Sink Wireless Sensor Networks

M. Farukh Munir*, Arzad A. Kherani** and F. Filali*
*Institut Eurcom, Department of Mobile Communications, Sophia Antipolis, France
**Indian Institute of Technology Delhi, New Delhi, India.
Email: {munir,filali}@eurecom.fr
alam@cse.iitd.ernet.in

Abstract

Wireless sensor networks are commonly used to monitor and control the physical world. To provide a meaningful service such as disaster and emergency surveillance, meeting real-time constraints and the stability of transmit queues are the basic requirements of communication protocols in such networks.

In this paper, we propose a closed architecture with two transmit queues at each sensor i, i.e., one for its own generated data, and the other for forwarding traffic. Our first main result concerns the stability of the forwarding queues at the nodes. It states that whether or not the forwarding queues can be stabilized (by appropriate choice of weighted fair queueing weights) depends only on routing and channel access rates of the sensors. Further, the weights of the weighted fair queues play a role in determining the tradeoff between the power allocated for forwarding and the delay of the forwarding traffic. We finally propose a distributed routing scheme for a broad class of wireless sensor networks. Each link is assigned a weight and the objective is to route through minimum weight paths using iterative updating scheme. The proposal is validated by analytical analysis and simulations.

I. Introduction

Distributed systems based on networked sensors with embedded computation capabilities enable an instrumentation of the physical world at an unprecedented scale and density, thus enabling a new generation of monitoring and control applications. Such networks consist of large number of distributed sensor nodes that organize themselves into a multihop wireless network. Each node has one or more sensors, embedded processors, and low-power radios, and is normally battery operated. Typically, these nodes coordinate to perform a common task.

In this paper, we propose a closed architecture for data sampling (application layer) in a wireless sensor network. We consider a new data sampling scheme: Node i, $1 \leq i \leq N$, has two queues associated with it: one queue Q_i contains the data sampled by the sensor node itself and the other queue F_i contains packets that node i has received from any of its neighbors and has to be transmitted to another neighbor. In this architecture, there is coupling between the sampling process and the channel access scheme. The objective in the closed architecture is to study the impact of channel access rates, routing, and weights of the weighted fair queues on system performance. Furthermore, a distributed routing algorithm (which is allowed to split flows) is proposed that maintains the system at a Wardrop equilibrium and guarantees low delay.

The organization of this paper is as follows. In Section II, we formulate the problem. Section III briefly describes the network model and assumptions underlying this study. In Section IV, we detail the data collection mechanism and the stability analysis of this system. Sections V presents a distributed learning scheme that uses routes with the smallest delay. Results from the implementation are presented in Section VI. Finally, Section VII concludes the paper and outlines the future work in this direction.

II. Problem Formulation

We consider a set of static sensors spread over a region to perform sensing operation. Each of these sensors has a wireless transceiver that transmits and receives at a single frequency which is common to all these sensors. Over time, some of these sensors generate/collect information to be sent to some other sensor(s). Owing to the limited battery capacity of these sensors, a sensor may not be able to directly communicate with far away nodes. In such scenarios, one of the possibilities for information transfer between two nodes that cannot communicate directly is to use other sensor nodes in the network. To be precise, the source sensors transmits its information to one of the sensors which is within its transmission range. The intermediate sensor then uses the same procedure so that the information finally reaches its destination (a fusion center, i.e., a common sink).

A set comprising of ordered pair of nodes constitute a *route* that is used to assist communication between any two given pair of nodes (i.e., a sensor and a sink). This is a standard problem of *multihop* routing in wireless sensor networks. The problem of optimal routing has been extensively studied in the context of wireline networks where usually a shortest path routing algorithm is used: each link in the network has a *weight* associated with it and the objective of the routing algorithm is to find a path that achieves the minimum weight between two given nodes. Clearly, the outcome of such an algorithm depends on the assignment of weights associated to each link in the network. In wireline context, there are many well-studied criteria to select these weights for links, e.g., the queueing delay etc. In WSNs, the optimality in the routing algorithm is set to extend network lifetime (where lifetime is defined as the time spanned by the network for some data aggregation till first node death due to energy outage) in a single sink network. In networks with multiple sinks [1], the flow is splitted and sent to different basestations with the aim of extending the network lifetime of these limited battery sensor networks. *However, a complete understanding of effect of routing on WSN performance and resource utilization (in particular, the stability of transmit buffers and hence, the end-to-end delay and throughput) has not received much attention.*

In this work, we assume a wireless sensor network that is deployed on a remote location and is representative of some collection/aggregation of data generated in the network. We consider the random access mechanism for wireless channel where the nodes (having packets to be transmitted in their transmit queue) attempt transmissions by delaying the transmission a random amount of time. This mechanism acts as a way to avoid collisions of transmissions of nearby sensors in the case where sensors cannot sense the channel while transmitting. We assume that the time is slotted into fixed length slots. In any slot, a node (provided it has packets to be transmitted) decides with a fixed probability to make a transmission attempt. If there is no other transmission by the sensors whose transmission can interfere with the one under consideration, the transmission is successful.

At any instant of time, a sensor may have two types of packets to be transmitted:

1) Packets sensed/generated by the sensor itself.
2) Packets from other neighboring sensors that arrived at this sensor and need to be forwarded.

Clearly, a sensor needs to have some scheduling policy to decide on which type of packet it wants to transmit, if it decided to transmit. A first come first served scheduling is one simple option. Another option that we would be considering in this paper is to have two separate queues at each sensor node and do a weighted fair queueing for these two queues. In this paper, we will study the effect of channel access probability, weights of the weighted fair queueing, and routing on stability and fairness properties of the WSNs.

III. Network Model

In this paper, we consider a static wireless sensor network with N sensor nodes.

Neighborhood relation model: Given is an $N \times N$ neighborhood relation matrix $\mathcal{N}$ that indicates the node pairs for which direct communication is possible. We will assume that $\mathcal{N}$ is a symmetric matrix, i.e., if node i can transmit to node j, then j can also transmit to node i. For such node pairs, the $(i,j)^{th}$ entry of the matrix $\mathcal{N}$ is unity, i.e., $\mathcal{N}_{i,j} = 1$ if node i and j can communicate with each other; we will set $\mathcal{N}_{i,j} = 0$ if nodes i and j can not communicate. For any node i, we define $\mathcal{N}_i = \{j : \mathcal{N}_{i,j} = 1\}$, which is the set of neighboring nodes of node i.

Sampling Process (application layer in wireless sensor networks): Each sensor node is assumed to be sampling (or, sensing) its environment at a predefined rate; we let λ_i denote this sampling rate for node i. The units of λ_i will be packets per second, assuming same packet size for all the nodes in the network.

Forwarding (Relaying): Each sensor node wants to use the sensor network to forward its sampled data to a *common* fusion center (assumed to be a part of the network). Thus, each sensor node acts as a forwarder of data from other sensor nodes in the network. This motivation is taken from the model of [2].

Traffic Model: We let ϕ denote the $n \times n$ routing matrix. The $(i,j)^{th}$element of this matrix, denoted $\phi_{i,j}$, takes value in the interval $[0,1]$. This means a probabilistic flow splitting (Flow splitting provides an extra degree of freedom to utilize available routes in a *fair* manner) as in the model of [2], i.e., a fraction $\phi_{i,j}$ of the traffic *transmitted* from node i is forwarded by node j. Clearly, we need that ϕ is a stochastic matrix, i.e., its row elements sum to unity. Also note that $\phi_{i,j} > 0$ is possible only if $\mathcal{N}_{i,j} = 1$.

Channel Access Mechanism: We assume that the system operates in discrete time, so that the time is divided into (conceptually) fixed length slots. We also assume that the packet length (or, transmission schedule length) is fixed throughout system operation. The system operates on CSMA/CA MAC. Assuming that there is no exponential back-off, the channel access rate of node i (if it has a packet waiting to be transmitted in either Q_i or F_i) is $0 \leq \alpha_i \leq 1$ (to avoid *pathological* cases). Thus, α_i is the probability that node i, if it has a packet to be transmitted, attempts a transmission in any slot. A node can receive a transmission from its neighbor if it is not transmitting and also no other neighboring node is transmitting, i.e., if the transmission is meant for some node j, $j \in \mathcal{N}_i$, then the transmission from node i to node j is successful iff none of the nodes in the set $j \cup \mathcal{N}_j \backslash i$ transmits.

Packet Loss: In this paper, we assume that the queues at sensors are large enough to avoid packet drop due to buffer overflows. We only consider packet losses arising from the excessive number of repeated collisions of a transmitted packet. If a packet is attempted transmissions M number of times by a node and has suffered a collision every time, the packet is dropped.

IV. Stability Analysis Of Our Data Collection Mechanism

In this paper, we consider the following data collection mechanism:

Closed System with two-Queues: Under this mechanism, there is a coupling between the channel access process and the sampling process. The closed system presented here is entirely different from the one in [2]. The combined channel access/data sampling mechanism is as follows: Node i decides to attempt a channel access with probability α_i in any slot (else, it is sensing the channel for any possible transmissions). If decided to attempt a transmission, the node first checks the number of packets available to be forwarded, i.e., packets from other nodes that node i is having to be forwarded to some of its neighbors. We have following possibilities:

1) Both F_i and Q_i are empty: In this case, the MAC layer of node i will ask the appropriate upper layer to sense data and provide it with a new packet. This packet is then attempted a transmission.
2) At least one packet waiting to be forwarded: In this case, node i will do the following:
    a) with probability $1 - f_i$, ask the appropriate upper layer to sense data and provide it with a new packet. This packet is then attempted transmission.
    b) with probability f_i, forward the head-of-line packet waiting to be forwarded.

We assume that the queue Q_i is always nonempty, i.e., nodes make new measurements as soon as the older ones are transmitted. *Note* that this kind of model with assumption of saturated nodes are intended to provide insights into the performance of the system and also helps study effects of various parameters.

It is to be noted that this system can also be thought of as the one in which the sensor node always have a backlog of their own sampled data. We briefly give the correct stability condition for our system. We fix a node i and look at its forwarding queue F_i. It is clear that if this queue is stable then the output rate from this queue is equal to the input rate into the queue. Only issue to be resolved here is to properly define the *output rate*. This is because, owing to the bound M on maximum number of attempts for transmission of any packet, not all the packets arriving into F_i may be successfully transmitted. Therefore, the output rate is defined as the rate at which packets are either successfully relayed or dropped due to excessive number of collisions. We start by obtaining the detailed balance equations, i.e., the fact that if the queue F_i is stable, then the output rate from queue F_i is equal to the input rate to this queue. Let

$$S_i = \sum_{j \in \mathcal{N}_i} \phi_{i,j} \left(1 - \alpha_j\right) \prod_{k \in \mathcal{N}_{j \backslash i}} \left(1 - \alpha_k\right)$$

be the probability that a transmission from node i is successful. Also let

$$\begin{aligned} E_i = \textstyle\sum_{m=1}^{M} m \left(1 - S_i\right)^{m-1} S_i + M \left(1 - S_i\right)^{M} \\ = \tfrac{1 - \left(1 - S_i\right)^{M}}{S_i} \end{aligned}$$

be the expected number of attempts till success or M consecutive failures of a packet from node i.

Proof: From theory, we have that $\sum_{m=1}^{M} \left(1 - S_i\right)^{m-1} S_i = 1 - \left(1 - S_i\right)^{M}$. Taking the derivative of L.H.S and solving, we get

$$= \sum_{m=1}^{M} \left[\left(1 - m\right)\left(1 - S_i\right)^{m-2} S_i + \left(1 - S_i\right)^{m-1}\right]$$

$$\begin{aligned} = \textstyle\sum_{m=1}^{M} \left(1 - S_i\right)^{m-1} S_i - \sum_{m=1}^{M} m \left(1 - S_i\right)^{m-1} S_i \\ + \textstyle\sum_{m=1}^{M} \left(1 - S_i\right)^{m-1} \end{aligned}$$

Similarly, the R.H.S gives us $M \left(1 - S_i\right)^{M-1}$. Multiplying both sides by $\left(1 - S_i\right)$ and solving, we get

$$\begin{aligned} \textstyle\sum_{m=1}^{M} \left(1 - S_i\right)^{m-1} S_i - \sum_{m=1}^{M} m \left(1 - S_i\right)^{m-1} S_i \\ + \textstyle\sum_{m=1}^{M} \left(1 - S_i\right)^{m} = M \left(1 - S_i\right)^{M} \end{aligned}$$

$$\begin{aligned} \textstyle\sum_{m=1}^{M} m \left(1 - S_i\right)^{m-1} S_i + M \left(1 - S_i\right)^{M} = \\ \textstyle\sum_{m=1}^{M} \left(1 - S_i\right)^{m-1} S_i + \sum_{m=1}^{M} \left(1 - S_i\right)^{m} \end{aligned}$$

$$= \frac{S_i \left(1 - \left(1 - S_i\right)^{M}\right) + \left(1 - S_i\right)\left(1 - \left(1 - S_i\right)^{M}\right)}{S_i}$$

$$= \frac{1-(1-S_i)^M}{S_i}$$

Lemma 1: For a given routing, let π_i denote the probability that a node i has packets to forwarded, then the long term average rate of departure of packets from node $i's$ forwarding queue is

$$\pi_i \alpha_i f_i E_i. \tag{1}$$

Proof: Let T_t be an indicator function which is unity if F_i is nonempty. Let I_t be an indicator function that $T_t = 1$ and a transmission is made from F_i (it can be a success or a failure). Then the output rate from $F_i$of packets is then

$$\lim_{t\to\infty} \frac{1}{t}\sum_{l=1}^{t} I_l = \lim_{t\to\infty} \frac{\sum_{l=1}^{t} T_l}{t} \lim_{t\to\infty} \frac{\sum_{l=1}^{t} I_l}{\sum_{t=1}^{l} T_l}.$$

Since we are working under assumption that node i attempts forwarding of any packet at most M times, we have, with probability one,

$$\lim_{t\to\infty} \frac{\sum_{l=1}^{t} I_l}{\sum_{t=1}^{l} T_l} = \alpha_i f_i E_i$$

Also, with probability one,

$$\lim_{t\to\infty} \frac{\sum_{l=1}^{t} T_l}{t} = \pi_i.$$

Clearly, the long term output rate from the queue F_i is, with probability one,

$$\lim_{t\to\infty} \frac{\sum_{l=1}^{t} I_l}{t} = \lim_{t\to\infty} \frac{\sum_{l=1}^{t} T_l}{t} \lim_{t\to\infty} \frac{\sum_{l=1}^{t} I_l}{\sum_{t=1}^{l} T_l} = \pi_i \alpha_i f_i E_i.$$

Lemma 2: The long term average rate of arrival of packets into F_i is

$$\sum_{j\in\mathcal{N}_i} \phi_{j,i}\left(\alpha_j E_j\right)$$

The proof for average rate of arrival is straight forward in the sense that i can only receive packets from $j, j \in \mathcal{N}_i$. $\phi_{j,i}$ is amount of traffic on link (j,i). α_j is the probability with which j is transmitting and E_j is the expected number of attempts of packet till success or M consecutive failures.

Proposition 1: In the steady state, if all the queues in the network are stable, then for each i

$$\pi_i \alpha_i f_i E_i = \sum_{j\in\mathcal{N}_i} \phi_{j,i}\left(a_j E_j\right) \tag{2}$$

Proof: If the queue F_i is stable, then the rate of arrival of packets into the queue is the *same* as the rate at which the packets leave the queue. Let $w_{j,i} = \frac{\sum_j \phi_{j,i}(a_j E_j)}{\alpha_i E_i}$ and $y_i = 1 - \pi_i f_i$ (transmission probability from Q_i). Note that $w_{j,i}$ is independent of f_j, $j \in \mathcal{N}_i$ and depend only on the α_j and routing.

In the steady state, if all the queues in the network are stable, then we can write for each i

$$1 - y_i = \sum_{j\in\mathcal{N}_i} w_{j,i} \tag{3}$$

The relation of eq. (3) has some interesting interpretations. Some of these are:

<u>The Effect of f_i :</u> The quantity $y_i = 1 - \pi_i f_i$ is independent of the choice of f_j, $j \in \mathcal{N}_i$. It only depends on the routing and the value of α_j.

<u>Stability:</u> Since the values of y_i are independent of the values of f_j, $j \in \mathcal{N}_i$, and since we need $\pi_i < 1$ for the forwarding queue of node i to be stable, we see that for any value of $f_i \in (1 - y_i, 1)$, the forwarding queue of node i will be stable, Thus we obtain a lower bound on the weights given to the forwarding queues at each node in order to guarantee stability of these queues. To ensure that these lower bounds are all feasible, i.e., are less than 1, we need that $0 < y_i \le 1$; $y_i = 0$ corresponds to the case where F_i is unstable. Hence, if the routing and $\alpha_j's$ are such that all the y_i are in the interval $(0, 1]$, then all the forwarding queues in the network can be made stable by appropriate choice of $f_i's$. Now, since y_i is determined only by routing and the probabilities $\alpha_j's$, we can then choose f_i (thereby also fixing π_i, hence the forwarding delay) to satisfy some further optimization criteria so that this extra degree of freedom can be exploited effectively.

<u>Throughput:</u> We see that the long term rate at which node i can serve its own queue is $\alpha_i\left(1 - \pi_i f_i\right) = \alpha_i y_i$, which is independent of f_j. Also, the throughput, i.e., the rate at which the packets reach the destination, i.e., $\alpha_i E_i$ is independent of f_j. Similarly, the long term rate at which the packets from the forwarding queue at node i are attempted transmission is $\pi_i \alpha_i f_i = \alpha_i\left(1 - y_i\right)$, which is also independent of the choice f_j, $j \in \mathcal{N}_i$.

<u>Throughput-Stability Tradeoff:</u> In the present case, we can tradeoff throughput with stability and not directly with delay. Let $\pi_i f_i = c$, if $c > 1$, $\forall i$ simultaneously, the system is unstable. We know that the throughput at node i is $1 - \pi_i f_i$. Then, if a node tries to maximize its own throughput, it is actually minimizing c, thus trying to stabilize the system. This is an interesting property in itself.

<u>Choice of f_i:</u> Assume that we restrict ourselves to the case where $f_i = P_f$, $\forall i$. Then, for the stability of all nodes, we need that

$$P_f > 1 - \min_i y_i.$$

Since the length of interval that f_i is allowed to take is equal to y_i, we will also refer to y_i as the stability region.

Energy-Delay Tradeoff: For a given set of $\alpha_i' s$ and routing, the throughput obtained on a route $\mathcal{R}_i$ is fixed, independent of f_i. Hence, there is no throughput-delay tradeoff obtained by changing f_i. However, we do obtain energy-delay tradeoff. For a given *stable* routing, we need f_i, which will determine π_i. Clearly, f_i represents the *forwarding-energy* and π_i gives a measure of *delay*. Therefore, the service rate given to F_i determines the *exact* energy-comsumption and delay for relaying, and hence, we can perform an *exact analysis* of the effect of different network parameters on performance in multi-hop wireless sensor networks.

V. Delay Analysis And Routing Algorithm

In this paper, we allow for traffic split and then try to route the traffic, hoping for a better performance (as the situation without traffic split is a special case of traffic splitting). Under this added freedom of traffic splitting, the routing algorithm is expected to put traffic of a node on those routes for which the delays are smallest and equal. This is what is well known as the Wardrop equilibrium, first appeared in [3]. We propose a stochastic approximation algorithm based distributed algorithm to converge to a Wardrop equilibrium. Under the above model there will be a delay, say $\tau_{j,i}$ of the packet from node j to be served at node i; this packet could have originated at node j or may have been forwarded by node j. The Expected-delay of a packet transmitted from node j is thus $\sum_{i \neq j} \phi_{j,i} \tau_{j,i}$. Since delays are additive over a path, packets from any node will have a delay over any possible route to the fusion center. A route will be denoted by an ordered set of nodes that occur on that route, i.e., the first element will be the source of the route, the last element will be the fusion center and the intermediate elements will be nodes arranged in the order that a packet traverses on this route. Let the total number of possible routes (cycle-free) be R. Let route i, $1 \leq i \leq R$ be denoted by the set $\mathcal{R}_i$ consisting of R_i elements with $\mathcal{R}_{i,j}$ denoting the j^{th}entry of this route. Then, a traffic splitting matrix will correspond to a Wardrop equilibrium iff for any i

$$\begin{gathered}\sum_{1 \leq j \leq R : \mathcal{R}_{j,1} = i} \left(\prod_{k=1}^{R_j - 1} \phi_{\mathcal{R}_{j,k}, \mathcal{R}_{j,k+1}} \right) \\ \left(\sum_{k=1}^{R_j - 1} \tau_{\mathcal{R}_{j,k}, \mathcal{R}_{j,k+1}} \right) = \sum_{k=1}^{R_l - 1} \tau_{\mathcal{R}_{l,k}, \mathcal{R}_{l,k+1}} ,\end{gathered} \tag{4}$$

for any l with $\mathcal{R}_{l,1} = i$ and such that $\prod_{k=1}^{R_l - 1} \phi_{\mathcal{R}_{l,k}, \mathcal{R}_{l,k+1}} > 0$, i.e., *the delays on the routes that are actually used by packets from node i are all equal*. The objective now is to come up with an algorithm using which any node (say i) is able to converge to the corresponding row of the matrix ϕ corresponding to the Wardrop equilibrium.

Closed System with two-Queues: The nodes iteratively keep updating the one-hop routing probabilities based on the delays incurred for every possible path. Let $\phi(n)$ denote the traffic splitting matrix at the beginning of the n^{th} time slot. Node i does some computation to update the i^{th} row of this matrix. Let $\delta^k(n)(\mathcal{R}_{k,1} = i)$ be the new value of the delay of a packet sent by sensor i through route $k(i = \mathcal{R}_{k,1})$. Node i keeps an estimate of the average delay on route k.

$$\tau^k(n+1) = (1-\zeta)\tau^k(n) + \zeta\delta^k(n). \tag{5}$$

Further, after calculating the expected delays at the start of a time slot, each node adapts its routing probabilities to the new expected delays as [4]. For the convergence of our routing algorithm in practice, we need that the probabilities $\phi_{i,j}$ are strictly positive for all feasible routes to ensure that we are able to probe for a change in the state of all the available routes.

VI. Implementation Results

We consider a 6-node sensor network shown in Fig. 1. *We consider this simple network to clearly demonstrate the stability region*. The transmit queue of node i can have multiple packets in the transmit queue (both Q_i ,i.e, self generated, and F_i, i.e., those packets that were initially generated at some other node, and have arrived at node i to be forwarded to some other node). Therefore, we need to implement two-queues at the MAC layer for sensor nodes for prioritizing traffic (based on the appropriate weights given to Q_i and F_i). We have implemented the Closed system with two-queues as a *cross-layer* (application-mac) module in TinyOS [5]. The routing layer is initiated with the minimum-hop routing, which is updated during the network lifetime according to the algorithm proposed in Section V. In this section, we present the simulation results once the neighbors are discovered and routes are established toward the fusion center. We have utilized the TOSSIM simulator of TinyOS to validate our proposals. All simulation runs for 10^8, seconds.

We present in Table I, the results on stability region and throughput for sensors 1, 2, and 4 as sensors 3 and 5 do not forward any traffic and y_i for $i = 3, 5$ is set to 1.

In order to demonstrate the results on *delay-and-stability* together using a closed-system with two-queues, we have implemented a 50-nodes sensor network with a common sink. In Fig. 2 we plot, against the slot number, the average delays for our closed-system with two-queues and single-queue system. The data sampling rates were set at $\lambda_i \leq 0.1$, $\forall i$. Note that the data sampling rates are small. We were forced to select small data rates in order to guarantee stability of the nodes in the network.

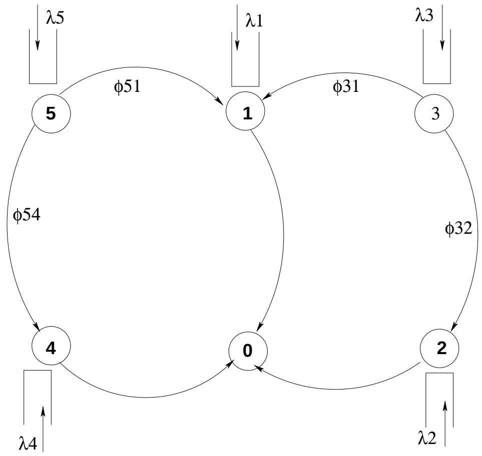

Fig. 1. The simulated network: 5 sensors and 1 fusion center.

TABLE I. Throughput and stability region

$\alpha' s$	Throughput			y		
Nodes→	1	2	4	1	2	4
0.0	0.00	0.00	0.00	1.0	1.0	1.0
0.1	0.50	0.48	0.51	0.88	0.91	0.90
0.2	0.80	0.75	0.79	0.80	0.82	0.85
0.3	0.95	0.92	0.96	0.70	0.71	0.74
0.4	0.85	0.83	0.85	0.72	0.75	0.78
0.5	0.64	0.60	0.62	0.76	0.80	0.82
0.6	0.45	0.45	0.40	0.81	0.83	0.85
0.7	0.32	0.29	0.30	0.85	0.86	0.89
0.8	0.10	0.11	0.11	0.97	0.98	1.00
0.9	0.01	0.02	0.01	1.00	1.00	1.00
1.0	0.00	0.00	0.00	1.00	1.00	1.00

Observations from the Simulations: The average delays on routes in two-queues closed system are very small compared to single-queue system. This is due to the appropriate choice of weights given to both F_i and Q_i(as discussed in Section IV) compared to the single queue system where we do not have the service differentiation. The routing schemes (Section V) allows both systems to pick the shortest-delay paths based on delay estimates. These results comply with our motivation that service differentiation at MAC layer results in better over all performance of the system and can help study the impact of different network parameters on its performance.

VII. Conclusions and Future Work

For wireless sensor networks with random channel access, we proposed a data sampling approach that guarantees a *long term* data sampling rate while minimizing the end-to-end delays. We have also obtained some important insights into various tradeoffs that can be achieved by varying certain network parameters. Some of them include:

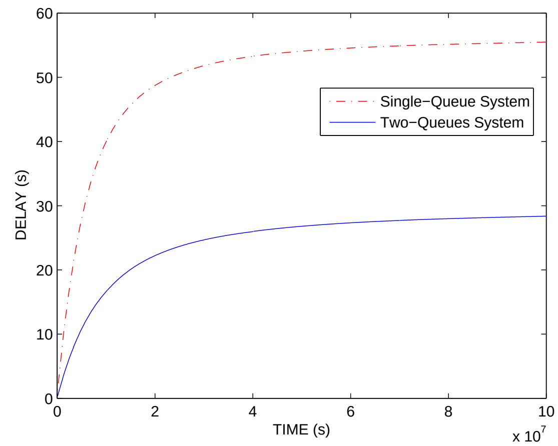

Fig. 2. Average delays for two-queues vs. single-queue system

1) Routing can be crucial in determining the stability properties of the networked sensors. 2) Whether or not the forwarding queues can be stablilized (by appropriate choice of WFQ weights) depends only on routing and channel access rates 3) We have also seen that the end-to-end throughput is independent of the choice of WFQ weights. We therefore, proposed a distributed learning algorithm to achieve Wardrop equilibrium for the end-to-end delays incurred on different routes from a sensor node to a fusion center (sink). From the implementation results, we have seen a very high delay for a single-queue system (provided the system was stable) compared to two-queues system.

In future, we will present a detailed implementation study of the two-queues closed system in comparison with the single-queue system to show the impact of different network parameters on system performance.

References

[1] J.-H. Chang and L. Tassiulas. Energy conserving routing in wireless ad hoc networks. In IEEE Infocom Conference Proceedings, March 26-30, 2000.

[2] M. Farukh Munir, Arzad A. Kherani, and F. Filali. On stability and sampling schemes for wireless sensor networks. In IEEE/IFIP WiOpt Conference Proceedings, April 16-20 2007.

[3] J. G. Wardrop. Some theoratical aspects of road traffic research. In Proceedings Inst. Civil Eng., Part 2:325-378, 1952.

[4] M. Farukh Munir, Arzad A. Kherani, and F. Filali, A distributed algorithm to achieve cesaro-wardrop equilibrium in wireless sensor networks. In IEEE CCNC Conference Proceedings, January 9-12 2008.

[5] http://www.tinyos.net/