QoS-Heterogeneity and Buffer Management in Constrained Intermittent-Connected Mobile Sensor Networks

H. Liu, A. Srinivasan, Y. Chen, J. Stankovic, K. Whitehouse
Department of Computer Science
University of Virginia
{hl4d|av3b|yc6z|stankovic|whitehouse}@cs.virginia.edu

Abstract

In intermittent-connected mobile sensor networks, generally nodes have a small number of neighbors and connections between the nodes rarely exist. This paper presents a new networking system for this type of network, aimed at maximizing resource utilization to achieve better system performance in terms of delay time and reliability. The system is motivated by the observation that a large number of sensor network applications, such as road monitoring sensor systems and animal tracking applications, have two kinds of co-existing data packets: One with soft real-time constraints while the other need to be delivered to the base station with high probability. Rather than focusing on energy consumption and network lifetime, other crucial issues are to design the communication system to satisfy the application requirements, and to manage the local buffer storage to avoid potential buffer overflow and data corruption or loss. Our proposed system achieves the above goals and makes significant improvements on system resource optimization. It is simulated in Matlab and evaluated under various practical scenarios. Simulation results demonstrate that our QoS-heterogeneous delay-tolerant routing approach performs better than ordinary epidemic routing and probabilistic forwarding strategies when packets with different types of QoS requirements exist in the network.

1 Introduction

Intermittent-connected sensor networks are composed of large numbers of minimal capacity sensing, computing, and communicating devices which do not have a permanent connection with the base station. Previous mechanisms and protocols developed for ordinary sensor networks, used for collecting and transporting data cannot be used in these scenarios because most of the solutions depend on a fixed connectivity to the base station and/or to the outside world. The severe memory and power constraints gets further aggravated in these situations where intermittent connectivity exists. Such networks are also called Delay Tolerant Networks (DTN) and this paper focuses on these types of networks.

Sensor network deployments in several applications [37, 29, 15] may be far removed from communications infrastructures such as the Internet. Yet, these networks of sensors must ultimately be connected to data storage and analysis facilities in order to increase the utilization of data collected by the system. The main challenge is connectivity to the Internet. Providing connectivity for such networks may involve exotic and unusual methods of data transfer. Delay Tolerant Networking architectures [5], which provide reliable data communication across heterogeneous, failure-prone networks, might be a potential solution. In recent years, large amounts of research have focused on delay tolerant sensor networks, ranging from energy-efficient routing protocol to in-network storage system design [23, 16, 27].

Normally, an epidemic-style routing strategy [36] is adopted in such networks to minimize the latency and improve network reliability. However, within such intermittent-connected sensor networks, problems of inefficient resource usage due to extremely constrained power and flash memory, stateless communication patterns, multi-type data packets with different QoS requirements, and unexpected buffer overflow are frequently encountered.

Our work is motivated by the observation that, a large number of sensor network applications, such as animal tracking applications, have two kinds of co-existing data packets: One with soft real-time constraints, i.e., the more quickly they are sent back, the more valuable they will be; while the other need to be delivered to the base station with high probability, but do not have latency constraints. For example, in a proposed project which deploys static sensor nodes along the Shenandoah Valley in Virginia and attaches sensor tags on fishes in the surrounding streams, environmental scientists are curious about the location of fishes that are recorded and stored in those static sensors functioning...
as local base stations. These types of packets, are not required to be delivered soon, instead are expected to have a data loss rate less than a threshold; on the other hand, when chemical pollution is detected and information is collected by nearby sensor tags, it must be delivered to the base station as soon as possible so we can analyze the situation and make early decisions. Another good example of networks that have a need for both fast and reliable packets is the road monitoring system in which the location of a vehicle is recorded and this type of message becomes less important after some time since this vehicle moves far away from the recorded position. Additionally, messages for the maintenance of this system are required to be delivered to the base station with high probability instead of quickly. These applications motivate us to investigate the QoS-heterogeneity issue in intermittent-connected sensor networks.

In this paper, we focus on studying QoS-heterogeneous epidemic routing in delay-tolerant sensor networks. In such environments, if we use a single replication-based epidemic protocol, which implies that we treat all packets the same, the subsequent system will not perform efficiently enough. In contrast, we propose an analytical framework to satisfy the specified requirements for data packets while using system resources efficiently. The main contributions of our work are as follows:

- To our best knowledge, this is the first paper explicitly dealing with the QoS-Heterogeneity in delay-tolerant sensor networks. We achieve this by applying different routing strategies to data packets with different quality of service requirements, optimizing system resource utilization, and designing a priority-based local buffer eviction policy. Concretely, we consider two types of packets in the network, one with soft real-time constraints and the other with reliability constraints. We use best-effort forwarding and probabilistic epidemic forwarding approaches on them respectively, and additionally calculate the optimal transmission probability to achieve better system performance. Simulation results demonstrate that our proposed system can satisfy the various QoS requirements when buffer is limited, i.e., packets with quickness requirements can reach the base station much faster and finally packets with reliability constraints can be delivered to the destination with extremely high probability.

The remainder of this paper is organized as follows. We compare our work with related work in Section 2. It is followed by the detailed system description including networking model in Section 3.1, the QoS-heterogeneous propagation model in Section 3.2, and a buffer management mechanism in Section 3.3. The simulation results for model validation are shown in Section 4. Finally we conclude the paper in Section 5.

2 State of the Art

After the pioneering work by Kumar et.al in [9] describing the disheartening results that per-node throughput decays as $n^{-1/2}$ for static wireless network, Grossglauser et.al showed that mobility could assist the capacity of wireless ad-hoc networks and long-term throughput can remain constant by guaranteeing that messages travel at most two hops, which implies either the data packet is delivered to the destination directly or is forwarded to at least one other node that reaches the destination [8]. Additionally, they proved that messages are guaranteed to be offloaded in maybe very long time, which is reasonable since intermittent-connectivity networks must inherently tolerate huge delays.

As a community that is interested in this research field, the Delay Tolerant Networking Research Group (DTNRG) mainly focus on the information processing in intermittent-connected networks without infrastructures. One important contribution made by DTNRG is the definition of expiration time for data packets as the time at which they are no longer meaningful, usually in networks where 100% reliability of packet delivery is unnecessary. In addition, exploiting communication strategies to combine delay tolerant networks and wireless sensor networks was first proposed by Ho and Fall in DTNRG in 2004 [12], which suggested a reasonable basis for a complete delay-tolerant sensor network architecture. Afterwards, more literature both inside and outside DTNRG, focused on various areas in delay-tolerant sensor networks, from power consumption [35, 28] to co-operative distributed storage issues [23].

“Stateless” routing in intermittent-connected sensor networks involve nodes that make forwarding decisions not dependent on node identity, mobility patterns, time of the day, etc. Obviously, this approach may lead to a “broadcasting storm” in the network and result in significant energy consumption and buffer usage. To alleviate the disadvantages, an “Epidemic” style of forwarding [36] makes sure that the same data packets would be not accepted as long as they are already offloaded. The primary idea in epidemic routing is that when two mobile separated nodes meet, an epidemic process is executed: the two nodes exchange messages they hold and finally carry the same set of messages. Epidemic routing is analogous to spreading of a disease. The source of the packet can be viewed as the first carrier of a new disease, the first infected node, and it copies the packet to (inflects) every node it meets. These new infected nodes act in the same way. As a result, the population of susceptible nodes (i.e., nodes without a copy of the packet) decreases over time. Once a node carrying a packet meets the base station, it passes the packet on to the base station. However, it can still introduce a lot of packets that occupy local buffer space and drain significant energy by infecting data packets to other nodes. Introducing a maximal hop count
can somewhat relieve this problem, when the hop count of a data packet has been reduced to 1, it can only be offloaded directly to the base station. Other improvements include the Probabilistic Forwarding [22] and Limited Time Forwarding [38].

To further extend the epidemic routing concept, Davis et al. in [4] exploited the in-network mobility patterns in intermittent-connected sensor networks. In their proposed approach, nodes can learn the existing mobility patterns of other sensor nodes when forwarding packets, and can erase packets by calculating the probability of delivery when local buffer space is full. In this paper, we only rely on the natural mobility of the nodes, which means we can neither control the node mobility nor store the mobility information in the local buffer.

The authors in [7] contain the development of a model based on an assumption that the meeting times between nodes are given by a Poisson process of some rate. It is shown that the Poisson assumption is valid for Random Way-point, Random Direction, and Random walk mobility models, which implies that Markov models of epidemic routing can lead to quite accurate performance predictions. Up to now the Markov models have been used to study the performance of epidemic routing [7, 11, 32], 2-hop forwarding [7], and spray-and-wait [33, 34] and fluid models have been derived from Markov models [38]. For example, [7] develops Markov chain models for epidemic routing and 2-hop forwarding, and derives the average source-to-destination delivery delay and the number of extant copies of a packet at the time of delivery; model predictions are validated through simulations [26].

Other hot research topics in delay tolerant sensor networks include routing for static sensor networks [30, 31, 13, 6]; modeling of epidemic forwarding under contention with limited buffer/bandwidth [14, 11]; buffer management in resource-constrained nodes; group communication (e.g. multicast) in delay tolerant sensor networks [3, 21]; and opportunistic network coding in delay tolerant sensor networks [39, 17, 20]. However, none of them have studied the QoS-heterogeneity issue, while this is a common characteristic in environmental monitoring applications [37, 29]. In this paper, we study the analytical representation for buffer management of intermittent-connectivity networks with various quality of service constraints. Using the analytical results from our proposed model, it is easy to precisely control the network parameters to make the system efficient as well as robust. Also, the model is designed general enough so that can handle other mobility patterns by recalculating the mobility coefficients.

In summary, a lot of achievements in the area of intermittent-connected sensor networks have been made recently, however, almost all of them occur at the delay-tolerant network level. In our proposed work we study the system performance from the sensor network applications point of view. In the next section, we describe the system model we used in detail.

3 QoS-Heterogeneous Delay-Tolerant Networking System

3.1 Problem Definition

As shown in Figure 1, the networking environment we consider has a set of $N + 1$ sparse mobile sensor nodes with a finite transmission range moving in a closed area. There are $N$ slave nodes and one mobile base node which is the final destination of all the data generated by the slave nodes. Their mobility pattern obeys the random way-point model. Specifically, they randomly pick an angle, randomly select a velocity in a predefined range $(v_{\text{min}}, v_{\text{max}})$, and run straight for a period of time, and then stop and pick another angle and velocity.

Our networking model can provide information sharing among sensor nodes in large areas through the process of replicating and diffusing of packets through the nodes. The mobile sensor nodes act as the physical carriers of information and the physical entity, which can be anything depending upon the application (say, for example horses in horse tracking application), act as the physical carriers of sensor nodes. All sensor nodes function cooperatively in order to deliver the data to the base station.

Packets are shared and exchanged in-network based on the Epidemiological Model, which is widely used to describe the transmission dynamics of a communicable disease. At first a man without this disease is called “susceptible” then becomes “infected” after he carries the disease and is infectious, finally he is “Healed” from the disease and becomes “immune”. The state transition of this man is shown in Figure 2.
Similarly, we say that two nodes “meet” when two sensor nodes come within transmission range of each other, and they then exchange their stored information and store a single copy of each packet at each node. The new infected nodes act in the same way. Since the transmission of the packet copy requires energy, one goal of our system design is such that the sensor nodes exchange only new information that is not been exchanged already. This reduces a lot of unnecessary network traffic. As a result, the amount of nodes without a copy of the packet decreases over time. Note that in our system design, when one node meets another one with the amount of packets larger than its remaining buffer space, it will only receive as many packets as possible instead. For example, when two sensor nodes meet one another and they are both 60 percents occupied, each of them will only be able to absorb 40 percent of packets from one another.

Later, when one of them reaches a collection station or a base station, the packets are offloaded into the station, deleted from the local buffer of this sensor node, which then retains the “packet-delivered” information (an “antibody”) to prevent it from receiving another copy of this packet in the future. In this “Antibody Broadcasting” way, useless packets in the network can be erased with high probability so as to save much more storage space for other usage. In addition, when this node meets other nodes, this copy is shared with the others, to release the stale data maintained by the nodes in the network.

Throughout our analysis and experiments, two different types of data as shown in the first section are collected periodically and divided into packets. Our system considers only two types of packets as of now: “Quickness” (Q) and “Reliability” (R). To distinguish the packets with quickness requirements from those with reliability requirement, we set a TTL (Time To Live) field for Q packets. This TTL is application-specific and set by the system designer, and means as soon as this TTL timer expires, the packet is obsolete regardless of whether the packet has successfully reached the base station or not. It will remain in the local buffer unless offloaded to the destination or the eviction policy has determined to erase it from the local buffer.

In this paper, we do not focus on the design of new hardware systems, nor the improvement of radio transmission and MAC layer schemes. Instead, we concentrate on how to make our system perform optimally for specified types of data packets. For the purpose of this work, any low-level architecture suffices as long as its flexibility and performance are good enough. Concretely, we study the performance of epidemic routing in such network environments where sensor storages are limited and QoS requirements are different packet by packet. As mentioned earlier, we notice that Groenevelt et al. showed in [7] that the inter meeting time between single pairs of nodes is asymptotically exponentially distributed if they have a relatively small transmission range and sufficiently high speed, as well as move according to the same mobility pattern. Although several sets of measurement evidence such as [2] indicated that the distribution of nodal meeting time has a heavy tail in some specified situations, much more related work believes the insight gained from Groenevelt’s analytical results, either explicitly assuming that the pairwise meeting time is exponential [20, 7, 26, 38] or making use of the Markov property implicitly [32, 33, 10, 11]. Therefore, we assume that the nodal meeting time is exponentially distributed throughout our paper.

To summarize formally, we consider a sparse mobile sensor network with N + 1 sensor nodes, with N slave nodes and one base station. All the nodes are assumed to have identical transmission range, r, relatively small to the network size. The mobility pattern we adopt is the popular Random Way-point model. In order to deal with the heterogeneity of packets, the slave nodes can generate two different types of packets depending on the nodes’ QoS constraints. The two types of QoS constraints we consider are: Q, which represents the packet set with latency requirements; and R, which stands for a set of packets with high reliability requirements. We assume that the two types of QoS constraints are mutually exclusive. The energy concerns are not taken into account in the current work. To simplify the analysis, we further assume the bandwidth is sufficient to deliver an arbitrary number of packets when node A and B meets each other or more than two nodes meet at the same time. We next describe the propagation model, especially how to deal with QoS-Heterogeneity and perform system-level optimization.

3.2 The Propagation Model

As mentioned before, our system generates two types of packets with different QoS requirements. The slave nodes can generate either a packet of type Q or a packet of type R depending upon its QoS requirement at that moment. We first describe the communication protocol details. Figure 3 describes the message exchange process when node A and B meet each other. As soon as two nodes come within the transmission range of each other, the nodes start exchang-
Packets with type \( Q \) needs to be delivered to the base station as soon as possible. Thus, for packets in \( Q \), we adopt a best-effort epidemic forwarding [38]. Whenever a node meets another node, they exchange \( Q \) packets in their own buffer as long as the receiver has enough buffer space. These type of packets are extremely common in many applications and can be very useful too. In case emergency conditions are detected by sensor nodes, these nodes need to inform the base station as soon as possible so people have time to make early decisions. In these situations, the sensor nodes can generate periodically packets of type \( Q \). In other cases, packets with type \( R \) are generated periodically. Note that the sampling rates for packets in \( R \) and \( Q \) are application-specific and probably different.

### 3.2.1 Protocol

As mentioned earlier, the quality of service requirements is not necessarily uniform among sensor nodes (\( Q \) packets for quickness and \( R \) packets for reliability), which calls for different data forwarding mechanisms to improve total system resource utilization. An ideal routing mechanism should make \( Q \) packets delivered to the base station as fast as possible and guarantee no loss for \( R \) packets in the meantime. A simple solution is to intuitively forward only \( Q \) packets in the network, while it is not efficient enough since \( R \) packets may suffer an intolerable long delay time to reach the destination or even get lost eventually because the sensor nodes run out of energy or space. Thus this solution is not practical.

From an efficiency performing perspective, it is therefore advantageous not to adopt a non-sharing way for \( R \) packets. In other words, a semi-epidemic style of forwarding is preferred. By controlling the data spreading as long as buffer overflow is not expected to occur, significant latency improvements can be achieved. For practical implementation reasons, we are interested in algorithms that use probabilistic epidemic routing for \( R \) packets and simple epidemic spreading for \( Q \) packets. Note that due to the randomness feature in practical applications, such algorithms should also accommodate certain potential buffer overflow, which we call an “Emergency”, and we deal with this buffer management issue in the next section.

Let us consider the following question we just posted: What is the optimal probability for spreading \( R \) packets, so that the expected buffer usage in sensor nodes is exactly the buffer capacity? Analytically, We can formulate it as a combined function with the following parameters: number of sensor nodes in the monitored area, the sampling rates for packets in \( R \) and \( Q \) respectively, local buffer capacity, and the pairwise meeting rate. It is apparently difficult to be calculated, however, we are able to construct analytical models for different types of packets and then calculate the optimal probability for those packets in \( R \), i.e., data packets with reliability requirement.

### 3.2.2 Analytical Model

We proceed to describe the analytical model. Our ultimate goal is to calculate the optimal transmission probability for the \( R \) packets, denoted by \( p \), with which the expected overall storage requirement is equal to the local buffer capacity in the steady state. To achieve this goal, consider the required parameters: number of sensor nodes, \( N \), in the monitored area \( A \), the sampling rates, denoted by \( \lambda_R \) and \( \lambda_Q \), for packets in \( R \) and \( Q \) respectively, local buffer capacity \( C_0 \), and the pairwise meeting rate \( \beta \).

As shown in [7], the pairwise meeting time between nodes is nearly exponentially distributed, if their transmission range \( r \) is relatively small compared to the region area \( A \), their speed is sufficiently high, and they move according to the common mobility patterns such as the random way-point model. Also the authors in [7] derived the following formula to estimate the nodal meeting interval (rate) \( \beta \):

\[
\beta = \frac{(2wrE[V^*])}{A},
\]

in which \( w \) is a mobility model specific constant, and \( E[V^*] \) stands for the average relative speed between the nodes. We note that since the superposition of independent Poisson processes is a Poisson process with rate equal to the sum of the rates, the meeting process between infected nodes start exchanging the information present in their respective buffers. The handshake message is mainly used to indicate the current available buffer space for each node and the other node uses this information to transmit messages to the first node without causing any buffer overflow in the receiving node. After the handshake process is completed, the nodes start exchanging the information present in their respective buffers.

![Figure 3. Communication Protocol](image)
nodes and the destination is a non-homogeneous Poisson process with rate $\beta I(t)$, in which $I(t)$ denotes the number of infected nodes at time $t$. The meeting process between infected nodes and susceptible nodes is similarly a non-homogeneous Poisson process with rate $\beta I(t)(N - I(t))$ [26].

First let us construct the analytical model for $Q$ packets. We denote $P(t)$ by the cumulative distribution function for this Poisson process, i.e., $P(t) = \text{Prob}[T_u < t]$. At the beginning of the infection, we assume $I(0) = 1$ and $P(0) = 0$. Since the infection rate is equal to the nodal meeting rate, we obtain the following first-order differential equation:

$$I'(t) = \beta I(N - I)$$

This equation is separable and thus can be solved with $I(0) = 1$ to give the solution as follows:

$$I(t) = N/(1 + (N - 1) \cdot e^{-\beta N t})$$

By integrating this particular function together with the initial condition $P(0) = 0$, we can find the cumulative distribution function:

$$P(t) = 1 - \frac{N}{1 + e^{-\beta p N t}}$$

The expected number of copies when it is offloaded to the base station, denoted by $E[C_T]$, can also be calculated using the approach described in [38]:

$$E[C_T] = \int_0^\infty I(t)P'(t) dt = (N - 1)/2$$

Based on the existing work on the cumulative distribution function and average number of copies, we extend the model to calculate the expected buffer usage in each local buffer in the steady state. The $P(T)$ curves and the numbers of infected nodes for each packet assist in the calculation of storage requirements for local buffers. We first predefine a threshold probability, denoted by $\alpha$, with which the packets are expected to be offloaded. For example, we choose probability of $\alpha = 0.99$, then find the appropriate value of $T$ from the plotted $P(T)$ curve. The TTL field is therefore set to this value of $T$, indicating that these packets become “obsolete” after $T$ time units with a confidence level of 0.99. In steady states of the system, local buffer usage is equal to the expected number of copies of generated packets multiplied by the expected number of packets generated during the delivery time. Thus, the expected storage requirement for $Q$ packets in the steady state, $E[S_Q]$, can be calculated as:

$$E[S_Q] = (E[C_T]) \cdot (\lambda Q P^{-1}(\alpha))(Packets) = \frac{\lambda Q}{2N^2} \left( N - 1 \cdot \ln((N/(1 - \alpha)) - (N - 1)) \right)$$

This value of $E[S_Q]$ represents the average buffer usage for storing $Q$ packets in the steady state; as a result, the remaining available storage space for $R$ packets, $S_{limit}$, should be $C_0 - E[S_Q]$. For packets with type $R$, reliability is the major criterion and they are not limited by deadlines. Since these packets only have a “soft” latency requirement, a pure epidemic flooding is unnecessary because it leads to an inefficient buffer resource usage. Therefore, instead of a pure epidemic flooding scheme, we use a probabilistic epidemic forwarding [22], that is, when two nodes meet one another, they would exchange their data with a probability $p$. Again, $p$ can be used to tradeoff delay and storage constraints depending on the application requirements and our concern is to obtain the optimal $p$ while guaranteeing the storage constraint. The goal in this case is to somehow successfully deliver the packet to the base station and this type of packets doesn’t have timing requirements.

Similar to the basic epidemic routing case, we are able to derive an ODE model to calculate the expected delay time and average buffer occupancy, which are summarized as follows:

The corresponding ODE equations for this scenario is:

$$I'(t) = \beta p I(N - I), P'(t) = \beta I(1 - P)$$

Note the difference here from the previous ODE where there was no probabilistic routing. The average number of infected nodes, $I(t)$, in this case is:

$$I(t) = N/(1 + (N - 1) \cdot e^{-\beta p N t})$$

and the cumulative distribution function is,

$$P(t) = 1 - ((N/(1 + e^{\beta p N t}))/p$$

The expected number of copies of generated packets is $E[C_T] = (N - 1)$ as calculated in [38]. Again, we extend this ODE model to calculate the average storage requirement in the steady state, $E(S)$:

$$E(S) = (N\lambda R P^{-1}(\alpha)) \cdot \frac{p}{1+p} (N - 1)$$

Thus the final formula for storage requirement is:

$$\frac{\lambda R}{\beta(1 + p)}(N - 1) \cdot \ln((N/(1 - \alpha)p) - (N - 1)) \leq S_{limit}$$

By solving the above inequality, we can obtain the optimal probability that minimize the latency for packets in $R$ while satisfying the storage constraints of the system. As long as we have the parameters $N, \lambda R, \lambda Q, \alpha, C_0, r, v_{max}, v_{min}$, and $A$, we can calculate the optimal probability. The above equations illustrate the heterogeneity of our system depending on the QoS requirements which is very common in many applications. This type of heterogeneity is novel and has not been implemented in earlier systems.
3.3 A Prioritized Eviction Policy

Thus far, we have described the whole communication-related system. Another novelty of our work is a prioritized buffer management policy. Most related work in intermittent-connected sensor networks have assumed that each node has sufficient space to accommodate all packets, however, mobile sensor nodes have limited buffer space in practice. It is difficult to manage local buffers to minimize data loss due to buffer overflow. From the sensor network point of view, several distributed storage services have been proposed and implemented on TinyOS, such as PRESTO [19] and TinyDB [24]. However, they are not designed for the intermittent-connected environment. Tailored for disconnected operation, our buffer management policy focuses on completely different concerns. All our design concepts make efforts on taking QoS-heterogeneity factors into account and thus can be seen as application-oriented. Concretely speaking, we propose a prioritized eviction policy to deal with the emergency case we mentioned before, which erases packets with the lowest priority when new packets are generated.

With the communication protocol design above, the network should run correctly in the steady state by sending packets with type $Q$ to the base station quickly and packets with type $R$ without any data loss. However, due to the random mobility pattern, it is necessary to handle the possible case of buffer overflow. This is because the sensor nodes exchange packets with other nodes when they meet. But since the movement of the nodes is random, there is a high chance that a node might meet a large number of different nodes within a limited time frame. The buffer of that node will be filled gradually, thereby leading to buffer overflow. This is a serious issue because we are dealing with sensor nodes, which have a very limited memory and the memory can be filled very quickly. This situation requires the usage of a proper eviction policy, which can reduce the burden on the buffers by evicting unnecessary packets. In other words, the eviction policy tries to evict those packets that it considers not so important. The eviction policy starts once the flash/buffer occupancy has reached the capacity limit.

Since our system includes packets with different QoS requirements, our eviction policy must suit the specific QoS requirement of the packet. Based on the QoS-heterogeneity, we propose a prioritized eviction policy. The basic idea of this eviction policy is to erase packets that have the least reliability requirement, which can be translated as follows:

- First, those packets in $Q$ that are already obsolete are deleted one by one, since searching through the buffer space may result in extra overhead and it can be avoided by this "FCFS" way.
- Active $Q$ packets generated by other sensor nodes are deleted, according to a "oldest-first" policy, since older packets are more likely shared with other nodes or even offloaded to the base station. Ties will be broken randomly (the same for other cases).
- Active $Q$ packets generated by a node itself are deleted, according to a "oldest-first" policy, since older packets are more likely shared with other nodes or even offloaded to the base station.
- Packets in $R$ generated by other sensor nodes are deleted randomly, since they have the same priority.
- Finally, packets in $R$ generated by a node itself are deleted according to a "oldest-first" policy, since older packets are more likely shared with other nodes or even offloaded to the base station.

3.4 Supporting QoS heterogeneity

After describing the whole system design in detail, now we proceed to explain how to combine the propagation model and the eviction policy to support different types of QoS such as $Q$ packets and $R$ packets in the network. As described above, to deliver the $Q$ packets to the destination as quickly as possible, we adopt a best-effort forwarding scheme to achieve minimum delay time; meanwhile, since they do not have reliability constraints, we set lower priorities on them. On the other hand, we use the probabilistic epidemic routing for packets in $R$ instead of non-sharing approach to avoid intolerably long delay time, but within the buffer capacity as expected in the analytical model. Also, we need to set higher priorities for them since they have stronger reliability requirements. In this way we provide a "tailored" optimal solution for each type of data packets existing in the network to meet their own QoS requirements. As an example, we can define $Q$ packets to have value $p = 1$ and priority 3 (active, generated by itself) or 4 (active, shared from others) or 5 (already obsolete), while $R$ packets have optimal value $p_0 = 0.32$ as calculated and priority 1 (generated by itself) or 2 (shared from others). Equipped with these system parameters, designers can easily implement the corresponding system and benefit from QoS heterogeneity instead of suffering decreased performance when treating all packets the same.

4 Evaluation

In this section, we use simulations to evaluate the performance of our proposed QoS-heterogeneous epidemic routing framework together with a prioritized eviction policy. The simulator we have developed implements our proposed algorithms and protocols and provides a simulation environment of intermittent-connected mobile sensor network applications.
The simulation environment is a 500m by 500m rectangle region with no obstacles. In this region, 20 nodes and one base station move obeying the random way-point model. Specifically, they will randomly pick an angle and a velocity from a minimum value $2m/timeunit$ to a maximum value $6m/timeunit$ and run straight for 5 time units, and then stop and pick another angle and velocity. The sampling rate for packets in $R$ and $Q$ is one per 4 time units. The TTL value for packets in $Q$ is set to 500 time units. The transmission range of nodes and the base station is assumed to be 12.5m, which is relatively small than the network size. The storage constraint for sensor nodes is 500 packets. In later experiment scenarios, we modify the number of nodes and buffer capacity to check the effect of these parameters on the system performance, respectively. For convenience, we list the value of important experiment parameters in Table 1 in the Appendix.

### 4.1 Basic System Performance

We first investigate the basic operation of our system. The goal of this experiment is mainly to study the data delivery for packets with different quality of service requirements. In addition, we study the impact of the number of sensor nodes on the delivery latency and packet reception ratio for generated packets in $R$ and $Q$. To achieve this goal, we generate 10 $Q$ packets and 10 $R$ packets for each node, denoted by $K_Q = 10$ and $K_R = 10$, that is, there are totally 200 $Q$ packets and 200 $R$ packets in the network. We use the cumulative packet reception ratio (PRR) for the base station as the evaluation metric under the different parameter settings ($n = 20$ and 10, capacity = 1500 and 500), and run the simulator for a long enough time (200,000 time units). To clearly see the effect of transmission probability on the cumulative distribution function, we set the transmission probability of $R$ packets to be 0.5. Note that here we are not calculating the optimal transmission probability for packets in $R$, but setting it to be a predefined value. In later experiments where buffer capacity is limited, we use the storage usage formula in the analytical model to calculate the optimal $p_0$.

To illustrate the in-network data aggregation, Figure 4 compares the cumulative packet reception ratio under different experimental settings, by varying the network density and buffer capacity of the system. The figure clearly shows that the packet reception ratios of $Q$ packets approach 100 percent very quickly when compared to the $R$ packets, that is, the $Q$ packets are offloaded to the base station more quickly than $R$ packets. For example, when we check the PRR result after running for 50,000 time units, we observe that the corresponding value for the experiment with 10 sensor nodes is 60 percent for $Q$ packets and 30 percent for $R$ packets. Similarly, the PRR for $Q$ packets is much higher than that for $R$ packets for the cases with 20 nodes.

We also notice from Figure 4 that as the network density increases, data delivery happens much more quickly for $Q$ packets. On the other hand, for $R$ packets it remains almost the same at first, then after a long time, the base station collected more packets from the case in which more sensor nodes are moving. This is mainly because for $Q$ packets we use a best-effort forwarding strategy and thus additional nodes may assist delivering data packets to the destination much more quickly; while since we use probabilistic epidemic routing for $R$ packets, the overall effect of higher network density is not that obvious until after running for a long enough time.

Finally, we note that for all cumulative curves there exists a “heavy tail”, which implies that several data packets in the network still need much more time to reach the base station. This further proves the necessity and benefit of system optimization we described before. The non-sharing routing approaches may result in intolerable long delay time and meanwhile significantly many storage spaces are wasted in nodal local buffers.

### 4.2 Buffer Management Evaluation

The previous subsection evaluates the basic functionality of the system which illustrates the QOS heterogeneity of
our system. But an important issue is to utilize the available buffer space efficiently along with QOS heterogeneity. This section evaluates the effect of various buffer capacities on the reliability performance of our system. The buffer utilization is evaluated by calculating the packet reception ratios for $Q$ packets and $R$ packets for varying buffer capacities. By utilizing the optimal probability calculation formula, we obtain that $p_0$ for buffer capacities equal to 50, 100, 200, 500, 1500 is 0.233, 0.776, 1, 1, 1, respectively.

Figure 5 plots the PRR for $Q$ packets and $R$ packets for different buffer capacities. As shown in the figure, the PRR for $R$ packets stays at 100 percent when the buffer capacity varies from 50 to 1500 and the PRR for $Q$ packets gradually increases as the buffer size increases. This is natural because when the system has larger buffer capacity the number of packets lost will be less and the $R$ packets are well protected due to our proposed prioritized eviction policy. The large PRR gap between the buffer capacity 50 and 100 indicates that constrained storage may lead to more and more dropping of $Q$ packets. A small dip in the graph when the capacity is 500 can be attributed to the collisions that might have occurred when three or more nodes are transmitting information between them.

Intuitively, our proposed prioritized eviction policy satisfies the quality of service requirements better compared with other two commonly used eviction policies: Oldest-first deletion and random erasure. Oldest-first deletion searches for the packet that is generated earlier than others and erases it from the buffer; and random erasure picks up one packet randomly and does the same operation. Both of these methods erase packets with reliability constraints from the local buffer when storage capacity is limited, even when there are better choices to keep in the buffer. On the contrary, once priorities are predefined in our method, the $R$ packets are held to the last minute.

4.3 Representing the Whole System

We proceed to study the performance of the whole routing system. The goal of this experiment is to investigate the effect of our proposed routing system on various types of network traffic. Ideally, our routing system performs the best when different types of packets co-exist. When all packets in the network are $Q$ type, network flooding leads to the continuous exchange of messages between nodes, gradually filling the buffer and ultimately resulting in the dropping of many packets. When all are $R$ packets, although we use the probabilistic epidemic routing approach, as time goes on and new packets are generated, old packets are discarded because of limited buffer capacity and low probability to meet the destination, resulting in decreasing overall packet reception ratio in the network.

To illustrate this, we did three groups of experiments under the limited buffer condition: The first group contains the same amount of $Q$ packets and $R$ packets in the network; the second group has only $Q$ packets and third one has only $R$ packets. The number of sensor nodes stays at 20. The buffer capacity is set to 50 packets and we generate $K_Q = 20$ and $K_R = 20$ in the first case, $K_Q = 40$ in the second case, and $K_R = 40$ in the third case. The QOS-heterogeneous epidemic routing as well as prioritized eviction policy are used in the first experiment. Ordinary epidemic routing and oldest-first eviction policy are used in the second one; and probabilistic routing together with random evicition are adopted in the third one. We first calculate the optimal probability $p_0$ for packets in $R$ in the first experiment and the result is 0.106, then we generate a random number from $(0, 1)$ to be the transmission probability for the third experiment, that is 0.439. We then run the simulation until there are no packets remaining in the buffer of sensor nodes and show both the cumulative distribution functions in a short period in the beginning and the final PRR values in Figure 6.

We first observe that data packets are delivered to the destination faster in the first experiment. It is a little bit surprising because the second experiment uses the ordinary epidemic routing and the transmission probability in the third condition is four times higher than that for $R$ packets in the first case. The main reason lies in the constrained buffer capacity. Note that the limit is 50 and the total amount of packets generated by itself is already 40. Thus in the second and third cases, although they share more aggressively, it only results in more and more packets dropped and erased. In addition, the eviction policies in these two cases are inefficient since they do not care whether one sin-
ingle packet in the buffer is generated by itself or shared from other nodes. All these reasons lead to the excellent results of our QOS-heterogeneous DTN routing approach.

We also note that the final PRR value for \( Q \) packets in the first experiment is 77 percent and this is because some nodes received several \( R \) packets from others and later when the eviction policy took place, its own generated \( Q \) packets are partially erased since they have higher priorities than those \( R \) packets from others. This is reasonable since those shared \( R \) packets should be protected in case the original node has crashed and original copies are lost. Also from the graph we notice that all \( Q \) packets are delivered back to the base station in the first \( 15,000 \) time units in the second experiment and the final packet reception ratio for \( R \) packets in the first experiment is 100 percent. These promising results show that our proposed routing strategy performs particularly well when there are different QoS kinds of packets existing in the network.

We may observe from Figure 6 that \( Q \) packets do not reach the base station more quickly than \( R \) packets as expected. This is mainly because the traffic pattern we used cannot show this. All packets are generated within the first \( 100 \) time units so we cannot see clearly what happens at the beginning for a long time. To further investigate the system performance, especially to check whether \( Q \) packets can be offloaded to the destination faster than \( R \) packets in a long enough period, when the system keeps on generating packets continuously, we adjust the settings to the following: \( K_Q \) and \( K_R \) are set to 1000 and the sampling rate for \( Q \) and \( R \) packets are both per 5 time units. This means each node keeps generating packets for 5000 time units. The \( Q \) packets become obsolete after 100 time units. The buffer capacity is set to 500 packets which is so small that the \( R \) packets will not share with others. The evaluation metric is the number of packets received by the base station.

Figure 7 plots the number of \( Q \) packets and \( R \) packets received by the base station for the first 5000 time units. We can see that the \( Q \) packets do reach the base station much faster than \( R \) packets especially during the first 300 units. Also, we notice that the gap becomes smaller as time goes on since more and more packets in \( R \) reach the destination while at the mean time many packets in \( Q \) are erased from the buffer due to the prioritized eviction policy. These results together with the above experiment demonstrate that our proposed system can satisfy the various QoS requirements when the buffer is limited. Packets with quickness requirements can reach the base station much faster and finally packets with reliability constraints can be delivered to the destination with extremely high probability.

5 Conclusion

In this paper, we introduce and study the problem of dealing with QoS-Heterogeneity in intermittent-connected mobile sensor networks. Our proposed routing model captures the dynamics of existing epidemic routing protocols, and makes use of them to achieve better system performance in a sparse sensor network where two types of packets co-exist: one with quickness requirement and the
other with reliability. We adopted a best-effort epidemic routing for quickness-constrained packets and probabilistic forwarding for reliability-constraint ones. In addition, we further propose a prioritized eviction policy to handle unexpected buffer overflow. Simulation results demonstrate that our QoS-heterogeneous delay-tolerant routing approach performs better than ordinary epidemic routing and probabilistic forwarding strategies when packets with different types of QoS requirements exist in the network.

As the first attempt to solve the QoS-heterogeneity problem, our work inevitably has several limitations. We now assume that the QoS types are mutually exclusive, while practical applications probably require packets that both have several QoS requirements at the same time. Also our assumption of random way-point mobility model may be invalid in animal/people tracking applications since they may have a more precise biological-based mobility pattern. Finally, energy concerns may somehow affect the system performance due to the Radio Irregularity Model. These limitations are the gap between our theoretical model and implementations in practical applications. However, we claim that our work is valuable since our theoretical routing model provides a precise prediction of the simplified situation and practical issues may be solved by other variations of our model since it is general enough to be extended. Also simulation results showed that our work could indeed assist in designing an appropriate routing system for various network conditions.

For our future work, we will first extend our basic analytical model to accommodate more complex cases, such as packets with both latency and reliability constraints. Furthermore, we would like to investigate the case under more realistic mobility model, i.e., to study the non-uniform mobility patterns, in for example animal tracking applications.

References


### Appendix

#### Table 1. Notations Used in the Paper

<table>
<thead>
<tr>
<th>Notation</th>
<th>Description of Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>$N$</td>
<td>number of nodes in the network</td>
</tr>
<tr>
<td>$A$</td>
<td>area of the network region</td>
</tr>
<tr>
<td>$r$</td>
<td>transmission range</td>
</tr>
<tr>
<td>$Q$</td>
<td>packets with quickness requirement</td>
</tr>
<tr>
<td>$R$</td>
<td>packets with reliability requirement</td>
</tr>
<tr>
<td>$\beta$</td>
<td>nodal meeting rate</td>
</tr>
<tr>
<td>$p$</td>
<td>transmission probability for packets in $R$</td>
</tr>
<tr>
<td>$C_0$</td>
<td>buffer capacity for sensor nodes</td>
</tr>
<tr>
<td>$\lambda_R$</td>
<td>number of $R$ packet generated per time unit</td>
</tr>
<tr>
<td>$\lambda_Q$</td>
<td>number of $Q$ packet generated per time unit</td>
</tr>
<tr>
<td>$I(t)$</td>
<td>number of infected nodes at time $t$</td>
</tr>
<tr>
<td>$P(t)$</td>
<td>cumulative distribution function at time $t$</td>
</tr>
<tr>
<td>$E[C_F]$</td>
<td>expected number of copies when offloading</td>
</tr>
<tr>
<td>$E[S_Q]$</td>
<td>expected buffer usage for $Q$ packets</td>
</tr>
<tr>
<td>$S_{limit}$</td>
<td>expected maximum buffer usage for $R$ packets</td>
</tr>
<tr>
<td>$\alpha$</td>
<td>predefined threshold for expected offloading</td>
</tr>
<tr>
<td>$TTL$</td>
<td>packets become obsolete after $TTL$ time steps</td>
</tr>
</tbody>
</table>