IOSR Journal of Computer Engineering (IOSR-JCE)
e-ISSN: 2278-0661, p- ISSN: 2278-8727Volume 16, Issue 1, Ver. VII (Feb. 2014), PP 44-51
www.iosrjournals.org

Routing Framework for Delay Tolerant Networks using BayesiaLab

B.Sujatha Asst.professor, CH.Bhavani Asst.professor, A.Nagasri Asst.professor

Abstract: *Routing in delay tolerant networks (DTN) can benefit from the fact that most real life DTN, especially in the context of people-centric networks (e.g. Pocket Switching Networks (PSN)), exhibit some sort of periodicity in their mobility patterns. For example, public transportation networks follow periodic schedules. Even most individuals have fairly repetitive movement patterns, for example, driving to and from work at approximately the same time everyday. This paper proposes a BayesiaLab tool based DTN routing framework that adopts a methodical approach for computing the routing metrics by utilizing the network parameters (e.g. spatial and temporal information at the time of packet forwarding) that capture the periodic behavior of DTN nodes. After the calculation of routing metrics, different routing instantiations are possible based on this framework. We simulate a real-world vehicular DTN network using mobility traces from a metropolitan public transportation bus network and demonstrate that even a simplistic single-copy forwarding scheme based on our framework outperforms existing gradient-based single copy schemes by 25% in terms of delivery ratio. To the best of our knowledge this work is one of the first studies that adopts Bayesian inference in the context of DTN routing.*

I. Introduction

Delay Tolerant Networks (DTN) [3] are a class of challenged networks wherein the node contacts are intermittent and disconnections are common place. To deal with this episodic connectivity in DTN, the proposed routing strategies rely on the inherent mobility of the participating nodes to *store-carry-and-forward* [1] the messages for delivery to the destination. Such networks are well-suited in areas where there is no communication infrastructure due to harsh environments (battlefields, forests, space) or economic conditions (rural and remote areas, developing countries). Two particularly attractive instances of people-centric DTN include (i) *Pocket Switched Networks (PSN)* [7], wherein personal communication devices carried by humans self-organize to form an intermittently connected network, thus enabling a new class of social net- working applications (e.g. PeopleNet [15]) and (ii) Vehicular DTN, which can leverage the large data storage and energy capabilities offered by vehicles to create a large-scale powerful DTN. Examples include the use of vehicle-based DTN to provide low cost digital communication to remote villages [16] and vehicular sensing platforms such as CarTel [9] for urban monitoring.

Over the past years, several routing schemes have been proposed for DTN (an overview of these can be found in [11]).

A particularly effective forwarding principle that is often em- ployed is *gradient routing* [11], wherein the message tends to follow a gradient of increasing utility function values towards the destination. The utility function serves as the routing metric and is based on a variety of parameters, which depend on the *a priori* information about the network characteristics, such as: last encounter [2] with the destination, encounter frequencies of nodes [1], etc.

However, most of the schemes (e.g. [14], [1]) need to maintain routing metrics for all potential destinations, which is clearly not scalable, particularly for large-scale DTN (e.g. PSN or vehicle-based DTN). A more scalable approach, presented in [8], involves grouping of the nodes into a finite set of classes based on certain community affiliations (i.e. institutional affiliation) that exist amongst the nodes. Consequently, nodes only need to maintain routing metrics on a per-class basis. The goal is to transfer the message to any node that belongs to the same class as that of the destination since members of the same class have a high likelihood of encountering each other. However, this type of simple classification may be harder to achieve in larger and complex networks such as people-centric DTN because it is not obvious how and which properties should be utilized for the classification purpose.

In most people-centric DTN, the mobility patterns of the nodes generally exhibit some level of time periodicity. For example, public transport buses follow fixed schedules and routes; also individuals often follow repetitive patterns, for example - driving to and from work at the same time every weekday. Given that the mobility of most real-world DTN follow repetitive patterns to some extent, statistical approaches are particularly suitable for the above mentioned classification of nodes. In this paper, we present a novel routing framework, which utilizes the concept of Bayesian classification [6] for determining the class membership probabilities. The simplicity and accuracy of Bayesian classifier [6] make it an attractive candidate for this classification problem. In our approach the computation of these class membership probabilities relies on the availability of

historical statistics about relevant network parameters (e.g. node mobility traces, node encounter statis- tics, etc.). After computing class membership probabilities, our routing framework employs class-based gradient routing similar to the ideas presented in [8]. The use of posterior probabilities in computing class membership enables us to accommodate more information as compared to prior probabilities.
For example, in vehicular DTN, packet forwarding behavior varies in different periods of a day due to heterogeneous traffic patterns throughout the day. Also landmarks like bus-stops are good place for packet forwarding since there is a high likelihood of finding a suitable forwarder in those areas. The effect of these additional attributes (e.g. time, location) can be factored in methodically by using posterior probabilities.
In particular, we make the following contributions:

- We introduce an extensible routing decision framework based on Bayesian classification, which seamlessly inte- grates knowledge about network characteristics and node mobility patterns in order to make better routing decision.
- We present a simple instantiation of our framework, referred to as *Bayesian*, which uses two simple classes based on prior packet delivery statistics. The routing metrics employed are posterior delivery probabilities conditioned on two attributes, location and time.
- We demonstrate the efficacy of our approach through simulation-based evaluations using mobility traces of a realworld public transport network. The simple *Bayesian* scheme achieves a 25% improvement in the packet delivery ratio as compared to that of MaxProp [1], an effective routing protocol that employs prior probabilities.

The rest of the paper is organized as follows. Section II provides an overview of related work. Section III presents our proposed Bayesian classifier based routing decision frame- work. A detailed example illustrating the use of this framework is also provided. Section IV presents the simulation-based evaluation of our scheme and compares it with other routing strategies. Finally, Section V concludes the paper.

II. Related Work

The idea of using prior information about network char- acteristics has been proposed in prior literature [1], [14], [2], [4], [11]. For example, the routing schemes proposed in [2] and [4] maintain age-of-last-encounter timers and choose a forwarding node that has most recently encountered the destination. In general, most of those routing schemes are based on the principle of Gradient Routing [11]. The basic idea is to transfer the message to the contact that has better delivery metric than the current incumbent. Lindgren et al. [13] introduced a probabilistic routing strategy wherein each node maintains a utility function, which is an exponential weighted moving average of prior contact probability for every other node. This utility function is then used as a metric for gradient routing. Burgess et al. [1] use incremental averaging of node encounters to calculate the delivery predictability.

However, most of the above mentioned stochastic routing protocols use prior probabilities (i.e. the probability of an event regardless of other events) in making a routing decision. For example, MaxProp [1] uses unconditional encounter-based prior probabilities to calculate shortest path. On the contrary, our proposed decision model is based on posterior probabilities, i.e., the probability of an event when relevant other attributes are taken into account. Consequently, this enables our scheme to make a more informed decision in choosing suitable forwarders and thus improves the routing performance in terms of delivery ratio.

In machine learning research, studies [12] comparing classification algorithms have found that even a naive Bayesian classifier performs comparably with Logistic Regression and Support Vector Machine . Text classification, weather predition, large database management (e.g. in NASA's space flight centre) are some of the practical uses of Bayesian classifiers. To the best of our knowledge this is one of first attempts at adopting bayesian inference in DTN routing.

BayesiaLab
A single tool for all your Decision Support and Data Mining Tasks

• Bayesian networks will allow you to model your expert knowledge relative to risk, fraud, customer s' behavior …The graphical representation of Bayesian networks and the BayesiaLab's ease of use make it a invaluable Brain Storming and communication tool.
• You will be able to exploit all the power of unsupervised learning to extract from your data bases the set of probabilistic relations that are really significant. This kind of learning is a real knowledge discovery tool and is very helpful for the understanding of your problems.
• Supervised learning will allow you to characterize your target variable. This variable will represent, for example, the fraud, the propensity to buy a product, or the customer satisfaction. The evaluation of the automatically learnt Bayesian network on an independent test set (cases that have not been used for learning) will return you the model global precision, its confusion matrix (occurrences, reliability and precision) that will

enable you to precisely know the prediction behavior of the network. You will also have an interactive lift curve that will help you finding the threshold representing the best economic compromise for your marketing actions.
• You will also be able to exploit the powerful Markov Blanket Learning algorithm for the selection of the minimal subset of variables that are really useful with respect to the target variable.
• Clustering of your data bases will enable you to discover groups of homogeneous individuals sharing the same characteristics. The HTML report that will give you the probabilistic profile of each identified cluster will help your experts to put a name to these clusters and to use them in your marketing campaigns.
• If Expert knowledge is available, BayesiaLab will rigorously merge it with your data bases.
• BayesiaLab will enable you testing levers effects (e.g. action for image improvement or for training) by manually adding nodes to your learnt networks. By associating cost nodes to these levers, you then will be able to evaluate various action policies.
• The BayesiaLab's adaptive questionnaires will return you the most relevant questions with respect to the information brought to the knowledge of your target variable and with respect to the cost associated to questions. A new set of ordered questions will be automatically returned after each answer.
• You will be able to use your Bayesian networks off-line to automatically classify new cases described in a data base. Two additional fields will be added to each case: the predicted value and its probability.
• By using the BayesiaLab's analysis toolbox, you will really be able to understand your data: analysis of the strength of the relations, analysis of the interaction between your target variable and the other variables, analysis of the relations linking all the variables with a specific value of the target variable, contradiction analysis to know if all the evidences support the same conclusion or if there are some contradicting evidences, causal analysis to transform the arcs that can be inverted without changing the probabilistic meaning of the network into edges.
• You will be able to "play" with your networks to easily test your hypothesis by carrying out What-if scenarios.
• Lastly, you will have access to a rigorous imputation tool that will use your Bayesian network jointly with all the available evidences to compute the probability distribution of your missing values, and then Replace them accordingly.

III. Bay E S I A N Routing Framewo Rk

Our routing framework consists of two phases: 1) classification and 2) packet forwarding. The goal of the first phase is to create an abstract grouping of nodes based on certain prior information about the network. Gradient routing is employed in the second phase, wherein a node tries to forward the packet to a neighbor which has higher *affinity* with the destination, in anticipation that the packet gradually moves towards destination. The affinity of a node with the destination is computed as a posterior probability using Bayesian inference and is based on the history of past packet forwarding behavior which is conditioned on network specific factors such as: node location, time etc. The intuition being that a suitable forwarder which was encountered at the same time of day and location in the past is likely to be a good candidate in the future. Bayesian inference allows us to model the imprecise node mobility and outcomes of interest (e.g. routing success or failure) by combining common-sense knowledge and observational evidence. Its ability to express all forms of uncertainty in terms of probability makes it an attractive tool to quantify a node's probability (i.e. affinity) to meet destination.

The proposed routing framework has two phases which we elaborate in the next section, followed by an instantiation of the framework. It should be noted that depending on the network attributes (e.g. node location, forwarding time, etc.), several routing instantiations are possible based on this framework.

A. Phase One: Bayesian classification

Generally speaking, a Bayesian classifier takes the attributes of an unknown sample and tries to predict its class membership probability based on the history of previously known samples. Let us assume, a forwarding node P has
${}^{x}N_1, N_2, \ldots, N_m$ neighbors which belong to one of the classes ${}^{x}C_1, C_2, \ldots, C_n$, where m n. In practice, a node may belong to more than one classes but its class membership probabilities would usually be different. Now, assume that node P has a packet to send to destination D. Node P and node D are members of classes C_P and C_D, respectively. If node P encounters another node that has a higher affinity towards the destination class C_D, then P should pass on the message to this node.
Let, the attributes (which have direct or indirect effect on the packet delivery) of a node be represented by an attribute vector, $X = {}^{x}x_1, x_2, \ldots, x_n$. Current time, physical location,
contact probability and contact duration with destination are some examples of attributes, which affect packet delivery probability. It is upto the network administrator (or system developer) to determine appropriate classes and attributes for a particular network.

Now, referring back to our example, a node can calculate its affiliation probability with other classes (C_i) conditioned on the attributes X (i.e. posterior probability) as: $P(C_i|X)$. Intuitively, a larger value of the posterior probability would imply better affiliation to that class. The posterior probability $P(C_i|X)$ is based on more information than prior probability $P(C_i)$ because it factors in the effect of the attributes on the class memberships, and is hence able to identify better forwarding candidates. Using Bayes theorem, the posterior probability can be calculated from prior probability as:

$$P(C_i|X) = P(X|C_i)P(C_i)/P(X) \qquad (1)$$

The denominator of Eq. 1, $P(X)$ does not depend on C_i and is only used for normalization. Clearly, $P(C_i|X)$ will be maximum when $P(X|C_i)P(C_i)$ is maximum. To quantify the affiliation of a node to a class, we define a metric called the *Affiliation Index*, which is computed using Eq. 2.

$$\text{Affiliation Index} = P(X|C_i)P(C_i) \qquad (2)$$

$P(X|C_i)$ and $P(C_i)$ can be calculated based on the available historical information.
The term $P(X|C_i)P(C_i)$ is equivalent to the joint prob- ability model of $P(C_i \cap X)$. However, an expansion of $P(X|C_i)P(C_i)$ becomes unmanageable for dimension of $X =$ $x_1, x_2, \ldots, x_n$. If we can assume that every attribute x_i is conditionally independent on every other attribute x_j $(i = j)$ then we can come up with Eq. 3.

$$P(X|C_i) = \prod_{k=1}^{n} P(x_k|C_i) \qquad (3)$$

Despite the naive assumption of conditional independence among the attributes, the naive Bayesian classifier exhibits remarkable accuracy in classification process [17]. Now, the affiliation index can be computed from Eq. 2 and 3 as:

$$\text{Affiliation Index} = P(C_i) \prod_{\forall k} P(x_k|C_i) \qquad (4)$$

The conditional probability $P(x_k|C_i)$ can be estimated from the history data set. Assume that the historical data contains s_i samples that belong to class C_i. Assume that from this sample set, there are s_{ik} samples for which the value of an attribute is equal to x_k. In this case, $P(x_k|C_i) = \frac{s_{ik}}{s_i}$. If it is known that certain attributes follow well-known probability distributions (e.g. exponential, gaussian, etc.), then we can directly usethat distribution in place of $P(x_k|C_i)$. It should be noted that if any of the $P(x_k|C_i)$ in Eq. 4 becomes zero, then the corresponding affiliation index will also be zero, nullifying the effect of other attributes. In order to circumvent this problem, we assume a low value (e.g. 0.01) for $P(x_k|C_i)$ whenever its value becomes zero.

B. Phase Two: Forwarding

The second phase of our framework uses the affiliation indices (computed in the first phase) to forward the packets towards the destination in gradient manner. When a node encounters one or more neighbors, it needs to make a decision on which node is a suitable forwarder. The node can readily compute its own affiliation index with the destination using Eq. 4. However, it also needs to know the affiliation indices of its neighbors with the destination. The node can calculate the class affiliation probabilities for all other neighbors provided that the node knows the attributes of its neighbors. However, calculating the posterior class affiliation probabilities is com- putationally non-trivial. This is especially true if there are a large number of neighbors (i.e. dense topology) and/or if there are a large numbers of classes and attributes to be considered for calculating the posterior probabilities. For example, in vehicular DTN, some regions (e.g. bus stops, road crossings etc.) have dense vehicle distributions and hence the number of neighbors are usually much large in these areas. It should also be noted that these computations need to be performed on per- packet basis. In order to reduce the computation overhead, the forwarding node distributes the calculation of the affiliation indices among its neighbors using a simple query-response based protocol: the forwarder broadcasts a request containing its own affiliation index with the destination and overhearing neighbors respond it back with their own affiliation indices. As a result, the forwarder can select the neighbor which has maximum affinity towards destination.

In the next sub-section, with the help of an illustrative example, we discuss a simplified single-copy version of our scheme. A multi-copy version can also be implemented in a straight forward manner.

C. Example

We present a simplified example to illustrate the operation of our routing protocol. We consider the context of a vehicle- based DTN. It is assumed that the past delivery statistics are already known. In public transportation networks, where nodes are equipped with huge storage, computing power and sensors, these information together with nodes' coordinate can be logged as the nodes deliver packets to destinations.
We assume a very basic classification scheme consisting of
only two classes: *delivered*, which includes nodes which acted as relays in successfully delivering packets to the destination, and *not-delivered* which indicates nodes which failed to deliver packets. For simplicity, we have chosen two attributes ${}^{\chi}x_1, x_2$ which are:

- x_1 = *Region Code* ${}^{\chi}R_1, R_2, \ldots, R_m$, identifying the location where the node was situated at the time of packet forwarding. We assume that the entire physical domain of the network is divided into m smaller rectangular grids of equal size, each with a distinct identifier. The motivation for using location as an attribute is because in most DTN, the packet delivery probability depends on the physical location of the nodes. For example, in vehicular DTN, the probability of finding a suitable forwarder is higherin the vicinity of landmarks like bus-stops and parking lots, than on isolated suburban roads.
- x_2 = *Time Slot* ${}^{\chi}T_1, T_2, \ldots, T_n$, indicating the time slot when the packet was forwarded. This metric has been chosen because packet delivery also depends on different time periods of a day. In vehicle-based DTN, different traffic patterns are prevalent in different periods of a day and hence the packet forwarding behavior also varies with time. The granularity of the time slot as well as region code can be increased or decreased depending on the availability of historical data. In general, choosing finer grained time slots will result in a more precise affiliation index but requires larger historical records.

In the rest of this section, we explain the operations of our proposed scheme in the context of a vehicular DTN.

A sample topology of a vehicular DTN and past packet forwarding statistics are shown in Fig. 1. The vehicles repre- sent nodes and the links represent the connectivity among the neighbor nodes. Each node maintains a history of past packets that have been forwarded. The history can be gathered via the the propagation of acknowledgments (details in Section IV-C). Each tuple of the history database contains: (a) destination address of the packet, (b) region code where the node was situated at the time of packet forwarding, (c) time slot when the packet was forwarded and (d) the class of the packet, i.e., whether the packet was successfully delivered to the destination or not). We use a simple YES and NO to indicate class memberships. For example, the first entry of the history data maintained by node A (${}^{\chi}D, R_1$, T_2 , YES) indicates that while node A was in region R_1 , it had forwarded a packet (with destination D) at time slot T_2 and the packet eventually reached the destination.
Let us assume that node A has a packet to send to node
D; A is in region R_2 and the current time slot is T_1. Node A calculates its *affiliation index* for D using Eq. 2, which is elaborated below. The prior class probability (for class $C_{delivered}$) is: P $(C_{delivered})$ = 0.5 (since there are two entries in the training dataset of node A for destination D, from which one tuple belongs to the class labeled *yes* or next hop. This process continues until the packet reaches the destination or is discarded (due to expired TTL value).

IV. Simulation-Based EvalUatIONs

To demonstrate the efficacy of our approach we have carried out a simulation-based evaluation in the context of a vehicular DTN. We consider a metropolitan public transport bus network [10] and assume that each bus is equipped with a wireless radio, thus simulating a large-scale DTN. We choose to use a public transport network because of the inherent repetitive nature of the nodes movements.
We use the simple example discussed in Section III-C (with two classes *delivered* and *non-delivered* and two attributes: *region code* and *time slot*) as an instantiation of our framework. This is referred to as *Bayesian* in the rest of the simulations. We compare this scheme with three other routing strategies: (i) *Epidemic* [18], (ii) single copy version of *MaxProp* [1] and (iii) *Wait* (or direct delivery) [11]. Epidemic and Wait exhibit greatly contrasting properties. Epidemic routing is known to achieve the best case delivery ratio with minimum delay since it relies on flooding. On the contrary, Wait requires the minimum possible cost to deliver a packet since the packet is directly forwarded to the destination without the need for relays. In this study, we assume that there are sufficient buffers at each node, since we wish to exclusively focus on the performance of the routing strategies. The simulations have been conducted using a custom built simulator and PostgreSQL has been used as the back-end database server for storing and processing the history data set at each node.

A. Details of Mobility Traces

We have used the mobility traces of buses in the King County Metro bus system in Seattle, USA [10] to simulate a DTN network. This public-transport system consists of 1163 buses plying over 236 distinct bus routes covering an area of 5100 square kilometers. The traces were collected over a two week period in November 2001. The traces are based on location update messages sent by each bus.

Our *Bayesian* protocol requires information about the loca-*delivered*; hence the probability is: 1 2 or 0.5). Now, in order tion (region code) and time (time slot). For this purpose, we to calculate $P\ (X|C_i)$ where $x_1 = R_1$, $x_2 = T_1$, we need to calculate: $P\ (x_1|C_{delivered})$, $P\ (x_2|C_{delivered})$. From the history data of A (Fig. 1), we find, $P\ (x_1|C_{delivered}) = 0.01$ and $P\ (x_2|C_{delivered}) = 0.01$. Recall that null values need to be replaced by a small representative value (0.01 in this example). Now the node can calculate the *affiliation index*, $\boldsymbol{I}$, using, Eq. 2 as, assume that the entire region under consideration is divided into 1km × 1km square grid blocks. In general, selection of smaller grid size results in a more precise affiliation index but increases the amount of historical data that needs to be maintained to calculate statistically significant posterior probabilities. For similar reasons, the granularity of the time slots is chosen as 10 minutes (600 seconds).

$\boldsymbol{I} = P\,(C_{delivered})P\ (x_1|C_{delivered})P\ (x_2|C_{delivered})$

or, $\boldsymbol{I} = 0.5 \times 0.01 \times 0.01 = 0.00005$

Node A broadcasts a request packet containing destination node and its delivery index $\langle D, \boldsymbol{I} \rangle$ to its neighbors. Upon re- ceiving the request, the neighbors B and C also calculate their own affiliation index, which are 0.125 and 0.1437 respectively. Since these are greater than that of node A, both nodes transmit their respective indices to A. Node A then chooses C as the

Though we have mobility traces for the entire 24 hour duration of a weekday, we focus on a 9 hour period from 6am -

3pm to ensure that the simulations are tractable. This period is sufficient to capture the periodicity of the bus mobility, since a typical trip along a bus route takes 2 - 2.5 hours. For the same reasons, we have concentrated on a 58 km x 88 km region, which includes the central business district of the city. Further,

History database maintained by Node B

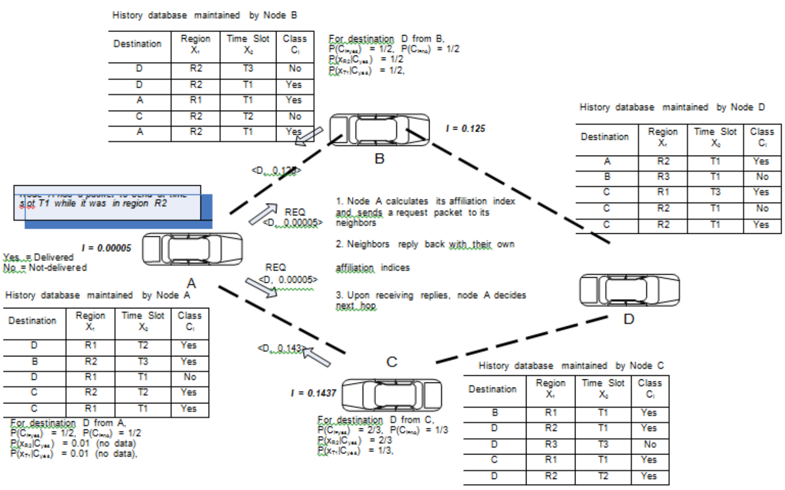


Fig. 1. Illustrative Example of our Routing Framework

we only consider the 35 buses whose routes are for the most part contained within the region of interest.

We use 50 random source-destination pairs (from the combination of 35 × 35 unique source-destination pairs) where each source is a CBR node which transmits a new packet of size 1000 bytes every 100 seconds. The source nodes begin to transmit packets at time $1000+t_{rand}$ ($0 < t_{rand} < 4600$) after the simulation starts in order to ensure that all the buses in our trace file become active by this time. The whole simulation runs for 32400 seconds (i.e. 9 hours). The entire simulation is repeated 20 times with different sets of source-destination pairs and the results presented are averaged over these runs.

C. Gathering Packet Traversal History

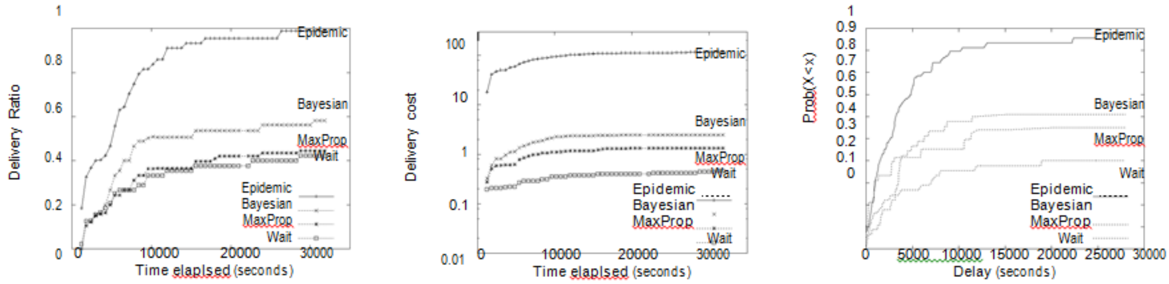
Recall that our routing framework requires prior statistics about certain network parameters, which can serve as the input to the Bayesian classifier. In particular, the example *Bayesian* protocol requires past packet delivery statistics and information about the region code and time slot when these packets were forwarded. In order to emulate the requisite history, we simulated 500 sample runs of the simulation with exactly the same parameters as discussed in Section IV-B. The only difference is that we used a 9 hour period (6am - 3pm) for the weekday prior to that used for the actual evaluations. Though we have not updated the history in our simulation (since lack of updated history only affects long-lasting simulations), it can be easily done in real-world scenario. In our instantiation, the nodes need to know if the forwarded packets eventually reach the destination or not. Nodes which deliver the packets to the destination can readily log these packet traversal history. Or, some form of acknowledgment packets can be used to let the forwarders to update their records. In fact, in many multi-copy DTN routing protocols, acknowledgment packets (e.g. *vaccine* [5]) are sent back to the sender so that the sender and intermediate nodes can purge the packet from their queues. These acknowledgments can be used to update the history.

D. Delivery Ratio Results

The most important metric in DTN routing is the delivery ratio, which is the ratio of the messages delivered to the mes- sages created. As seen from Fig. 2a, the delivery ratio achieved by Epidemic routing is near perfect, which is expected. Wait performs the poorest since it relies on direct delivery. Bayesian outperforms MaxProp by 25%. The reason behind that is, MaxProp suffers from inferior quality encounter probabilities (due to its prior probability calculation method which does not depend on other network parameters) and hence often makes non-optimal routing decisions. It should be noted that MaxProp performs path lookup upto *n* hops away to find the best route to the destination. This method of searching all possible paths is clearly not scalable in large networks. On the contrary, our framework computes the posterior probabilities in a distributed manner and hence can scale easily.

E. Delivery Cost Results

We measure the delivery cost by counting the total number of messages transmitted and normalizing it by the total number of unique message created. The Epidemic routing protocol achieves its higher delivery ratio with a very high cost. Fig.
2b shows the semi-log plot of delivery cost vs. time. The delivery cost is about 30 times less in our method compared to that of Epidemic routing. The delivery costs of MaxProp and our proposed method are similar, though Bayesian exhibits marginal better performance. This is because they both are



(a) Comparison of Delivery Ratios (b) Comparison of Delivery Costs (c) Cumulative Delay Distributions

Fig. 2. Simulation Results

single copy schemes and the delivery costs have been cal- culated from the delivered packets only, neglecting the fact that the schemes have different delivery ratios. So the single- copy routing methods which have low delivery ratios have low delivery costs in general. Besides, we have not considered the added cost of exchanging the summary vectors which is an integral part of MaxProp. The need for exchanging summary vectors also limits MaxProp's deployment in large networks.

F. Delay Distribution Results

Though a large packet delay is common in delay tolerant networks, it is always desirable to receive a packet as early as possible. In order to get an idea of packet delivery delay, we plot the cumulative delay distribution of the routing protocols in Fig. 2c. Epidemic routing achieves the minimum possible packet delivery delay (receives 50% of the packets during first≈3000 seconds) whereas the direct delivery approach (Wait) shows worst case delivery delay (receives only 20% packets during the first 3000 seconds). Bayesian and MaxProp again exhibit similar properties with Bayesian having a slight edge. It can be concluded from the results above that the deliv- ery ratio of even a simple implementation (which uses two simplistic classes and just two attributes) of our framework (which is a single-copy method) is about 60% of the best case (i.e. multi-copy Epidemic), while incurring an overhead that is 30 times less than that of the base case. The delivery ratio is significantly better than that of MaxProp. Also, it is intuitive that the use of better attributes further improves the performance of routing protocol.

V. Conclusion

In this paper, we present a Bayesian classifier based routing decision framework which simplifies the integration of various routing attributes and utilizes the repetitive nature of people- centric DTN in making better routing decisions. Preliminary simulation study with traces from a real-world delay tolerant public transport network demonstrates considerable perfor- mance gain (about 25% improvement in terms of packet delivery ratio) than existing gradient-based routing schemes.

It is intuitive that choosing proper attributes and classes in our routing framework may increase the performance of the routing protocol even further. We intend to investigate methodologies for choosing appropriate attributes and classes for our routing framework. We also plan to evaluate the effectiveness of our framework in other types of DTN with imprecise schedules (e.g. PSN, etc.).

References

[1] J. Burgess, B. Gallagher, D. Jensen, and B. N. Levine. Maxprop: Routing for vehicle-based disruption tolerant networks. In *Proceedings of IEEE INFOCOM*, April 2006.

[2] H. Dubois-Ferriere, M. Grossglauser, and M. Vetterli. Age matters: Efficient route discovery in mobile ad hoc networks using encounter ages. In *Proceedings of ACM MobiHoc*, June 2003.

[3] K. Fall. A delay-tolerant network architecture for challenged internets. In *Intel Research Technical Report, IRB-TR-03-003*, February 2003.

[4] M. Grossglauser and M. Vetterli. Locating nodes with ease: Last encounter routing in ad hoc networks through mobility diffusion. In *Proceedings of IEEE INFOCOM*, March 2003.

[5] Z. J. Haas and T. Small. A new networking model for biological applications of ad hoc sensor networks. In *ACM Transaction on Networking*, volume 14, 2006.

[6] J. Han and M. Kamber. *Data Mining: Concepts and Techniques*. Morgan Kaufmann Publishers, ISBN: 1558609016, 2001.

[7] P. Hui, A. Chaintreau, J. Scott, R. Gass, C. Diot, and J. Crowcroft. Pocket switched networks and human mobility in conference environ- ments. In *Proceedings of SIGCOMM*, August 2005.

[8] P. Hui and J. Crowcroft. How small labels create big improvements. In *Proceedings of International Workshop on Intermittently Connected Mobile Ad hoc Networks*, March 2007.

[9] B. Hull et al. Cartel: A distributed mobile sensor computing platform.In *Proceedings of ACM SenSys*, October 2006.

[10] J. G. Jetcheva, Y. C. Hu, S. P. Chaudhuri, A. K. Saha, and D. B. Johnson. Design and evaluation of a metropolitan area multitier wireless ad hoc network architecture. In *Proceedings of fifth IEEE workshop on Mobile Computing Systems and Applications*, October 2003.

[11] E. P. C. Jones and P. A. S. Ward. Routing strategies for delay-tolerant networks. In *Proceedings of ACM SIGCOMM*, August 2004.

[12] P. Langley, W. Iba, and K. Thompson. An analysis of bayesian classifiers. In *Proceedings of the 10th National conference on Artificial Intelligence*, July 1992.

[13] A. Lindgren, A. Doria, and O. Schelen.Probabilistic routing in intermittently connected networks. In *Proceedings of ACM MobiHoc*, June 2003.

[14] S. Merugu, M. Ammar, and E. Zegura. Routing in space and time in networks with predictable mobility. Technical report, Georgia Institute of Technology Technical report, GIT-CC-04-07.

[15] M. Notani, V. Srinivasan, and P. Nuggenhalli. Peoplenet: Engineering a wireless virtual social network. In *Proceedings of ACM MobiCom*, October 2004.

[16] A. Pentland, R. Fletcher, and A. Hasson. Daknet: Rethinking connec- tivity in developing nations. In *IEEE Computer*, volume 37(1), January 2004.

[17] I. Rish. An empirical study of the naive bayes classifier. In *IJCAI Workshop on Empirical Methods in Artificial Intelligence*, August 2001.

[18] A. Vahdat and D. Becker. Epidemic routing for partially connected ad hoc networks. In *Technical Report CS-200006, Duke University*, 2000.