A Novel Buffer Management Architecture for Epidemic Routing in Delay Tolerant Networks (DTNs)

Ahmed Elwhishi[1], Pin-Han Ho[1], K. Naik[1], and Basem Shihada[2]

[1] Dept. of Electrical and Computer Eng., University of Waterloo, Canada
`aelwhish@engmail.uwateloo.ca`, `{pinhan,snaik}@uwaterloo.ca`
[2] Department of Computer Science, KAUST University, Thuwal, Saudi Arabia
`basem.shihada@kaust.edu.sa`

Abstract. Delay tolerant networks (DTNs) are wireless networks in which an end-to-end path for a given node pair can never exist for an extended period. It has been reported as a viable approach in launching multiple message replicas in order to increase message delivery ratio and reduce message delivery delay. This advantage, nonetheless, is at the expense of taking more buffer space at each node. The combination of custody and replication entails high buffer and bandwidth overhead. This paper investigates a new buffer management architecture for epidemic routing in DTNs, which helps each node to make a decision on which message should be forwarded or dropped. The proposed buffer management architecture is characterized by a suite of novel functional modules, including Summary Vector Exchange Module (SVEM), Networks State Estimation Module (NSEM), and Utility Calculation Module (UCM). Extensive simulation results show that the proposed buffer management architecture can achieve superb performance against its counterparts in terms of delivery ratio and delivery delay.

Keywords: Routing, Buffer management.

1 Introduction

One of most important characteristics of a DTN is the lack of an end-to-end path for a given node pair for extended periods [1]. To cope with frequent and long-lived disconnections due to node mobility, a node in a DTN is allowed to buffer a message and wait until it finds an available link to the next hop. The next hop node buffers and forwards the received message accordingly if it is not the destination of the message. This process continues until the message reaches its destination. This model of routing constitutes a significant difference from conventional ad hoc routing, and is usually referred to as encounter-based routing, store-carry-forward routing, or mobility-assisted routing. The names come from the fact that the routing of a message in DTNs has taken the nodal mobility as a critical factor in the decision on whether to forward the message.

To achieve better robustness, shorter delivery delay, and higher delivery ratio, extensive research efforts have been reported in design of efficient multi-copy routing algorithms [6,3,7]. However, many DTN routing protocols have assumed negligible storage overhead [2,4]. They have not considered the fact that each node could be a hand-held and battery-powered device with stringent power consumption and buffer size limitation. The buffer limitation may cause message drop/loss due to buffer overflow, which leads to a big challenge in the implementation of most previously reported schemes such as those belonging to the class of epidemic (flooding) routing. With Epidemic routing, two nodes simply exchange all messages that are not in common when they encounter. Without an appropriate countermeasure, the message copies could be spread throughout the network like an epidemic and overwhelm the network resources in terms of buffer spaces and bandwidth.

The paper studies a novel buffer management architecture for DTNs under epidemic routing, aiming to enable an effective decision process on which messages should be dropped in the case of buffer overflow. In specific, the proposed buffer management architecture is based on a fluid flow limit model of Markov chain that can simply approximate the solution via ordinary differential equations (ODEs). Note that an extremely high computation complexity is required in directly solving a Markov chain model under epidemic routing even in presence of a small number of nodes [9,8]. The use of ODEs, although serves as an approximation of the Markov chain result, can nonetheless solidly improve the computation efficiency and provide a closed-form expression. On the other hand, the formulation with the proposed fluid flow limit model is very scalable to the network size, where the complexity does not increase when the number of network nodes increases.

The ODE solution gives per-message utility values, which are calculated based on the estimation of two global parameters: the number of message copies, and the number of nodes who have "seen" this message, i.e., the nodes that have either carried the message or rejected the acceptance of this message. The per-message utility values at each node are then used for the decision on whether a buffered message should be dropped in any contact. We will demonstrate a closed-form solution to the proposed ODE approach, such that each the per-message utility can be calculated efficiently. Simulation results confirm the efficiency and effectiveness of the proposed buffer management scheme under the epidemic routing.

The rest of this paper is organized as follows. Section 2 describes the related work in terms of buffer management and scheduling in DTNs. Section 3 provides the background and system description, including a brief of fluid flow model and network model adopted in this study. Section 4 introduces the proposed buffer management scheme under epidemic routing, including a number of key functional modules: Summary Vector Exchange Module (SVEM), Prediction Module (PM), and Utility Calculation Module (UCM). Section 5 provides experiment results which verify the proposed buffer management architecture. Section 6 concludes the paper.

2   Related Work

Although routing issues in DTNs have been extensively researched in resent years, only a few studies have examined the impact of buffer management and scheduling policies on the performance of the routing techniques. Zhang et al. in [9] addressed this issue in the case of epidemic routing by evaluating simple drop policies such as drop-front and drop-tail, and presented an analysis of the situation that occurs when the buffer at a node has a capacity limit. The paper concluded that the drop-front policy outperforms the drop-tail when higher priority is given to the source node. Lindgren et al. in [10] evaluated a set of heuristic buffer management policies and applied them to a number of routing protocols. Since these policies are defined according to locally available network status, the performance could be improved by considering network-wide node status along with a scheduling scheme for the buffered messages at each node. Khrifa et al. in [11] proposed an interesting approach for solving the problem of buffer management by way of a drop policy and a scheduling scheme. This was the first study that explicitly took global knowledge of node mobility as a constraint in the task of buffer management. Specifically, their method estimates the number of copies of message i based on the number of buffered messages that were created before message i. Although interesting, the method may become inaccurate when the number of network nodes is getting larger, especially for newly generated messages. Meanwhile, the effect due to the change of the number of message copies during the remaining lifetime of a message is not considered in the utility function calculation, which means the utility function is only affected by the current message copies and its remaining lifetime. Obviously, the above mentioned studies leave a large room to improve, where a solution for DTN buffer management that can well estimate and manipulate the global status is absent.

3   Background and System Description

The section presents the background of our mathematical model as well as the network model for encounter-based epidemic routing.

3.1   Background on Fluid Flow Model

In a nutshell, the paper has the buffer management task in DTN epidemic routing to be formulated as a fluid-flow Markov-chain process, which is solved by a novel ODE based approach [13,9]. The fluid flow model can then be used to formulate the rate of message propagation among nodes, calculating the expected time until a given node (destination) is infected, and then calculating the delivery ratio (delivery probability). The following notation are used throughout the paper.

- $n_i(t)$ denotes the number of nodes with message i in their buffers (also referred to as "infected" at time t), where t is counted from the creation time of message i. The following relation is used to calculate $n_i(t)$:

$$n_i^{'}(t) = \beta n_i(t)(N - n_i(t))) \tag{1}$$

where N is the number of nodes in the network, and β is the meeting time rate between nodes. Solving equation (1) with the initial condition $n_i(0)$ yields

$$n_i(t) = \frac{Nn_i(0)}{n_i(0) + (N - n_i(0))e^{-\beta Nt}} \qquad (2)$$

– $P_i(t) = P_i(T_d < t)$ denotes the cumulative probability (CDF) of message$i$ being delivered at time t, where T_d denotes a random variable for the time instant that the message i is successfully delivered. $P_i(t)$ can be expressed in a differential equation form[9]:

$$P_i^{'}(t) = \beta n_i(t)(1 - P_i(t)) \qquad (3)$$

solving the equation (3) with the initial condition $P_i(0) = 0$ yields

$$P_i(T_d < t) = 1 - \frac{N}{N - n_i(0) + n_i(0).e^{\beta Nt}} \qquad (4)$$

(1) and (4) are valid only for unlimited buffer space. To extend the above relations to the scenario with limited buffer space, an additional factor should be considered (denoted as P_{fi}), which represents the probability that the nodal buffer space is available and the message can be transferred to an encountered node. Note that P_{fi} can be obtained by historical data of nodal encounters. Thus accordingly, (1) and (4) are reformulated as follows:

$$n_i(t) = \frac{N}{n_i(0) + (N - n_i(0))e^{-P_{f_i}\beta Nt}} \qquad (5)$$

$$P_i(T_d < t) = 1 - \left(\frac{N}{N - n_i(0) + n_i(0)e^{\beta P_{fi}Nt}} \right)^{\frac{n_i(0)}{P_{fi}}} \qquad (6)$$

3.2   Network Model

In this paper, a homogeneous DTN is modeled as a set of N nodes, all moving according to a specific mobility model in a finite area, where inter-encounter time between each pair of nodes follows *iid*. Let the number of total messages in the network be denoted as $K(t)$, and the buffer capacity of each node be denoted as B. messages. The messages are generated arbitrarily between source and destination nodes. Each message is destined to one of the nodes in the network with a time-to-live (denoted as Tx). A message is dropped if its Tx expires.

For any given node, a, it is assumed that $J_a(t)$ messages are stored in its buffer at time t. Each message $i(t)$, $i \in [1, J_a(t)]$ is denoted by a tuple $< Sr(t), Dst(t), T_i, R_i, n_i(t), m_i(t), P_{f_i} >$, which represents the source, destination, elapsed time since the creation of the message, remaining lifetime of the message ($R_i = Tx_i - T_i$), number of copies of the message, and the number of nodes who have

"seen" message i, respectively. Obviously we have $m_i(t) = n_i(t) + s_i(t)$, where $n_i(t)$ is the number of nodes who are carrying a copy of message i, and $s_i(t)$ is the number of nodes who have "seen" but never accepted the message. Thus, P_{f_i} can be calculated as follows: $P_{fi} = \frac{n_i(t)}{m_i(t)}$.

Let the expected inter-encounter time of any two nodes a and b be denoted as EMT_{ab}, which is defined as the time period taken by the two nodes to enter into their transmission again. The encounter (or mixing) rate between a and b, denoted as β_{ab}, is the inverse of the expected inter-encounter time for the two nodes: $\beta_{ab} = \frac{1}{EMT_{ab}}$. We assume that EMT_{ab}, a, $b \in [1, N]$ follows an exponential distribution (or referred to as with an exponential tail [14]). It has been shown that a number of popular mobility models have such exponential tails (e.g., Random Walk, Random Waypoint, Random Direction, Community-based Mobility [5,16]). In practice, recent studies based on traces collected from real-life mobility examples [17] argued that the inter-encounter period and the encounter durations of these traces demonstrate exponential tails after a specific cutoff point. Based on the *iid* of the mobility model of the nodes, the distribution of the inter-meeting time can be predicted. The historical inter-encounter information between nodes a and b can be calculated by averaging cumulatively all inter-encouter times until current time t. Parameter β is calculated as follows:

$$\beta = \frac{1}{E(MT_{a,b})} \approx \frac{1}{\frac{1}{n} \sum MT_{a,b}} \qquad (7)$$

The historical information becomes more accurate and the adaptation of the mobility characteristics becomes precise with a greater elapse of time.

4   Proposed Buffer Management Architecture

Fig. 1 provides a whole picture on the proposed DTN buffer management architecture, which illustrates the functional modules and their relations. The summary vector exchange module (SVEM) is implemented at a node during a contact; then the prediction module (PM) is applied to estimate the values of $m_i(T_i)$ and $n_i(T_i)$ according to the most updated network information. The two parameters are further taken as inputs in the calculation of the proposed per-message utility function in the utility calculation module (UCM). The decision of forwarding or dropping the buffered messages is made based on the buffer occupancy status and the utility value of the messages. The rest of the section introduces the details of each module.

4.1   Summary Vector Exchange Module (SVEM)

During each contact, the network information summarized as a "summary vector" is exchanged between the two nodes, which includes the following data: (1) statistics of inter-encounter time of every node pair maintained by the nodes, (2) statistics regarding the buffered messages, including their IDs, remaining time to live (R), destinations, the stored $n_i(T_i)$ and $m_i(T_i)$ values for each message
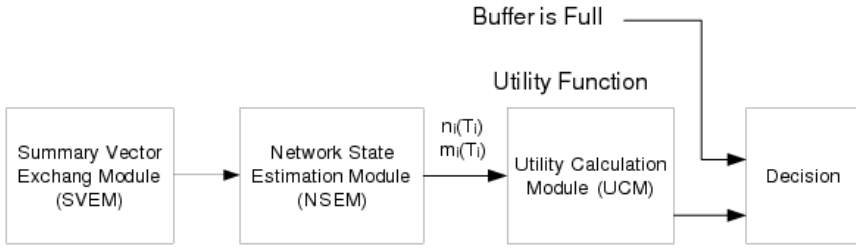
Fig. 1. The buffer management architecture

that were estimated in the previous contact. The SVEM ensures the above information exchange process, and activates PM for the parameter estimation based on the newly obtained network statistics right after each contact.

4.2   Prediction of Message Dissemination

The PM is used to obtain the estimated $m_i(T_i)$ and $n_i(T_i)$ such that the UCM can make decision on the buffer management. For this purpose, we propose a novel estimation approach called Global History-Based Prediction (GHP), which estimates the parameters by considering their statistics since the corresponding message was created.

Let $M_i(T_i)$ and $N_i(T_i)$ denote two random variables that fully describe the two parameters $m_i(T_i)$ and $n_i(T_i)$ at elapsed time T_i, respectively. We have: $E[M_i(T_i)] = \frac{\sum_{i=1}^{j} m_i(T_i)}{j}$ and $E[N_i] = \frac{\sum_{i=1}^{j} n_i(T_i)}{j}$, where j is the total number of messages currently in the buffer of node a and b which are more senior than message i. In the same manner, the average elapsed times for all messages that were generated before message i is calculated as $\widehat{T} = \frac{\sum_{i=1}^{j} T_i}{j}$. Thus, we can have the following estimations for message i: $\widehat{m_i(T_i)}$ and $\widehat{n_i(T_i)}$. These values are then incorporated into the per-message utility metrics, which are calculated as $\widehat{m_i(T_i)} = \frac{T_i E[M_i]}{\widehat{T}}$ and $\widehat{n_i(T_i)} = \frac{T_i E[N_i]}{\widehat{T}}$.

4.3   Utility Calculation Module (UCM)

Based on the problem settings and estimated parameters, the UCM answers the following question at a node during each nodal contact: Given $n_i(T_i)$ and $m_i(T_i)$ and insufficient buffer space for supporting epidemic routing [2], what is an appropriate decision on whether the node should drop any message in its buffer or reject any incoming message from the other node during the contact, such that the average delivery ratio or delivery delay can be optimized? We will describe how this can be achieved in the rest of this section.

Maximization of Delivery Ratio. Let us assume that the buffer is full at node b and there is a message i with elapsed time T_i in a network that has K messages at the moment at which the decision should be made by a node

with respect to dropping a message from all messages in its buffer. Further, let $m_i(T_i)$ and $n_i(T_i)$ denote the number of nodes that have "seen" message i since its creation (excluding the source) and those who have a copy of it at this moment, respectively. It is clear that $n_i(T_i) \leq m_i(T_i) + 1$.

Theorem 1. *The best way to maximize the average delivery rate is to drop message i_{min} that satisfies the following:*

$$i_{min} = argmin_i \left[\left[e^{\beta N R_i P_{fi}} \left(\beta R_i n_i(T_i) + \frac{m_i(T_i)}{N} \right) - \frac{m_i(T_i)}{N} \right] . \right.$$

$$\left. \left(1 - \frac{m_i(T_i)}{N-1} \right)^2 \left(\frac{N}{N - n_i(T_i) + n_i(T_i).e^{\beta N R_i P_f}} \right)^{m_i(T_i)+1} \right] \tag{8}$$

Proof. The probability that a copy of message i will not be delivered by a node is given by the probability that the next meeting time with the destination is greater than its remaining lifetime R_i , assuming that the message i has not yet been delivered. The probability that message i will not be delivered (i.e., none of its copies will be delivered) can be expressed as

$Pr\{message\ i\ not\ delivered\ |\ not\ delivered\ yet\} =$

$$Pr(T_d > T_i + R_i \mid_{T_d > T_i}) = \left(1 - \frac{m_i(T_i)}{N-1} \right)$$

$$. \left(\frac{N}{N - n_i(T_i) + n_i(T_i)e^{\beta P_{fi} N R i}} \right)^{\frac{n_i(T_i)}{P_{fi}}} \tag{9}$$

The proof of (9) is provided in the Appendix.

By assuming network homogeneity, there is an equal likelihood that the message is "seen" by each node. Thus, the probability that message i has been already delivered to the destination is equal to

$$Pr\{ message\ i\ already\ delivered \} = \frac{m_i(T_i)}{(N-1)} \tag{10}$$

By combining (9) and (10), the probability that message i is successfully delivered before its Tx expires can be calculated as follows:
$Pr_i = 1 - P\{message\ i\ not\ yet\ delivered\}.$

$P\{message\ i\ will\ not\ be\ delivered\ within\ R_i\}$

$$Pr_i = 1 - \left(1 - \frac{m_i(T_i)}{N-1} \right)^2 \left(\frac{N}{N - n(T_i) + n(T_i).e^{\beta P_f N R i}} \right)^{\frac{n_i(T_i)}{P_f}} \tag{11}$$

When a node is operating at its maximum buffer capacity, it should drop one or multiple messages so as to achieve the best gain in the increase of the global delivery ratio $Pr = \frac{1}{K(t)} \sum_{i=1}^{K(t)} Pr_i$. To make the optimal decision locally at the node, Pr_i is differentiated with respect to $n_i(T_i)$, and $\partial n_i(T_i)$ is then discretized and replaced by $\triangle n_i(T_i)$.

The best drop policy is one that maximizes $\triangle Pr_i$:

$$\triangle Pr_i = \frac{1}{K(t)} \sum_{i=1}^{K(t)} \frac{\partial Pr_i}{\partial n_i(T_i)} * \triangle n_i(T_i)$$
$$= \frac{1}{K(t)} \sum_{i=1}^{K(t)} \left[\left[e^{\beta N R_i P_f} \left(\beta R_i n_i(T_i) + \frac{m_i(T_i)}{N} \right) - \frac{m(T_i)}{N} \right] \cdot \right.$$
$$\left. \left(1 - \frac{m(T_i)}{N-1} \right)^2 \left(\frac{N}{N - n_i(T_i) + n_i(T_i) e^{\beta N R_i P_{fi}}} \right)^{m_i(T_i)+1} \right] \triangle n_i(T_i)$$

Thus, the maximum delivery ratio can be achieved if the message that causes the least decrease in $\triangle Pr$ is discarded. On the other hand, when message i is discarded, the number of copies of message i in the network decreases by 1, which results in $\triangle n_i(T_i) = -1$. Thus the optimal buffer dropping policy that can maximize the delivery ratio based on the locally available information at the node is to discard the message with the smallest value of $\frac{\partial Pr_i}{\partial n_i(T_i)}$, which is equivalently to choose a message with a value for i_{min} that satisfies (8). This derivation is an attempt to handle changes in the number of copies of a message that may be increased in the future during new encounters. This goal can be achieved by predicting P_f, the probability of forwarding a copy of message i to any node encountered, which is incorporated into the estimation of the delivery ratio. It is clear that the accuracy of P_f is based mainly on the precision in estimating the values of $m_i(T_i)$ and $n_i(T_i)$.

Minimization of Average Delivery Delay. To minimize the average delivery delay, node b should discard the message such that the expected delivery delay of all messages can be reduced the most.

Theorem 2. *To achieve the minimum average delivery delay, node a should drop the message that satisfies the following:*

$$i_{min} = \left(\frac{N}{N - 1 + e^{\beta P_{fi} NTi}} \right)^{\frac{1}{P_{fi}}} *$$

$$\left[\frac{1}{(n_i(T_i))^2 \beta} \left(\frac{n_i(Tx)}{N - 1} \right) \right] \tag{12}$$

Proof. The expected delay in delivering a message that still has copies existing in the network can be expressed

$$D_i = P\{message\, i\, not\, deliverd\, yet\} * \frac{1}{P_{fi}} E[T_d \mid T_d > T_i]$$

$$D_i = \left(\frac{N}{N - 1 + e^{\beta P_{fi} NTi}} \right)^{\frac{1}{P_{fi}}} * \frac{1}{P_{fi}} E[T_d \mid T_d > T_i] \tag{13}$$

Since we have a homogeneous network, the expected delay of a message can be calculated as

$$E[T_d \mid T_d > T_i] = \left[T_i + \frac{1}{P_{fi}} \int_0^{Tx} t f(t) dt \right]$$
$$= \left[T_i + \frac{1}{P_{fi}} \int_0^{Tx} t \beta e^{-\beta t} dt \right] = T_i + \frac{1}{P_f \beta} \left(1 - e^{-\beta Tx} \right) - \frac{Tx}{P_f} . e^{-\beta Tx}$$

If there are $In(T_i)$ messages in the network, $E[T_d \mid T_d > T_i]$ then can be expressed as

$$E[T_d \mid T_d \geq T_i] = T_i + \frac{1}{n_i(T_i)P_{fi}\beta}\left(1 - e^{-\beta n_i(T_i)Tx}\right)$$

$$-\frac{Tx}{P_f}.e^{-\beta n_i(T_i)Tx} \tag{14}$$

The above equation does not take into consideration that new copies of message i might be created during its remaining message life time (R_i). To take this fact into consideration, the total asymptotic number of the copies of message i is calculated during Tx of message i, which is calculated as below:

$$n_i(Tx) = \frac{n_i(T_i)N}{n_i(T_i) + (N - n_i(T_i))e^{-\beta P_{fi}NR_i}} \tag{15}$$

The proof of (15) is given in the Appendix. The second term of (14) can be replaced by $\frac{n_i(Tx)}{N-1}$ which represents the cumulative density function (CDF) of message delivery ratio within Tx. The number of the copies of message i in the third term is substituted by $n_i(Tx)$ as well.

The final expression is written as

$$D_i = \left(\frac{N}{N - 1 + e^{\beta P_{fi}NT_i}}\right)^{\frac{1}{P_{fi}}} * \left[T_i + \frac{1}{n_i(T_i)P_{fi}\beta}\left(\frac{n_i(Tx)}{N-1}\right)\right.$$

$$\left. -\frac{Tx}{P_{fi}}.e^{-\beta n_i(Tx)Tx}\right] \tag{16}$$

The proof of (16) is provided in the Appendix.

When a node buffer is full, the node should make a drop decision that leads to the largest decrease in the global delivery delay of message i, D_i . To find the local optimal decision, D_i is differentiated with respect to $n_i(T_i)$, and ∂D_i is then discritized and replaced by $\triangle D_i$:

$$\triangle D_i = \frac{\partial D_i}{\partial n_i(T_i)} * \triangle n_i(T_i)$$

$$\triangle D_i = \left(\frac{N}{N - 1 + e^{\beta P_{fi}NTi}}\right)^{\frac{1}{P_{fi}}} * \left[\frac{-1}{(n_i(T_i))^2 P_f\beta}\right.$$

$$\left.\left(\frac{n_i(Tx)}{N-1}\right)\right]\triangle n_i(T_i)$$

To reduce the delivery delays of all the messages existing in the network, the best decision is to discard the message that maximizes the total average of the delivery delay, $D = \frac{1}{K(t)}\sum_{i=1}^{K(t)} D_i$, among all the messages. Therefore, The optimal buffer-dropping policy that maximizes the delivery delay is thus to discard the message that has the min value of $\mid \frac{\partial D_i}{\partial n_i(T_i)} \mid$, which is equivalently to choose a message with a value for i_{min} that satisfies (12).

Forwarding and Dropping Policy. With the per-message utility, the node firstly sort the buffer messages accordingly from the highest to the lowest. The messages with lower utility values have higher priorities to be dropped when the node's buffer is full, while the messages with higher utility values have higher priorities to be forwarded to the encountered node. Fig. 2 illustrates the forwarding and dropping actions: if the utility u_1 of message j buffered in a is higher than u'_B of message i at node b, then message i is dropped and replaced by a copy of message j if the buffer of b is full during the contact of the two nodes.
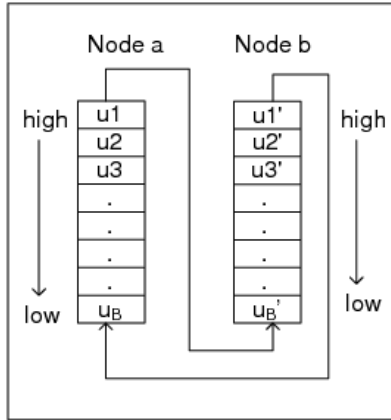

Fig. 2. The forwarding and dropping at a node

5   Simulation

5.1   Experimental Setup

To examine the efficiency of the proposed buffer management architecture, experiments are conducted and presented in this section. To better understand the performance of the proposed estimation strategy–GHP, we also implement two other estimation strategies for the values of $m_i(T_i)$ and $n_i(T_i)$, namely Global Knowledge-based Management (GKM) and Encounter History-Based Prediction (EHP).

The GKM assumes knowing the exact values of $m_i(T_i)$ and $n_i(T_i)$, and is supposed to achieve the best performance. Since such an assumption is not practical[12], the result of GKM is taken as a benchmark for the proposed GHP scheme. With EHP, The two encountered nodes update each other with respect to the messages they have in common, and the values of $m_i(T_i)$ and $n_i(T_i)$ are updated accordingly. This policy of update provides a sub-optimal solution and has been employed in [15] and [11]. In addition to the prediction strategies, we compared the proposed buffer management architecture with a number of counterpart policies listed as follows:

- Drop oldest (DO) drops the message with the shortest remaining time to live.
- Drop front (DF) drops the message that entered the queue the earliest when the buffer is full. This policy obtains the best performance of all the policies used by Lindgren et al. in [10].
- History-based drop (HBD) [11] is based on the history of all messages (on average) in the network after an elapsed time. The variables of the message utility are estimated by averaging the variables of all messages in the network after during the elapsed time.

We assume that a node will never discard a message sourced at the node in favor of a relayed message. It means that the messages issued at a node have the highest priority at the node. If all buffered messages are sourced ones, and the newly arrived message is also a source message at the node, then the oldest one is dropped. This idea was examined in [9] and has been proved with improved delivery ratio. Without loss of generality, random waypoint mobility model is employed in the simulation, where a number of 100 nodes are launched moving e independently on a 500×500 grid with reflective barriers [16]. Each node has a radiation distance as $D \geq 0$ meters, and each message transmission takes one time unit. Euclidean distance is used to measure the proximity between two nodes (or their positions). A slotted collision avoidance MAC protocol with Clear-to-Send (CTS) and Request-to-Send (RTS) features has been implemented in order to arbitrate between nodes that contend for a shared channel. The message inter-arrival time is uniformly distributed in such a way that the traffic can be varied from low (10 messages generated per node) to high (60 messages generated per node). The bandwidth of the network is assumed to be unlimited. Message delivery ratio and the delivery delay are taken as two performance measures of the simulation. Each data is the average of the results from 30 runs.

5.2   Proposed Policy for Maximizing Delivery Ratio

This section examines the proposed policy for maximizing the average delivery ratio. Two scenarios are performed for each routing scheme: varying the traffic load and fixing the buffer capacity, and fixing the traffic and varying the buffer capacity.

Scenario (1): The Effect of Traffic Load. In this scenario, the traffic load varies from 10 to 70 messages generated per node, and the buffer size is set to a low capacity (10 messages). The plots of the delivery rate obtained for epidemic is shown in Fig. 3.

It can be seen that the GKM gives the best performance for all traffic loads, which meets our expectation. The GHP policy provides the next best result and is competitive with the GKM in the case of low traffic. As the traffic increases, the performance of all policies degrades, while the GHP still outperforms all the other policies except GKM. It can achieve a delivery rate 2.15 times higher than that achieved by DO, 1.7 times higher than DF, 1.22 times higher than HBD, 1.32 times higher than EHP, and only 0.15 times worse than GKM.
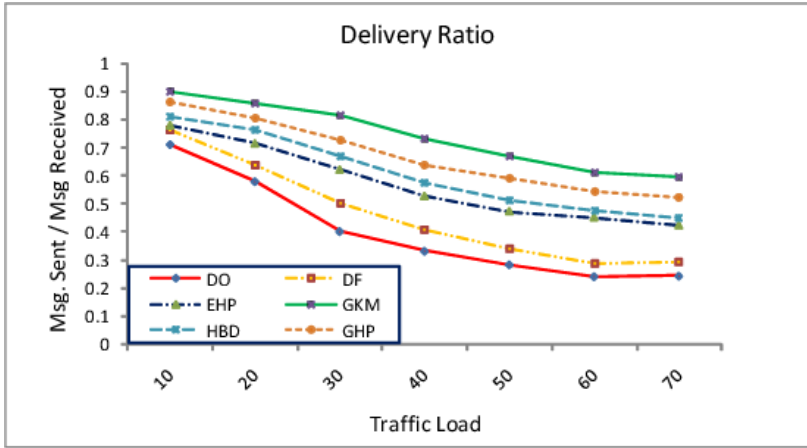
Fig. 3. The effect of traffic load on the delivery ratio

Scenario (2): The Effect of Buffer Size. The results for the second scenario are shown in Fig. 4. The buffer capacity varies from 5 (low capacity) to 600 (high capacity) and the traffic load is fixed at high traffic (70 messages per node). It can be observed that the GKM gives the best performance for all values of buffer capacity, while GHP outperforms all its counterparts except GKM when the buffer capacity is relatively small. As the buffer capacity becomes larger, the performances of all policies improve and become closer to one another. GHP still yields the highest performance in this scenario. For a low buffer capacity, the delivery rate of epidemic routing with its GHP can be 3 times higher than that with DO, 2 times higher than DF, 1.2 times higher than HBD, 1.4 times higher than EHP, and only 0.38 times worse than GKM.

5.3   Proposed Policy for Minimizing Delivery Delay

This section evaluates the effect of the policy of each routing scheme on message delivery delay using the same scenarios in previous section.

Scenario (1): The Effect of the Traffic Load. Figure. 5 shows the results. AS expected, the GKM gives the best performance under all traffic loads for both routing techniques, while the GHP is the second best and is competitive with the GKM in the case of low traffic. As the traffic increases, the demand on the wireless channel and buffers increases, causing a long queuing delays and substantial message loss that negatively affect the performance of all the examined policies. We have observed that the GHP outperforms all other policies, which is better than DO by 1.5 times, DF by 1.40 times, HBD by 1.1 times, EHP by 1.20 times, and a longer delay of only 0.08 of that achieved by GKM.

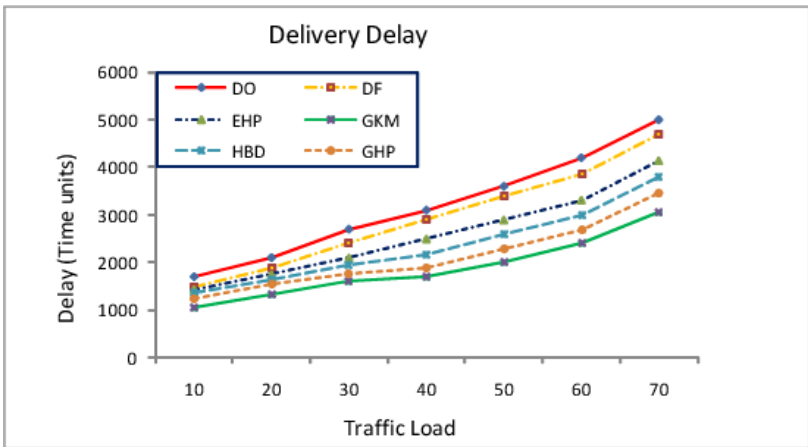**Fig. 4.** The effect of buffer capacity on the delivery ratio

Fig. 5. The effect of traffic load on delivery delay

Scenario (2): The Effect of Buffer Size. Fig. 6 shows the delivery delay obtained for epidemic forwarding with the policies under consideration in the case of varying buffer size and keeping traffic load constant.

As expected again, the GKM of each routing scheme gives the best performance for all values of buffer capacity, while the GHP outperforms all other counterpart schemes except for the GKM when the traffic demand is larger than the buffer capacity. As the buffer capacity became larger, the performances of all policies improve and become closer to one another. GHP still outperforms all other policies. When the buffer capacity is low (5-10 messages), epidemic routing under a GHP policy can achieve a delivery delay shorter than that achieved by
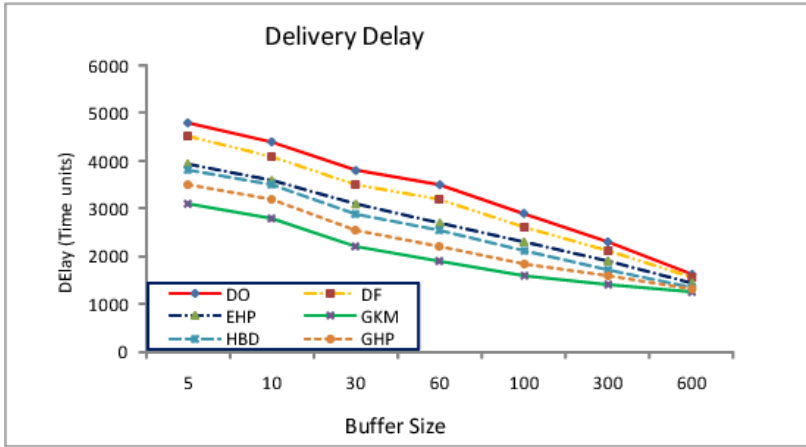
Fig. 6. The effect of buffer capacity on delivery delay

DO by 1.4 times, DF by1.35 times, HBD by 1.20 times, EHP by 1.29 times, and longer delay of only 0.13 of that achieved by GKM. It should be noted that, GHP outperforms all other policies that based on local knowledge about the network.

6 Conclusions and Future Work

This paper has investigated a novel buffer management architecture for epidemic routing in delay tolerant networks, aiming to optimize the message delivery ratio and delivery delay. The proposed architecture incorporates a suite of novel mechanisms for network state estimation and utility derivation, such that a node can obtain the priority for dropping each message in case of buffer full. The simulation results showed that the proposed buffer management architecture can significantly improve the routing performance in terms of the performance metrics of interest under limited network information.

proof of (9): Given $m_i(T_i)$, $n_i(T_i)$, and $P(T_i) = \frac{m(T_i)}{N-1}$, as initial values at T_i , the delivery probability in the interval $t : T_i < t < T_i + R_i$, $P(T_d < T_i + R_i \mid_{T_d > T_i})$, can be constructed using (3) as follows:

$P'_i(t) = \frac{dP}{dt} = \beta n_i(t)(1 - P_i(t))$

$\frac{dP}{1-p} = \beta n_i(t)dt$

$\frac{dP}{1-p} = \beta \frac{Nn_i(0)}{n_i(0)+(N-n_i(0))e^{-P_{fi}\beta Nt}}dt$

Integrate both sides for the interval R_i, we get

$$P(T_d < T_i + R_i \mid_{T_d > T_i}) = 1 - \left[\left(1 - \frac{m_i(T_i)}{N-1}\right)\right.$$

$$\left.\left(\frac{N}{N - n_i(T_i) + n_i(T_i)e^{\beta P_{fi}N(Ri)}}\right)^{\frac{n_i(T_i)}{P_f}}\right]$$

Proof of (15): Given $n_i(T_i)$ as initial value, the expected number of message copies within the interval R_i can be constructed using (1), as follows:

$$n_i^{'}(t) = P_{fi}\beta n_i(t)(N - n_i(t)).$$

$\beta P_{fi}dt = \frac{dn_i}{n_i(N - n_i)}$, by integrating both sides for the interval R_i we get

$$\beta P_{fi}\int\limits_0^{R_i} dt = \int\limits_{n_i(T_i)}^{n_i(T_i + Ri)} \frac{1}{(Nn_i - n_i^2)}dn_i,$$

which lead to:

$$n_i(Tx) = \frac{n_i(T_i)N}{n_i(T_i) + (N - n_i(T_i))e^{-\beta P_{fi}NR_i}}$$

References

1. Delay tolerant networking research group, `http://www.dtnrg.org`
2. Vahdat, A., Becker, D.: Epidemic routing for partially connected ad hoc networks. Technical Report CS-200006, Duke University (2000)
3. Lindgren, A., Doria, A., Schelen, O.: Probabilistic routing in intermittently connected networks. SIGMOBILE Mobile Computing and Communication Review 7(3) (2003)
4. Spyropoulos, T., Psounis, K., Raghavendra, C.S.: Efficient routing in intermittently connected mobile networks: The multiple-copy case. ACM/IEEE Transactions on Networking (February 2008)
5. Groenevelt, R., Koole, G., Nain, P.: Message delay in manet (extended abstract). In: Proc. ACM Sigmetrics (2005)
6. Elwhishi, A., Ho, P.-H.: SARP - A Novel Multi-Copy Routing Protocol for Intermittently Connected Mobile Networks. In: GLOBECOM 2009, pp. 1–7 (2009)
7. Erramilli, V., Crovella, M., Chaintreau, A., Diot, C.: Delegation forwarding. In: Proceedings of the 9th ACM International Symposium on Mobile Ad Hoc Networking and Computing, Hong Kong, Hong Kong, China (May 26-30, 2008)
8. Ip, Y.-K., Lau, W.-C., Yue, O.-C.: Performance modeling of epidemic routing with heterogeneous node types. In: Proc. of IEEE ICC (2008)
9. Zhang, X., Neglia, G., Kurose, J., Towsley, D.: Performance modeling of epidemic routing. In: Proceedings of IFIP Networking (2006)
10. Lindgren, A., Phanse, K.: Evaluation of Queueing Policies and Forwarding Strategies for Routing in Intermittently Connected Networks. In: First International Conference on Communication System Software and Middleware (2006)
11. Krifa, A., Barakat, C., Spyropoulos, T.: Optimal buffer management policies for delay tolerant networks. In: Proc. of IEEE SECON (2008)
12. Nelson, S.C., Bakht, M., Kravets, R., Harris, A.F.: Encounter based routing in DTNs. SIGMOBILE Mob. Comput. Commun. Rev. 13(1), 56–59 (2009), `http://doi.acm.org/10.1145/1558590.1558602`
13. Spyropoulos, T., Turletti, T., Obraczka, K.: Routing in Delay-Tolerant Networks Comprising Heterogeneous Node Populations. IEEE Transactions on Mobile Computing 8(8), 1132–1147 (2009), doi:10.1109/TMC.2008.172

14. Aldous, D., Fill, J.: Reversible markov chains and random walks on graphs (monograph in preparation),
`http://statwww.berkeley.edu/users/aldous/RWG/book.html`
15. Balasubramanian, A., Levine, B., Venkataramani, A.: DTN routing as a resource allocation problem. In: Proceedings of the 2007 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications, Kyoto, Japan (August 27-31, 2007)
16. Spyropoulos, T., Psounis, K., Raghavendra, C.S.: Performance analysis of mobility-assisted routing. In: Proceedings of ACM MOBIHOC (2006)
17. Thomas Karagiannis, M.V., Le Boudec, J.-Y.: Power law and exponential decay of inter contact times between mobile devices. In: Proc. of ACM/IEEE MobiCom (2007)