Multi-copy Routing with Trajectory Prediction in Social Delay-Tolerant Networks

Quan Yuan*, Ionut Cardei†, Jing Chen‡, and Jie Wu§
*Dept of Math and Comp. Sci., University of Texas-Permian Basin, TX, USA
†Dept. of Comp. Sci. and Eng., Florida Atlantic University, Boca Raton, FL, USA
‡Computer School, Wuhan University, Wuhan, China
§Dept. of Comp. & Info. Sci., Temple University, Philadelphia, PA, USA

***Abstract*—Routing in Delay-Tolerant Networks (DTNs) remains a challenging problem due to sporadic connectivity and high delays. To deal with this issue, researchers have investigated multi-copy schemes with predicting future contacts. However, most of the previous work has focused on the prediction of future contacts, without sufficiently considering contact times. This paper proposes McRTP, a multi-copy routing protocol with trajectory prediction for social DTNs. Based on estimating the probability distribution of future contact times, McRTP evaluates and selects multiple paths with the highest delivery probability for routing. Also, to control network traffic overhead, we develop a copy count assignment scheme limiting the number of message copies in networks. A simulation study shows that, with the estimation of both contact probability and contact times, and the path selection scheme, our McRTP outperforms traditional DTN multi-copy routing schemes, especially in sparse networks.**

***Index Terms*—DTN, multi-copy routing, trajectory prediction.**

I. Introduction

As handheld mobile wireless devices are increasingly developed, people can cooperate to establish intermittently connected networks for communication in the absence of base station support. Due to such economic and ad hoc features, delay tolerant networks (DTNs) [5], [14], [16] have been applied in many challenging environments (such as the battlefields, suburban areas, etc.). In DTNs, since wireless links are short-lived and end-to-end connectivity turns out to be sporadic, most routing methods utilize the *store-carry-and-forward* paradigm in which messages are stored at nodes and are forwarded only when an opportunity (meeting a relay node/destination) occurs.

One main category of those routing algorithms is prediction-based [3], [6], [11], [13], [15], where inter-node contacts and mobility behaviors are predicted, generally using prior contact history. The next hop to which a message is forwarded is determined based on such predictions in order to maximize a quality of service (QoS) metric (e.g. delay or delivery ratio). Further, the prediction-based methods can be divided into two types: *single-copy routing* and *multi-copy routing*. In single-copy routing, there is only a single copy for each message to forward until that copy reaches its destination. Even though single-copy routing protocols have the minimum traffic overhead, due to missing contacts and congestion issues, their performance in terms of delivery delay and delivery ratio could be further improved. For that, multi-copy routing schemes, which generate several copies of the same message and route them independently, are proposed. However, most of the existing prediction-based multi-copy routing protocols mainly focus on the prediction of future contact probability, without sufficiently considering the *contact times* which indicate when the future contact occurs. We believe that investigating the contact timing information can improve the prediction accuracy, and consequently, enhance routing performance.

Also, in most DTNs, since mobile devices are usually on humans or animals (like in battlefields, wild fields, etc.), many studies noted that DTNs have a social network nature [1], [2], [17], [18]. That is, nodes are grouped into communities such that the nodes within the same community have similar behaviors, and the nodes from different communities behave differently. Such features determine that the movement of nodes in social DTNs and the contacts between nodes are not completely random. Instead, nodes follow a semi-deterministic trajectory, with small deviations. For example, consider a school network where students are intermittently connected using different wireless devices (i.e. smartphones). All students have their own class schedules, and a student's movement is conducted based on his or her schedule with some small deviations, such as sick leaves. Also, students taking the same class have a larger probability of contacting each other than the students from other classes. When designing routing protocols for such networks, such social nature and node mobility patterns could be utilized to improve the prediction accuracy of future contacts.

In this paper, we propose a multi-copy routing protocol with trajectory prediction for social DTNs (McRTP). In McRTP, we first introduce the prediction model based on the social nature. With that, McRTP evaluates and selects multiple paths with the highest probability for delivery before a message expires. Also, to achieve good scalability, McRTP limits the total number of copies in the network for each message. In summary, our contributions in this paper are as follows:

- We refine the prediction model for social DTNs. Based on that, we design a candidate path selection algorithm to evaluate possible paths, and to qualify proper forwarders during delivery.
- A copy count assignment mechanism is proposed to control the total number of copies in networks, which can

avoid unaffordable traffic overhead caused by the multi-copy scheme.
- We validate the performance of McRTP, and compare the results against several popular DTN routing protocols through simulation in DTNs with different network sizes.

The reminder is organized as follows. In Section II, we review the existing methods that inspire our work. Section III discusses the key components of McRTP. Section IV presents the simulation results, and the paper is concluded in Section V.

II. Related Work

One option for DTN routing is prediction-based, where a node's future mobility is estimated based on historical observations to determine the optimal forwarding schedule. A typical example is in terms of utility-routing [9], where each node maintains a utility value for every other node based on a timer, indicating the time elapsed since the two nodes last encountered each other. Nodes forward message copies only to nodes with a higher utility for the message's destination. The utility value is considered as the prediction of two nodes' future contacts. Existing prediction-based DTN routing can be classified into: *single-copy routing* [3], [7], [11], [15], and *multi-copy routing* [4], [8], [10], [13], which trade off between the consumed network resource and routing performance. Burgess et al. [3] presented MaxProp, which mainly relied on the prediction of the path likelihoods to peers according to historical contact data for the single copy delivery. [15] proposed a single-copy based passive routing protocol in DTNs, where a prediction model was employed to estimate nodes' contacts. On the other hand, multi-copy routing schemes can achieve a better delivery ratio, by consuming more network resources like channel bandwidth, traffic overhead, and network storage, which are all limited in DTNs. In [4], Burns et al. proposed a routing protocol that made use of past frequencies of contacts, as well as the past contacts, which allowed two copies of messages for delivery. LeBrun et al. [8] proposed a routing algorithm for vehicular DTNs using current position and trajectories of nodes to predict their future distance to the destination. In [10], Spyropoulos et al. introduced Spray and Wait, in which a fixed number of copies of a packet were replicated and routed based on probability prediction in a random mobility network. However, most of those multi-copy prediction-based protocols focus on whether two nodes have future contacts, without sufficiently considering when those future contacts occur. Our approach not only utilizes multiple copies to improve delivery ratio, but also estimates future contact probabilities and contact times to increase the accuracy of contact prediction for routing.

III. Prediction-based Multi-copy Routing

A. System Model and Assumptions

We consider that a social DTN consists of l geo-communities (L_1 to L_l) in the network, and nodes that move between those geo-communities. Moreover, any two nodes that are at the same geo-community at the same time can exchange packets directly. A node arrives at a geo-community and stays there some time before it leaves towards the next geo-community. When traveling between two geo-communities, a node cannot establish contacts with others. A node's trajectory is defined by the sequence of the visited geo-communities, the time spent at each visited geo-community, and the time moving between geo-communities. A sample network with four geo-communities is shown in figure 1. We assume the node trajectory is semi-deterministic, which means that the node follows a scheduled trajectory. Trajectory is subject to a few random deviations that affect the selection of the next geo-community, the travel time between geo-communities, and the dwell time at a geo-community. This network model can successfully approximate a real-world social DTN on a school campus, which is formed by smartphones relying on WiFi operated in ad-hoc mode. The node mobility is predictable since it relies on a class schedule, and nodes in the same class may have the similar mobility schedules.

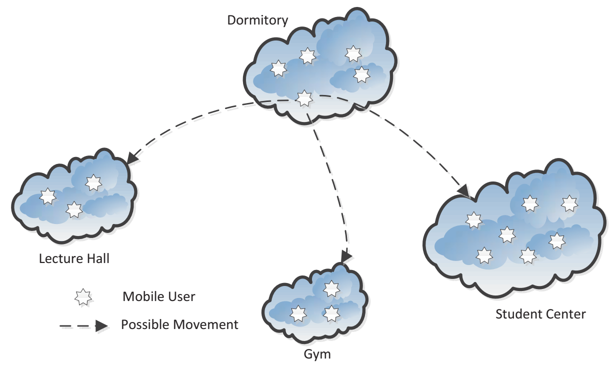


Fig. 1. A sample network model of social DTNs with four geo-communities

B. Protocol Overview

As we mentioned, McRTP is a multi-copy DTN routing protocol, where a node can forward one or more copies of the same message to different nodes, during successive contacts. The number of copies for a message m to forward is a message property m_c, called *copy counter*, and is decremented after each forwarding. When m_c reaches 1, the node stops forwarding that message to any other node, except the destination. For instance, if at node a, $m_c = 7$ and node a decides to forward 3 copies to node b, then a creates message m' (a copy of m), and sends it to b with the message property $m'_c = 3$. Meanwhile a updates its buffered message's copy counter: $m_c = m_c - m'_c = 7 - 3 = 4$. Node b will then have to forward its m'_c copies on its own. Copy counter m_c is initialized at the message source with a constant $C (C \geq 1)$, which limits the total number of transmissions for that message in the entire network to C. After that, McRTP controls m_c based on the contact prediction. Node a that has m_c copies of message m evaluates the possible paths from a to the destination, picks the most likely ones to deliver the message, and forwards sufficient copies during its contacts. Each message has TTL (time-to-live), represented by m_{ttl}. When the age of a message reaches its m_{ttl}, the message is discarded. Our McRTP protocol has three main components: 1) candidate paths selection, 2) copy count assignment, and 3) message forwarding.

C. Contact Prediction

We first model the time evolution of node trajectories with a Time-Homogeneous Semi-Markov process, as in our prior work [15], where there are l geo-community states (indicating a node is staying at a geo-community), and $l^2 - l$ travel states (meaning a node is moving between two geo-communities). We use L_xL_x to denote the state that a node is staying at geo-community L_x ($1 \leq x \leq l$), and L_xL_y to present the travel states that a node travels from geo-community L_x to L_y ($1 \leq x, y \leq l$). A total order for the set of Markov states M can be defined as $L_1L_1, L_1L_2, ..., L_1L_l, L_2L_1, ..., L_2L_l, ..., L_lL_1, ..., L_lL_l$. Hence, the states in set M are indexed from 1 to l^2 accordingly. From now on, when we refer to a state we mean a state from M. The mobility of a node can be described in time as a sequence of state transitions in the Markov process.

Let (X_n, T_n) be the discrete Markov process defined by the random variable $X_n \in M$ indicating the state reached by a node at its n^{th} transition, and random variable $T_n \in \mathbb{N}$ specifying the time of that transition. Define $p_{ij} = P(X_{n+1} = j|X_n = i)$ as the state transition probability from Markov state i to j ($1 \leq i, j \leq l^2$), and matrix $P = |p_{ij}|_{i,j \in M}$. Note that the Markov memoryless property applies to this model, so that p_{ij} does not depend on transitions prior to X_n. We then define $S_{ij}(k)$, the probability that a node will move from state i to j within k time units, as $S_{ij}(k) = P(T_{n+1} - T_n \leq k|X_{n+1} = j, X_n = i)$. Here we call $S_{ij}(k)$ as *sojourn time probability distribution*. Also, the probability distribution of the state residence time $S_i(k)$, describing the probability that a node leaves a state i within k time units, regardless of the next state, is $S_i(k) = P(T_{n+1} - T_n \leq k|X_n = i) = \sum_{j=1}^{l^2} S_{ij}(k)$. The homogeneous Semi-Markov kernel Q of this process is:

$$\begin{aligned} Q_{ij}(k) &= P(X_{n+1} = j, T_{n+1} - T_n \leq k|X_0, ..., X_n; \\ &\quad T_0, ..., T_n) \\ &= p_{ij}S_{ij}(k). \end{aligned}$$

The random variable Z_k ($k \in \mathbb{N}$) indicates in which state the node will be at time k. The node trajectory prediction is given by distributions $\phi_{ij}(k) = P(Z_k = j|Z_0 = i)$, which is the probability that the node is in state j at time k, provided it was at state i at time 0. $\phi_{ij}(k)$ is computed as follows:

$$\begin{aligned} \phi_{ij}(k) &= P(Z_k = j|Z_0 = i) \\ &= (1 - S_i(k))\delta_{ij} + \sum_{r=1}^{l^2}\sum_{\tau=0}^{k} \dot{Q}_{ir}(\tau)\phi_{rj}(k-\tau) \quad (1) \end{aligned}$$

δ_{ij} is the Kronecker symbol, equal to 1 if $i = j$, and 0 otherwise. $\dot{Q}_{ij}(k)$ is the probability the node transitions from state i to j at time k:

$$\dot{Q}_{ij}(k) = \begin{cases} Q_{ij}(1) & \text{for} \quad k = 0 \\ Q_{ij}(k) - Q_{ij}(k-1) & \text{for} \quad k > 0, \end{cases} \quad (2)$$

Note that distributions $S_{ij}(k)$ and probability matrix P can be estimated from history information, and $\phi_{ij}(k)$ can be computed iteratively with Eq.1, starting from $\phi_{ij}(0) = \delta_{ij}$. Two nodes will be in contact at time k if both will be at the same geo-community $L_x (1 \leq x \leq l)$ at time k. Hence, assuming that two nodes a and b are at states i_a, i_b at the past times t_a, t_b, respectively, the contact profile $C_{ab}(k)$ that gives the contact probability of those two nodes at time $k \geq \max\{t_a, t_b\}$ can be computed as in Eq.3:

$$C_{ab}(k) = \sum_{x=1}^{l} \phi^a_{i_a(L_xL_x)}(k - t_a)\phi^b_{i_b(L_xL_x)}(k - t_b) \quad (3)$$

The details of the above model could be referred to in our prior work [15].

D. Path Selection

Based on the above contact prediction, McRTP evaluates the possible paths from the source to the destination. Specifically, a source node computes a set of routes with the highest probability of delivery before a message expires. We define the delay distribution $d()$ as the probability of a message that at time T is at node a, to be delivered at node b at time $T+t$ using direct transmission:

$$d(T, t, ab) = \begin{cases} 1 \\ (a \textit{ and } b \textit{ are in contact at time } T) \\ C_{ab}(T+t)\prod_{k=0}^{t-1}(1 - C_{ab}(T+k)) \\ (a \textit{ and } b \textit{ not in contact at time } T \textit{ and } t > 0) \\ 0 \\ (\textit{if } a \textit{ and } b \textit{ not in contact at time } T \textit{ and } t = 0) \end{cases}$$

The 3^{rd} parameter of $d()$, ab, is the path – a sequence of nodes that the message must pass through. The delay distribution $d(T, t, R_1R_2)$ is then extended for a path made of the concatenation of two adjacent sub-paths R_1 and R_2 (such as $R_1 = 1, 2, 3$ and $R_2 = 3, 4, 5$). It can be derived similar to the convolution, and is equal to

$$d(T, t, R_1R_2) = \sum_{k=0}^{t} d(T, k, R_1)d(T+k, t-k, R_2)$$

Consider there are two acyclic paths R_1 and R_2 that both begin in node a and end in z, with no other common nodes. $R_1 = a...z$, $R_2 = a...z$ are disjoint except for the end nodes, $R_1 \cap R_2 = \{a, z\}$. The message delivery delay from a to z is the minimum of the delay on path R_1 and the delay on path R_2. Therefore, the delay distribution for a message located at node a at time T, transmitted in parallel on *both* paths, can be written using the *parallel composition operator* $|$:

$$\begin{aligned} d(T, t, R_1|R_2) &= d(T, t, R_1)|d(T, t, R_2) \\ &= d(T, t, R_1)(1 - \sum_{k=0}^{t} d(T, k, R_2)) \\ &+ d(T, t, R_2)(1 - \sum_{k=0}^{t} d(T, k, R_1)). \end{aligned}$$

It can be shown that the parallel composition operator is associative and commutative. Path disjointness is required for the independence condition. The probability that a message transmitted at time T on a path R arrives with maximum delay t is given by Eq.4:

$$D(T, t, R) = \sum_{k=0}^{t} d(T, k, R) \quad (4)$$

D is called the *maximum delay distribution* for messages on path R before time t, and it is used by McRTP as a prediction metric to select the most promising paths.

Since messages expire after TTL, the mobility prediction horizon should be limited by the remaining time-to-live. For a message m created at time m_t, the time-to-live computed at time T is $ttl(m) = m_t + m_{ttl} - T$. Our forwarding algorithm computes the *maximum delay distribution* D at the source for all possible paths of lengths 2, ..., λ, from the current node s to destination d: for length 2 ($s \rightarrow i \rightarrow d$), and 3 ($s \rightarrow i \rightarrow j \rightarrow d$). Considering that longer paths bring diminishing prediction accuracy and increasing processing overhead, λ is limited to 3 in this protocol. Denote the set of all these paths as $\mathcal{R}$, $|\mathcal{R}| < n^{\lambda-1}$. Then, use $D_R = D(T, ttl(m), R)$ to represent the maximum delay probability on path R. The paths in $\mathcal{R}$ are then sorted by a decreasing D_R value, and the corresponding paths are written in order as $R_1, R_2, ..., R_{|\mathcal{R}|}$ (Path R_1 has the maximum value $D(T, ttl(m), \cdot)$ from all paths). The algorithm is shown in Algorithm 1.

Assume that the source node s has $m_c - 1$ copies of message m to give up (one is reserved for direct transmission). At this point the algorithm selects a maximal subset $Q \subseteq \mathcal{R}$ so that the number of copies needed for forwarding on all paths in Q is less than $m_c - 1$, and the probability of delivery before $ttl(m)$ combined on all paths in Q is maximized. This problem is similar to the *0-1 Knapsack* Problem and is NP-hard. Finding the optimal subset Q of paths has a very high computational complexity, more so due to the parallel composition operator $|$ (see Eq.4) that must be used, compared to simple addition. Our protocol relies on a suboptimal greedy heuristic: it adds to set Q paths R from $\mathcal{R}$ with decreasing D_R value until the paths in Q need more copies than $m_c - 1$.

E. Copy Count Assignment

While the candidate paths set Q is built, the algorithm populates a vector β with n elements. Element β_i in β is the total number of m's copies needed for all paths in set Q that begin with node s_i.

A helper function $CountCopies(c, Q, \beta)$ computes the total number of message copies needed to send a message from a node s to destination d on all paths in the candidate paths set Q. Each element $R \in Q$ is a path $s...d$ with the maximum length (number of edges) $\lambda (\lambda \leq 3)$. If during the counting of copies, the current count reaches parameter c, the function returns c. Therefore, the complexity of $CountCopies(c, Q, \beta)$ is $O(\min\{c, n^{\lambda-1}\})$. For example, if $Q = \{sad, sbd, sacd\}$, then $CountCopies(10, Q, \beta)$ computes $\beta_{sabcd} = [4, 2, 1, 1, 0]$ and the function returns 4, which is the value of β_s.

F. Message Forwarding

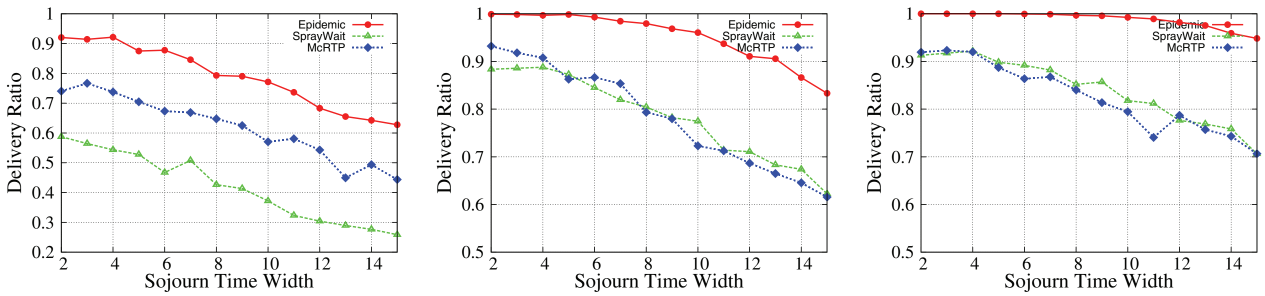
Once the β values are computed, node s sends β_j copies to each node j who is currently in contact with s, if $\beta_j > 0$. It does this by creating m', a copy of message m, and by setting field $m'_c = \beta_j$ before sending m' to a connected node j. The original copy counter m_c on node s is reduced accordingly, $m_c = m_c - \beta_j$. If $\beta_j = 0$, then node s does not forward the message to node j. If node s decides to forward β_j copies of m to node j and j already has m buffered with x copies, then node j updates the number of copies to $x + \beta_j$, and s reduces m_c to $m_c - \beta_j$, just as if the transmission went through but without actually transmitting m. This is done because better paths have been identified that pass through node j.

Algorithm 1 Candidate Paths Selection

Require: V, set of nodes; s, source node; T, current time; m_{path}, set of nodes already visited by message m.
Ensure: set Q with candidate paths on which to forward the message

set $\mathcal{R} = \emptyset$ {set with paths from s to destination d}
{for adding paths of length 2}
for all $i \in V \setminus m_{path} \setminus \{s, d\}$ **do**
  if $D(T, ttl(m), si) > 0$ **then**
    $R = [s, i, d]$
    $D_R = D(T, ttl(m), R)$
    if $D_R > 0$ **then**
      $\mathcal{R} = \mathcal{R} \cup \{R\}$
{for adding paths of length 3}
if $m_c > 2$ **then**
  for all $i \in V \setminus m_{path} \setminus \{s, d\}$ **do**
    if $D(T, ttl(m), si) > 0$ **then**
      for all $j \in V \setminus m_{path} \setminus \{s, d, i\}$ **do**
        $R = [s, i, j, d]$ {path from s to d of length 3}
        $D_R = D(T, ttl(m), R)$
        if $D_R > 0$ **then**
          $\mathcal{R} = \mathcal{R} \cup \{R\}$
sort paths R from $\mathcal{R}$ based on decreasing value D_R
create β vector with n elements
set $Q = \emptyset$ {set with selected paths from s to d}
$k = 1$ {index in sorted set $\mathcal{R}$}
while $k \leq |\mathcal{R}|$ **do**
  R_k = the k^{th} path from $\mathcal{R}$ sorted in decreasing order of D_R
  if CountCopies$(m_c, Q \cup \{R_k\}, \beta) \leq m_c - 1$ **then**
    $Q = Q \cup \{R_k\}$
  $k = k + 1$

We can evaluate the computational complexity of our algorithm as follows. Denote the runtime of computing $D(T, t, R)$ with $T(D_r)$, when path R has r edges. Assuming the contact profiles are known for all node pairs for time $1..ttl(m)$, use caching when computing $d(\cdot)$, the time complexity of $T(D_1) = O(ttl(m)^2)$, $T(D_2) = O(ttl(m)^3)$, and $T(D_3) = O(ttl(m)^3)$. In Algorithm 1, the first for loop is $O(nT(D_2))$, and the second top for loop is $O(n^2 T(D_3))$. Sorting set $\mathcal{R}$ re-



(a) Network with 10 nodes (b) Network with 20 nodes (c) Network with 30 nodes

Fig. 2. Comparison of delivery ratio in DTNs with different network sizes.

quires $O(n^2 \log(n))$ time. The while loop executes maximum $O(n^2)$ times. Since the protocol parameter $C \leq n$, then the runtime of function $CopyCount()$ is $O(n)$, and so the while loop's runtime is in $O(n^3)$. Thus, the algorithm runtime is given by $O(n^2 TTL^3 + n^2 \log(n) + n^3) = O(n^2 \cdot TTL + n^3))$.

IV. Performance Evaluation

A. Simulation Setup

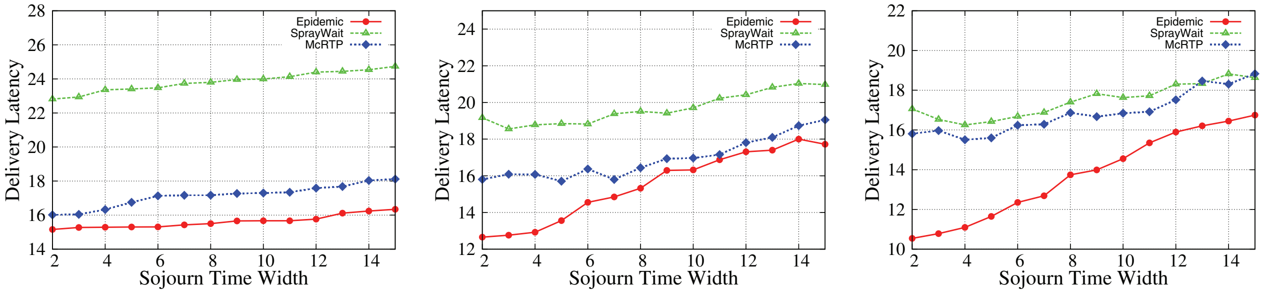
We developed a custom packet-based simulator implemented in Java to evaluate McRTP and compare its performance against the following protocols: Epidemic routing [12] ("*Epidemic*") and Spray-and-wait [10] ("*SprayWait*"). In Epidemic, a node would spread the message it has to any nodes it encounters. We implement this approach to investigate the optimal end-to-end delivery ratio between two nodes. In SprayWait, the system sprays a number of copies into the network, and then waits until one of these copies meets the destination. Discrete time is used in our simulation. Nodes are moving with a geo-community-based DTN scenario. Two nodes can communicate only when they are at the same geo-community. In our simulated DTNs, there are 10 geo-communities, and initially, n nodes are uniformly distributed among those geo-communities. Also, we assume that each node has a trajectory deviation probability p to indicate a mobility preference. Specifically, each node has a probability $1-p$ to head for a geo-community from where it is, and visits other geo-communities with probability $p/8$, individually. Here p is set as 10%. The travelling time between two geo-communities is uniformly distributed in $[2, 3]$. The sojourn time of a node at a specified geo-community is uniformly selected in $[w, w+2]$, where w varies from 2 to 15 time units. Note that larger w indicates nodes spend more time at geo-communities, and there would be less mobility in networks. We use the Poisson distribution to model the message generation in the network, with the average message arrival rate as 10 time units. For each message, we randomly select a pair of nodes as source and destination, respectively. The initial prediction window of McRTP is set to 35 time units, and packet TTL is 35 time units. Also, in our simulation we regulate that SprayWait sprays the same number of message copies as in McRTP, which is 5. We run the simulation for a "warm-up period" to reach a steady state, and collect sufficient mobility history information to generate P and $S_{ij}(k)$ for prediction. After that, the simulator runs for 2048 time units in each scenario for data collection, and we run each scenario 10 times to report the average. Considering that the average message arrival rate is 10, we believe that in 2,048 time units a sufficient number of messages can be generated for retrieving the stable average data tendency.

For performance evaluation, the following metrics are employed: 1) *Delivery ratio* is defined as the ratio of the number of successfully delivered messages to the total number of generated messages in the network; 2) *Delivery latency* is the average end-to-end delivery latency between a pair of source and destination in the network.

B. Results

The performance evaluation is conducted in three DTNs with different network sizes. The first one is a DTN with 10 nodes to represent a sparse network. The second one has 20 nodes, while the third one represents a crowded network with 30 nodes. Note that sparse networks have less contacts than crowded ones since they have fewer nodes.

We first investigate delivery ratio of the three protocols with different sojourn time in three simulated networks. Figure 2 plots the simulation results. We see that as sojourn time w increases, delivery ratio of three protocols declines in all of the three networks. This is because as nodes stays at a geo-community longer, there would be less mobility in networks, which reduces contact opportunity and consequently influences message delivery. Further, from figure 2(a), McRTP has much better delivery ratio than SprayWait in the sparse network with 10 nodes. The reason is that due to the prediction of contact probability and contact times, and the candidate paths selection mechanism, McRTP always picks the paths who have the highest possibility to reach destinations for routing, while SprayWait randomly spreads message copies in networks. But when the network size is enlarged, McRTP and SprayWait has similar delivery ratio, as shown in figure 2(b) and 2(c). Note that each node has a behavior deviation, which is indicated by the trajectory deviation probability p, per described in the simulation setup. As the number of nodes increases, the overall network behavior deviation is accumulated, which can heavily influence the prediction accuracy of McRTP. As a result, McRTP has a similar delivery ratio as



(a) Network with 10 nodes (b) Network with 20 nodes (c) Network with 30 nodes

Fig. 3. Comparison of delivery latency in DTNs with different network sizes.

SprayWait. Besides that, as network size arises, there are more contact opportunities, and more suitable nodes could be used as forwarders. Therefore, delivery ratio in the network with 30 nodes (crowded network) is larger than the one with 10 nodes (sparse network), as shown in figure 2(a) and 2(c).

Then, we discuss the delivery latency difference. Figure 3 compares the corresponding delivery latency of the three protocols in the simulated DTNs. It shows that when sojourn time w goes up, delivery latency of three protocols is raised in all three networks. The reason is that DTNs rely on nodes mobility for message delivery, but larger w brings less mobility, which results in increased delivery latency. Also, figure 3(a) presents McRTP has an obvious advantage over SprayWait on delivery latency in the network with 10 nodes (sparse network). This is because with the contact prediction model and the path evaluation mechanism, McRTP can better choose the relay nodes to forward messages. However, we also see that this advantage looks less obvious in figure 3(b) and 3(c). Especially in figure 3(c), McRTP even has the same delivery latency as SprayWait. As we mentioned before, when network size rises, network behavior deviation is also enlarged, which disturbs the prediction accuracy of our model. Thus, the advantage of McRTP on delivery latency turns smaller when network size is increasing. Furthermore, since there are more contact opportunities in crowded networks than in sparse networks, it makes relay nodes encounter destinations relatively easier. Thus, delivery latency in figure 3(c) is smaller than that in figure 3(a) and 3(b).

In summary, we see that McRTP outperforms traditional SprayWait, especially in sparse networks. This is because in sparse networks, where there are less contact opportunities, nodes have to rely on contact prediction for routing.

V. Conclusion

In this paper, we propose McRTP, a multi-copy routing protocol with trajectory prediction for social DTNs, where nodes follow scheduled trajectories among a set of geo-communities with small deviations. McRTP evaluates paths based on the prediction of future contact probability and contact times for routing. Also, we develop a copy count assignment mechanism constraining the number of message copies. Simulation results prove McRTP's performance, especially in sparse networks.

Acknowledgment

This research was supported in part by the National Natural Science Foundation of China under Grant No. 61272451.

References

[1] C. Boldrini, M. Conti, and A. Passarella. Design and performance evaluation of contentplace, a social-aware data dissemination system for opportunistic networks. *Comput. Netw.*, 54(4), 2010.

[2] E. Bulut, Z. Wang, and B. K. Szymanski. Impact of social networks on delay tolerant routing. In *Proc. of IEEE GLOBECOM*, 2009.

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

[4] B. Burns, O. Brock, and B. N. Levine. Mv routing and capacity building in disruption tolerant networks. In *Proc. of IEEE INFOCOM*, 2005.

[5] W. Gao, Q. Li, and G. Cao. Forwarding redundancy in opportunistic mobile networks: Investigation and elimination. In *Proc. of IEEE INFOCOM*, 2014.

[6] Z. Guo, B. Wang, and J. Cui. Prediction assisted single-copy routing in underwater delay tolerant networks. In *Proc. of IEEE GLOBECOM*, 2010.

[7] Y. Han, H. Wu, Z. Yang, and D. Li. Delay-constrained single-copy multi-path data transmission in mobile opportunistic networks. In *Proc. of IEEE SECON*, 2014.

[8] J. LeBrun, C. Chuah, and D. Ghosal. Knowledge based opportunistic forwarding in vehicular wireless ad hoc networks. In *Proc. of VTC Spring*, 2005.

[9] A. Lindgren, A. Doria, and O. Schelen. Probabilistic routing in intermittently connected networks. In *SIGMOBILE Mobile Computing and Communications Review*, 2003.

[10] T. Spyropoulos, K. Psounis, and C. Raghavendra. Spray and wait: An efficient routing scheme for intermittently connected mobile networks. In *Proc. of WDTN*, 2005.

[11] T. Spyropoulos, K. Psounis, and C. S. Raghavendra. Efficient routing in intermittently connected mobile networks: the single-copy case. In *IEEE/ACM Transactions on Networking*, 2008.

[12] A. Vahdat and D. Becker. Epidemic routing for partially-connected ad hoc networks. Technical report, Duke University, 2000.

[13] J. Wu and Y. Wang. Social feature-based multi-path routing in delay tolerant networks. In *Proc. of IEEE INFOCOM*, 2012.

[14] L. Xue, D. Kim, Y. Zhu, D. Li, W. Wang, and A.O. Tokuta. Multiple heterogeneous data ferry trajectory planning in wireless sensor networks. In *Proc. of IEEE INFOCOM*, 2014.

[15] Q. Yuan, I. Cardei, and J. Wu. An efficient prediction-based routing in disruption-tolerant networks. *IEEE Transactions on Parallel and Distributed Systems*, 23(1):19–31, 2012.

[16] H. Yue, H.L. Fu, L. Guo, Y. Fang, and P. Lin. An efficient prediction-based routing protocol in delay tolerant networks. In *Proc. of IEEE GLOBECOM*, 2013.

[17] Y. Zhang and J. Zhao. Social network analysis on data diffusion in delay tolerant networks. In *Proc. of ACM MobiHoc*, 2009.

[18] X. Zhuo, Q. Li, G. Cao, Y. Dai, B. Szymanski, and T. La Porta. Social-based cooperative caching in dtns: A contact duration aware approach. In *Proc. of IEEE MASS*, 2011.