

A Storage-Friendly Routing Scheme in Intermittently Connected Mobile Network

Hao Wen, Fengyuan Ren, Jia Liu, Chuang Lin, Pan Li, and Yuguang Fang, *Fellow, IEEE*

Abstract—Considering the constraints brought by mobility and resources, it is important for routing protocols to efficiently deliver data in intermittently connected mobile networks. Different from previous works that use the knowledge of previous encounters to predict future contacts, we propose a storage-friendly region-based protocol, i.e., RENA, in this paper. Instead of using temporal information, RENA builds routing tables based on regional movement history, which avoids excessive storage for tracking encounter history. We validate the generality of RENA through a time-variant community mobility model with parameters extracted from the Massachusetts Institute of Technology (MIT) Wireless Local Area Network (WLAN) trace and a vehicular network based on eight bus routes in the city of Helsinki, Finland. The comprehensive simulation results show that RENA is not only storage friendly but also more efficient than epidemic routing, the restricted replication protocol spray and wait (SNW), and the encounter-based protocol resource allocation protocol for intentional delay-tolerant networks under various conditions.

Index Terms—Mobile computing, routing.

I. INTRODUCTION

An intermittently connected mobile network (ICMN) is a delay-tolerant ad hoc network that is made up of mobile nodes. Typical examples include ZebraNet [1], the vehicular ad hoc network (VANET) [2], and the pocket switched network (PSN) [3]. In these networks, there is usually no supporting infrastructure (such as mesh nodes or base stations) for routing. Due to frequent network disruption and intermittent connections in ICMN, it is difficult to establish reliable end-to-end paths between mobile peers. Instead, the approach of delay-tolerant networking (DTN) [4], which builds upon carry and

forward between mobile nodes, is widely applied to enable communication in such challenging mobile environments. Considering the constraint brought by mobility and limited resources in ICMN, particularly for resource-constrained devices such as sensor nodes, it is important for routing protocols to efficiently deliver data. Typical sensor nodes have very limited storage, such as sensor Mica2 (program flash 128 kB + serial flash 512 kB) and sensor TelosB (program flash 48 kB + serial flash 1024 kB).

Previous work [5] suggests the benefit of gaining more knowledge about network conditions for routing decisions. Most DTN routing protocols use knowledge of previous encounters to forecast future contacts [1], [2], [6], [7]. In these works, encounter history, which is a kind of *temporal* information, is taken as guidance to predict mobility; every node records other users when they encounter each other. For example, the powerful resource allocation protocol for intentional delay-tolerant networks (RAPID) protocol [2] even needs to record the history of every forwarding operation to provide a more precise prediction. However, it requires excessive storage to track encounter history, which makes RAPID suffer from degraded performance in storage-limited conditions.

Aside from temporal information, it is also observed that the long-run trends in mobility widely reveal some spatial-related characteristics [8], [9]. Several wireless local area network (WLAN) traces indicate that the mobility of users demonstrates *location preference* to some extent; users usually visit only parts of locations, such as restaurants or office buildings, and stay there for long periods. Meanwhile, they will commute back and forth among these locations, resulting in *periodic reappearance*. These traces also indicate *heterogeneity* in mobility, e.g., some nodes visit popular locations more often than other locations.

Although some mobility models are derived from these observations, there is little work to make use of spatial properties for efficient routing in ICMN. Because it is not cost effective to build routing tables based on encounter history in highly opportunistic mobile networks, one reasonable approach is to maximize the benefit of *macro-level* spatial information without having to constantly track and update encounter information.

In this paper, we propose a storage-friendly REgion-bAsed protocol, i.e., RENA, in ICMN. Instead of using encounter information, RENA builds routing tables based on regional movement history, which avoids excessive storage for recording encounter history. More specifically, this paper makes the following contributions.

- 1) We introduce a novel concept of *region-related* pattern extracted from the spatial property and take advantage of

Manuscript received December 11, 2009; revised March 26, 2010, May 10, 2010, and October 11, 2010; accepted December 5, 2010. Date of publication January 6, 2011; date of current version March 21, 2011. This work was supported in part by the National Natural Science Foundation of China under Grant 60773138, Grant 60834004, Grant 60971102, and Grant 60932003, by the National Grand Fundamental Research Program of China973 under Grant 2009CB320504 and Grant 2010CB328105, and by the National Science and Technology Major Project of China under Grant 2009ZX03006-001-003 and Grant 2009ZX03006-003-01. The review of this paper was coordinated by Prof. A. Boukerche.

H. Wen, F. Ren, J. Liu, and C. Lin are with the Department of Computer Science and Technology, Tsinghua University, Beijing 100084, China (e-mail: wenhao@csnet1.cs.tsinghua.edu.cn; renfy@csnet1.cs.tsinghua.edu.cn; jliu@csnet1.cs.tsinghua.edu.cn; clin@csnet1.cs.tsinghua.edu.cn).

P. Li is with the Department of Electrical and Computer Engineering, Mississippi State University, Starkville, MS 39762 USA (e-mail: li@ece.msstate.edu).

Y. Fang is with the University of Florida, Gainesville, FL 32611-6130 USA (e-mail: fang@ece.ufl.edu).

Color versions of one or more of the figures in this paper are available online at <http://ieeexplore.ieee.org>.

Digital Object Identifier 10.1109/TVT.2011.2104378

this macromobility information. To the best of our knowledge, RENA is the first region-based routing protocol proposed in delay-tolerant ICMN under realistic resource constraints.

- 2) Considering the tradeoff between the control packets and the storage consumption, we emphasize the necessity to design a *storage-friendly* protocol. Our experiments demonstrate that the delivery performance of RAPID [2] will significantly be degraded in resource-constrained conditions if we consider storage occupied by control packets, whereas RENA achieves more stable performance, because it only requires fixed storage for control packets.
- 3) We validate the generality of RENA through the following two realistic simulation scenarios: 1) the time-variant community (TVC) mobility model, whose parameters are extracted from the Massachusetts Institute of Technology (MIT) WLAN trace and 2) a vehicular network based on eight bus lines of the city of Helsinki, Finland.

The rest of this paper is organized as follows. We present related work in Section II. A region-based mobility pattern is proposed in Section III. Section IV describes the design details of the region-based routing protocol RENA. Simulation results are presented in Section V. Finally, we discuss the feasibility of RENA and conclude this paper.

II. RELATED WORK

Numerous DTN routing and forwarding schemes have been proposed over the last few years. The primary difference between various DTN routing protocols is whether the future topology is deterministic [10]. Because we mainly focus on routing in a stochastic mobile network without supporting infrastructure, whose topology is time evolving and, thus, difficult to predict, our problem is different from a deterministic network or the category of controlling node movements.

Epidemic routing is a simple and fast way to perform routing throughout the network [11]. However, it is extremely wasteful of resources, such as wireless bandwidth and storage space. One approach for reducing the overhead of epidemic routing is to distribute a bounded number of copies [12], [13]. Because this approach does not make use of gaining knowledge about network conditions, its performance is not satisfactory under more realistic mobility conditions.

Previous work [5] suggests the benefit of gaining more historical knowledge to estimate the forwarding probability. Protocols in this category are referred to as history- or estimation-based forwarding [10]. Because node contact/encounter patterns represent the temporal characteristics of the node mobility, several DTN routing protocols exploit the past knowledge of encounters to optimize the delivery performance. They are differentiated from each other mainly on the following two decision methods: 1) one-hop information based [1], [14], [15] or 2) end-to-end information based [2], [5], [7]. Scheduling strategies and queue management strategies are also discussed as part of the protocol design.

However, these protocols face the fundamental challenge of choosing the right time scale for estimation [16]. The estimation

value quickly becomes a poorer indicator as distance increases, which is referred to as the “distance effect” [6]. Therefore, the long-run trends inherited in the mobility, such as spatial features, are difficult to capture from a short-scale temporal view. Even for the powerful RAPID [2], which requires almost all contact history for estimation, its performance significantly deteriorates due to the expansion of historical information.

To empirically understand mobility, there are several recent works on WLAN measurements that reveal the important spatial properties of real-world wireless network users [8], [9]. Although these research works are mainly obtained from the deployed wireless access networks with fixed infrastructure, they imply that spatial properties should be taken into account during the design of practical routing protocols.

MobySpace is, perhaps, the first routing protocol to make use of locations of landmarks (e.g., access point) to construct a virtual space [17]. Here, the mobility pattern of a node can be described by its visits to locations. However, this protocol mainly focuses on power-law probability distributions and assumes that nodes have full knowledge of mobility patterns and infinite resources, which is impractical in realistic applications.

Recently, several social-based routing protocols have been proposed [3], [18], [19]. In essence, temporal or spatial information is a kind of representation of the social context. BUBBLE, which is proposed in PSN, calculates node popularity and social community based on *encounter history* [3]. It evaluates the impact of community and centrality on forwarding and proposes a hybrid algorithm that selects high centrality nodes and community members of destination as relays.

In profile-casting routing in [18], the authors use the *mobility profile* to design a protocol based on the similarity in user mobility. The mobility profile here is related to the fraction of time the user spends at reference locations during every time slot. However, this profile-casting protocol only propagates message *within a group of similar users*, which makes the problem different from what we discuss in this paper. In addition, the authors have not discussed the storage constraint when recording the mobility profile for a large number of time slots.

The region-based solution presented in this paper is similar to the hub-based sociological-orbit-aware location approximation and routing (SOLAR) protocol [19]. As opposed to our proposed RENA, SOLAR assumes partially deterministic mobility of each user in a more densely deployed MANET, where packet transmission is based on greedy geographic forwarding. In greedy geographic forwarding, messages are delivered based on the current locations of nodes, and each relay node chooses its next hop from among its neighbors closest to the destination. However, in an ICMN that suffers from frequent network disruption and intermittent connections, the specific location of a destination node is difficult to obtain, and nodes encounter each other very occasionally and asynchronously such that there are no stable relay neighbors.

A preliminary version of RENA has been presented in [20]. In this journal version, we provide more details about the region-based pattern in Section III and the protocol design in Section IV. Extensive simulations results and complexity

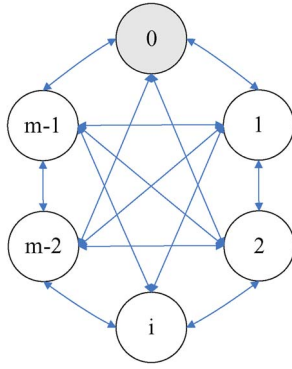


Fig. 1. m regions as the mobility domain.

analysis are given in Section V for better comparison. The part of related work is also enhanced.

III. REGION-BASED MOBILITY PATTERN

Our solution is motivated from a simple observation: location-preference, reappearance, and heterogeneity are usually observed as typical features not only from human beings but from other species (such as zebras) as well [1], [8], [9]. People or animals usually visit parts of locations, stay there for a period of time, and then commute back and forth. The encounter history is effective in predicting the next encounter in a specific area. However, when users change their geographic areas, the encounter information obtained at previous locations may no longer be accurate as the guidance for new locations.

For example, students who attend classes in the same building are likely to frequently encounter each other at study time. After class, they may go to different dormitories. It is difficult to estimate who will again encounter who at new locations based only on encounter history in the building, because two students who attend the same class do not necessarily live in the same dormitory. Although the statistical tracking information at a micro level (i.e., on small time scales or within small distances) may still be useful, it is necessary to take the macro-level information into account in a realistic scenario.

From a social context, both visited locations and people's encounters have a strong connection with affiliation and lifestyle. In contrast with encounter-based protocols, region-based patterns offer us a different way of solving the routing problem in ICMN.

A. Mobility Model

We consider m regions as the mobility domain (see Fig. 1), which is enumerated as $0, 1, m-2, m-1$. A node (we interchangeably use the terms "node," "user," and "peer" to describe a mobile network user in this paper) moves according to a transition matrix \mathbf{P} . It moves from region i to another region j with probability p_{ij} or remain in the same region with probability p_{ii} . The steady-state probability at region i is π_i . Then, this mobility model can be thought of as a Markov model where a possible state transition is made every τ unit time of period.

The model mainly focuses on the transition probability between regions i and j , which demonstrates spatial affiliations

and social contexts of nodes. Because nodes in an ICMN move in a distributed way, we do not ask every node to act in a synchronous fashion or with the same transition probability. Every one unit time, every node will track its own region and create its own record based on its affiliation and lifestyle.

Note that this Markov mobility model is only a simplified abstraction of real mobility, and users may not strictly follow the Markov property in real scenarios. Our model differs from several widely used mobility models, e.g., Manhattan street mobility, random waypoint mobility [21] and community-based mobility [22], in that our approach is mainly used for micro-level abstraction, whereas other approaches have specific movement rules and can directly be used in simulation. As long as there exists location preference, the region-based idea should be feasible. Our simulation will show that, even in fully random-based mobility, our protocol still achieves satisfactory performance.

After proposing the region-based mobility model to demonstrate the macro-level abstraction, we further go deep into this model and discuss several mobility characteristic parameters.

B. Staying Time

First, we could easily get the average continuously staying time in the same region i as

$$ST_i = \sum_{k=1}^{+\infty} k(1 - p_{ii})p_{ii}^{k-1} = \frac{1}{1 - p_{ii}}. \quad (1)$$

This formula indicates that when a node has higher probability p_{ii} to remain in the same region i , its average continuous staying time is longer, particularly when $p_{ii} = 1$, and ST_i is equal to infinity.

C. Hitting Time

Then, we consider the expected hitting time HT_{ij} of a node that arrives at region j from region i for the first time. This parameter is useful as a metric for evaluating the routing delay of different relay nodes.

To obtain this parameter, we make state j an absorbing state and other states transient states and then reset the transition probabilities from state j as

$$p_{jj} = 1 \quad p_{ji} = 0 (i \neq j). \quad (2)$$

Now, we can extract the matrix \mathbf{P}_T that specifies only the transition probabilities from transient states into transient states. Given the starting region i , we define s_{ik} to be the expected number of time periods that the Markov chain is in region k . Based on the relationship between transient states, we have $\mathbf{S} = \mathbf{I} + \mathbf{P}_T \mathbf{S}$, where \mathbf{I} is the identity matrix. Then, the matrix \mathbf{S} can be calculated as

$$\mathbf{S} = (\mathbf{I} - \mathbf{P}_T)^{-1}. \quad (3)$$

Table I shows an example where region c is taken as the destination.

TABLE I
EXAMPLE OF \mathbf{P} , \mathbf{P}_T , AND \mathbf{S}

Region	a	b	c
a	0.75	0.25	0
b	0	0.5	0.5
c	0.5	0	0.5

Region	a	b
a	0.75	0.25
b	0	0.5

Region	a	b
a	4	2
b	0	2

Because the expected time to enter absorbing state j from transient state i is equal to the expected time spent in all transient states before the node enters state j from transient state i , we can get HT_{ij} as

$$HT_{ij} = E(ht_{ij} | \text{start from } i) = \sum_{k \neq j} s_{ik}. \quad (4)$$

In the case of Table I, $HT_{ac} = 6$ and $HT_{bc} = 2$ time periods.

D. Return Time

Finally, we further propose another important metric: the expected return time RT of a node back to a region j after it leaves region j . This parameter is regarded as the time interval for a node to revisit the same region, which will be used in our protocol design.

In fact, RT_j is equal to the expected hitting time to region j from all j 's adjacent regions, i.e.,

$$RT_j = \sum_{k \neq j} \frac{p_{jk}}{1 - p_{jj}} HT_{kj} \quad (5)$$

where $1 - p_{jj} = \sum_{k \neq j} p_{jk}$. In the case of Table I, $RT_j = (0.5/0.5 + 0) \cdot 6 + (0/0.5 + 0) \cdot 2 = 6$ time periods.

IV. PROTOCOL DESIGN

In this section, we will present the design details for our region-based protocol RENA. In the following discussion, we assume that nodes are equipped with either the Global Positioning System (GPS) or other localization methods to obtain a geographic location.

A. Protocol Overview

The main idea of RENA is to find the popular regions for a destination node and distribute copies of packets inside those popular regions. However, we first need to define a region. In this paper, we simply adopt a square coverage in the movement area as a region. This way, the whole area is divided into m equal-sized grid cells, as shown in Fig. 2 ($m = 16$).

Given source node x and destination node y , RENA chooses candidate regions for y in terms of estimated delivery delay $D_{ij}(x, y)$, which includes the routing time $R_{ij}(x)$ from the current region i of node x to the candidate region j and the

x : source node y : destination node z_1 - z_3 : relay nodes

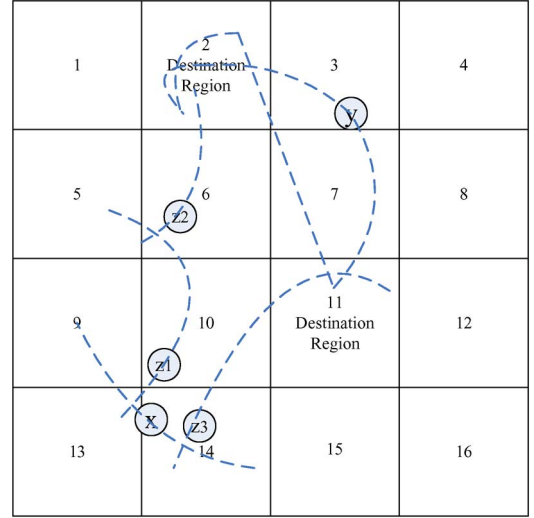


Fig. 2. RENA overview.

waiting time $W_j(y)$ of the relay node before encountering node y inside the region j . We have

$$D_{ij}(x, y) = R_{ij}(x) + W_j(y) \quad (6)$$

where $R_{ij}(x)$ and $W_j(y)$ are both derived from the transition matrix.

In this paper, we choose delivery delay, not encounter probability, as an optimized metric. The reason for this choice is that most existing probability-driven approaches have an incidental rather than an intentional effect on delay-related metrics [2]. For example, in Fig. 2, the selected region 2 is the most popular for node y but is far from the current location of source node x (i.e., region 14), whereas region 11 is not frequently visited by node y but may still be a feasible choice in terms of shorter routing time $R_{ij}(x)$.

When L regions are chosen as the candidate destinations (see Section IV-C), the source node will send one copy to every chosen region, respectively. The relay node is chosen according to its expected hitting time HT_{kj} from the current region k to the destination region j ; the node with a smaller hitting time to the destination region j would be a better relay node. Thus, in Fig. 2, when relay node z_1 determines that z_2 has a smaller hitting time to region 2, z_2 is chosen as a new relay node, and z_1 will delete the copy.

After the relay node reaches one of the destination regions, e.g., z_3 reaches region 11 in Fig. 2, it will trigger a spray-and-search (SAS) distribution. First, the relay node will start a binary spray to $W - 1$ nodes in the spray phase. Then, in the search phase, W nodes that carry copies will only forward their own copies to nodes with a smaller return time RT_j until encountering the destination node.

Essentially, RENA replicates a limited number of copies to L destination regions and distributes copies inside the chosen regions. Because the replication strategy and the storage-management policy are carefully designed based on spatial properties, RENA is a storage-friendly protocol that limits

TABLE II
REGION TRANSITION MATRIX FOR 3-2-2-2

Region	2	3
2	2	0
3	1	0

TABLE III
P FOR NODE y IN FIG. 2

Region	2	3	7	11
2	0.67	0	0.33	0
3	1	0	0	0
7	0	0.5	0	0.5
11	0	0	1	0

replication only inside selected regions instead of the whole area.

B. Region Transition Matrix

Because we assume that every node has a geographic location, it can obtain the current region where it is located. Every τ unit time, it will track the current region j and record into region transition matrix \mathbf{B} , where b_{ij} indicates transfer times from the last region i to the current region j . In Fig. 2, if node y tracks that its located region changes from 3 to 2, b_{32} will add 1. For example, if the tracking history of node y is 3-2-2-2, its region transition matrix is shown as in Table II.

According to region transition matrix \mathbf{B} , we can obtain transition probability matrix \mathbf{P} as

$$p_{ij} = b_{ij} / \sum_{j=1}^m b_{ij}. \quad (7)$$

For example, because the tracking history of node y is 3-2-2-2-7-11-7-3 in Fig. 2, its corresponding transition probability matrix \mathbf{P} is shown as in Table III.

Then, according to the calculation in Section III, we could obtain the hitting time HT_{ij} of node y from region i to region j and its return time RT_j to region j . This way, nodes will flood their return time matrix \mathbf{RT} in the whole network. Because RT is only related to the number of regions m , this flooding transmission only consumes limited storage and bandwidth. Unlike RT , only hitting-time values HT related to the current region will be exchanged between encounter nodes.

C. Choosing Destination Regions

When source node x chooses candidate regions for destination node y , it is better to find proper destination nodes that not only are frequently visited by y but could quickly be reached from the current location of x as well. However, without global network knowledge, it is almost impossible to give an accurate prediction in a delay-tolerant mobile network. Instead, RENA operates based on an efficient estimation of routing time and waiting time.

In terms of the routing time $R_{ij}(x)$ from the current region i of node x to the candidate region j , the source node builds a *local view* from its historical encounter nodes and itself, i.e.,

$$R_{ij}(x) = \frac{HT_{ij}(x) + \sum_{i=1}^{en} HT_{ij}(z_i)}{en + 1} \quad (8)$$

where node z_i indicates the i th encounter node of x whose HT_{ij} is not equal to infinity, and en means encounter times between x and other nodes. Here, the same encounter node may repeatedly be calculated, i.e., $z_i = z_j$ when $i \neq j$, because we argue that frequently encountered nodes should have greater weight than the local view of a source node.

In terms of the waiting time before encountering the destination node inside the region j , we simply take the return time of the destination node as an estimation value, i.e.,

$$W_j(y) = RT_j(y). \quad (9)$$

Based on (8) and (9), source node x can obtain the estimated delay to the destination node y as

$$D_{ij}(x, y) = R_{ij}(x) + W_j(y). \quad (10)$$

Then, we arrange the estimated delays of all regions in ascending order as

$$D'_{i1}(x, y) \leq \dots \leq D'_{ik}(x, y) \leq \dots \leq D'_{im}(x, y). \quad (11)$$

To avoid unnecessary storage consumption and propagation overhead, we choose L destination regions or copies based on the time to live (TTL) for packets and simply assume that they are exponentially distributed. Then, the minimum L are selected according to

$$\left(\sum_{k=1}^L \frac{1}{D'_{ik}(x, y)} \right)^{-1} \leq \text{TTL}. \quad (12)$$

In (12), we have used the following result. When the encounter times between destination nodes and different L copies are exponentially distributed with means $1/\lambda_1, 1/\lambda_2, \dots, 1/\lambda_L$, respectively, the expected minimum delivery delay that the destination node encounters one of the copies is

$$\frac{1}{\lambda} = \frac{1}{\lambda_1 + \dots + \lambda_L} = \left(\sum_{k=1}^L \lambda_k \right)^{-1}. \quad (13)$$

Note that the exponential distribution of intercontact times is a widely accepted assumption in the delay-tolerant ICMN [8], [9]. Although the distribution in realistic motilities may be more complicated [23], we assume an exponential distribution of delivery time in this paper. Moreover, simulation results show that this tractable simplification performs well in different realistic scenarios.

D. Choosing Relay Nodes

After L destination regions are decided, the next step is to find relay nodes to forward these L copies to chosen regions. Relay nodes are chosen according to the expected hitting time HT_{kj} from the current region k to the destination region j as follows.

- 1) When two nodes encounter each other in the region k ($k \neq j$), they will exchange and compare their hitting-time values HT_{kj} .
- 2) Copies will be *forwarded* to the node with a smaller HT_{kj} . The other node with a bigger HT_{kj} *no longer keeps* the copy.

E. SAS

When the relay node z that carries copies reaches the destination region j , it will trigger a SAS distribution to improve the encounter probability with destination node y as follows.

- 1) *Spray phase*. The relay node z will start a binary replication of W copies to distinct relay nodes. When any node z_1 with $c > 1$ copies encounters another node z_2 without those copies, z_1 hands over to z_2 ($c/2$) copies and keeps half of these copies. This process will continue until $c = 1$ or until destination node y is encountered.
- 2) *Search phase*. When any one of W nodes z_3 that carry one copy encounters another node z_4 without that copy, it will forward its own copy to z_4 if z_4 has a smaller return time RT_j .

Here, we limit the number of copies inside regions W to control the total number of copies $L \times W$. The spray phase aims at “jumpstart” spreading in a quick manner, and the search phase works to find better relays that are more likely to stay inside region j . Note that SAS only performs in the destination region j to avoid burdening propagation in the whole area. Therefore, when any relay node in the process moves out of region j , its SAS will be suspended. SAS will resume when related copies are relayed back (maybe by new relay nodes) to region j .

F. Updating Policy

Because it is impractical to synchronously obtain the global system state, RENA chooses to exploit the value of delayed estimation. Although it is helpful for routing decisions with more knowledge of network conditions, the routing protocol should consider the tradeoff between performance gain and system overhead and thereby update the control information in an efficient manner. RENA achieves this goal by propagating only necessary control packets.

In RENA, every node updates its region transition matrix every τ unit time but only updates its transition probability matrix, hitting-time matrix, and return time matrix when encounter events happen. For efficiency, a node z_1 executes the following operations when encountering a peer z_2 in region k .

- 1) Directly forward data destined to z_2 .

TABLE IV
COMPLEXITY ANALYSIS

Protocol	Epidemic	SNW	RENA	RAPID
Computation (routing)	$O(1)$	$O(1)$	$O(m)$	$O(n)$
Space (control packets)	$O(1)$	$O(1)$	$O(m^2 + nm)$	$O(n^2 + nF(t, n))$

- 2) Exchange acknowledgments (ACKs) of delivery packets that are not beyond TTL, and delete useless copies according to ACKs.
- 3) Exchange the return time matrix **RT** of its historical encounter nodes and itself.
- 4) Exchange the hitting-time values HT_{kj} from the current region k to other regions.
- 5) Forward copies to z_2 if z_2 is a better relay node, as discussed in Section IV-D.
- 6) Execute the SAS distribution if necessary.

Here, ACKs are used to avoid unnecessary overhead, and the information of the third and fourth steps are the decision-making bases for the last two steps. Although the updating information may be stale in such a dynamic network, this distributed policy proves to be efficient through comprehensive simulations.

G. Storage Management

For the multicopy replication mechanism, it is crucial to design a storage policy to ensure that the total storage consumption is small and controlled. RENA generates only a small carefully chosen number of LW copies (every W copies distributed in one selected region). Aside from data packets and ACKs, each node in RENA mainly records a return time matrix **RT** of all users and its own hitting-time matrix **HT**.

In addition, we manage packets according to their redundancy and importance degree as follows.

- 1) Immediately delete acknowledged packets.
- 2) Every τ unit time, remove packets and ACKs that have reached the threshold of TTL.
- 3) If the storage is close to an overflow state, drop copies in the search phase first, followed by copies in the spray phase. Further drop the copies that have not reached the destination region, if still necessary.

V. EVALUATION

In this section, we conduct a simulation study to evaluate our RENA and other protocols.

A. Protocol Comparisons

We compare RENA to the following three other routing protocols: 1) epidemic routing [11]; 2) spray and wait (SNW) [12]; and 3) RAPID [2]. A brief review of each algorithm is given as follows, and their complexity analysis is listed in Table IV.

For epidemic routing [11], whenever a node encounters another node, the two nodes exchange all packets that they do not have in common. In such an epidemic way, all messages are eventually spread to all nodes.

TABLE V
MAIN PARAMETERS OF TWO SCENARIOS

Simulation time	12 hours + 1 hour warmup
TTL	30-180 minutes (default 180)
Storage	50-300 KB (default 100)
Packet generating interval	20-50 (default 30)
Number of nodes (TVC)	10-100 (default 50)
Number of nodes (Bus)	8-80 (default 40)

SNW [12] distributes only a limited number of copies each to a different relay. Each copy is then carried all the way to the destination node by the designated relay. It restricts the number of replications to be equal to about 10%–15% of all n nodes.

RAPID [2] treats DTN routing as a resource allocation problem that translates the routing metric into per-packet utilities that determine how packets should be replicated. To estimate the utility given the routing metric, the control packets for a node include expected meeting times between nodes $O(n^2)$, the average size of past transfer between nodes $O(n^2)$, and carriers of those undelivered replicas $O(nF(t, n))$. Here, we define the number of undelivered replicas as a simplified function of time t and n . As far as we know, results in [2] show that RAPID is one of the most powerful protocols in terms of both delivery rate and delivery delay.

As discussed in Section IV, RENA chooses L regions based on estimated delays in ascending order $O(m)$. The control packets for a node include its transition probabilities \mathbf{P} between regions $O(m^2)$, its hitting-time matrix $O(m^2)$, and the return time of all nodes $O(nm)$.

To evaluate the effect of control packets, we also provide the results if we only count the storage consumed by data packets and set an *unlimited buffer* for control packets. These additional results are only used to *verify the storage friendliness* of control packets, because the assumption of an unlimited buffer is not practical in reality.

B. Experimental Setup

We implement four routing protocols using the opportunistic network environment (ONE) simulator [24]. We compare our protocol with other routing algorithms in terms of delivery rate and delay with variations of TTL, storage capacity, traffic load, and the number of nodes.

For connectivity, we assume interpersonal communication (10-m range and 2 Mb/s) between mobile users using Bluetooth devices, such as mobile phones. A simple channel model is adopted, where the transmission coverage is circular, and interference may occur only from immediate neighbors. When two nodes move within the communication range, they establish a connection and transmit to each other using a link-level ACK.

Every t interval time, one mobile user generates one packet of 1 kB to a random destination. For fairness of comparison, the numbers of copies of RENA and SNW are both restricted to 10% of all n nodes. All nodes in RENA record the current region every 5 s. The main simulation parameters are listed in Table V, and the default values are adopted, if not mentioned. Other parameters are given in the specific mobility model.

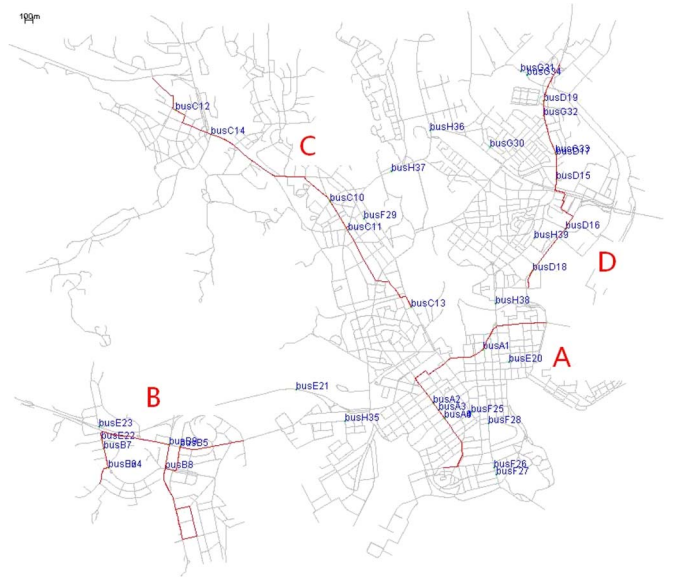


Fig. 3. Bus routes in Helsinki (A–D).

C. Scenario 1: TVC Mobility Model

The TVC mobility model is proposed to capture spatial and temporal correlations of realistic mobility [9]. It consists of two types of time periods—the normal movement period (NMP) and the concentration movement period (CMP)—and defines a single community for each time period. In NMP or CMP, a node has the following two different modes of movement: 1) a local epoch and 2) a roaming epoch. At the end of each epoch, the node picks the movement mode for the next epoch according to a two-state Markov model with probability p_r (from a local to a roaming epoch) and probability p_l (from a roaming to a local epoch). In a local epoch, the mobility of the node is confined within its community. In a roaming epoch, the node is free to move in the whole area.

To create a more realistic scenario, we adopt the *trace-driven* TVC model, which is referred to as model 1 in [9], that displays similar behavior to the MIT WLAN trace in terms of the hitting and encountering times. In this model, the simulation area is $1000 \times 1000 \text{ m}^2$, the community size is $100 \times 100 \text{ m}^2$, and the edge length of a region is chosen to be 250 m. This way, the number of regions is 16.

D. Scenario 2: Bus Mobility Model

We also choose a typical vehicular DTN based on the city area of Helsinki. The whole city area has four main districts A–D. In addition, three overlapping districts are defined to simulate movements between the center A and other districts, and one district covers the whole simulation area, i.e., E includes A and B, F includes A and C, G includes A and D, and H covers from A to D. Every district is assigned its own bus route, respectively (see routes A–H in Figs. 3 and 4).

A total of n buses are evenly distributed among eight bus routes and travel across these predefined bus routes according to a periodic schedule. With a small communication range, nodes communicate with each other when buses get closer. RENA generates 16 regions with unit size $2500 \times 2500 \text{ m}^2$. Other parameters are listed in Table VI.

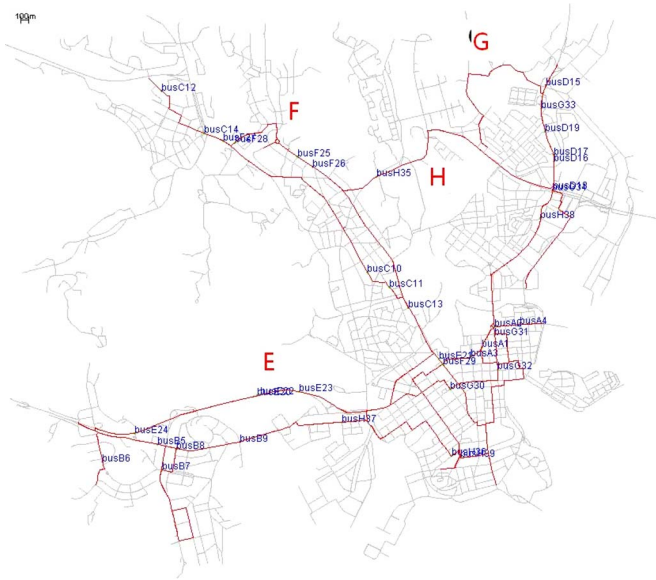


Fig. 4. Bus routes in Helsinki (E–H).

TABLE VI
PARAMETERS OF THE BUS MODEL

Simulation area	10000 × 10000m ²
Min and max of speed (uniformly)	$v_{min} = 7, v_{max} = 10$
Min and max pause time at bus stop	$T_{min} = 10, T_{max} = 30$
The number of bus stops of routes A–H	11,16,11,6,13,15,17,20

In this part, we are mainly concerned with resource-constrained devices carried by mobile users and not built-in vehicles. We imagine that people who carry the wireless devices may move in not only a small-scale scenario such as a campus area but a large-scale mobile network, such as a vehicular network, as well. Fairness is another big reason that we chose a bus network, because we want to fully compare RAPID.

E. Results

To avoid transient effects, the warm-up time is set to 1 h, and the actual simulation time is set to as long as 12 h. All results are averaged over 30 runs with different seeds and shown with a 95% confidence interval.

1) *Impact of TTL*: Fig. 5 shows the delivery rate of each protocol as TTL for data packets increases from 30 to 180 min in scenario 1. With this increase in TTL, we can see a decrease in delivery rate only for epidemic routing. What is striking is that RAPID-ODP (only data packets are restricted by offered storage) significantly outperforms RAPID. It demonstrates that the storage consumption of control packets largely affects the performance of RAPID, because its control packets almost occupy 98% of storage in the simulation. Although RAPID can achieve satisfactory performance in DieselNet with 40 GB of storage [2], where control packets are comparatively small enough to ignore, we conclude that RAPID may not be a good choice in resource-constrained devices, e.g., sensor nodes or Bluetooth devices.

In contrast with RAPID, RENA only needs limited storage (11 kB) for control packets and achieves almost the

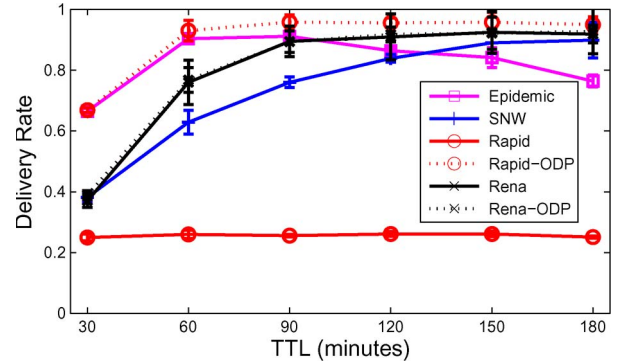


Fig. 5. TVC model: Delivery rate of TTL.

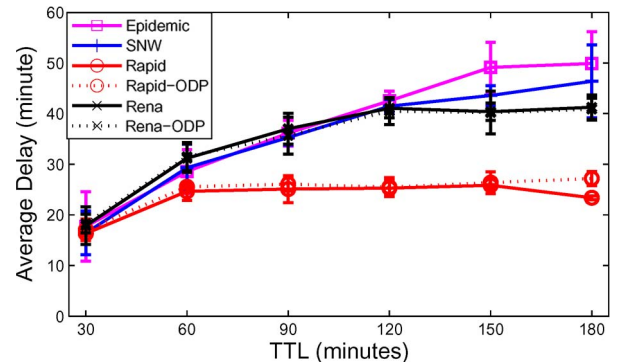


Fig. 6. TVC model—Average delay (TTL).

same delivery rate with RENA-ODP. More importantly, it performs better than SNW and RAPID in a stable manner when TTL increases, and its delivery rate is even within 7% of RAPID-ODP when TTL is bigger than 90.

Fig. 6 shows the average delay of the delivered packets of each protocol in scenario 1. It is observed that RAPID and RAPID-ODP have a smaller delay than other approaches as a result of more accurate estimation. RENA achieves either almost the same delay with SNW but a bigger delivery rate ($TTL < 150$) or a smaller delay than SNW and a similar delivery rate ($TTL \geq 150$). Note that the average delay is calculated from the delivered packet; therefore, we need to consider both average delay and delivery rate to make a fair comparison.

Figs. 7 and 8 show the delivery performance as TTL for data packet increases in the bus network scenario. Compared with the TVC model scenario, we observe similar trends in terms of both performance metrics. However, SNW achieves a much lower delivery rate than RENA and RAPID-ODP; even for 180 min of TTL time, the delivery rate of SNW is only 0.561. The reason is that SNW uses random distribution and passive waiting, which are not effective in large-scale group mobility situations.

2) *Impact of Storage*: Figs. 9 and 10 show results as the storage increases from 50 kB to 500 kB in scenario 1. RENA and SNW achieve a similar delivery rate, but RENA achieves much a smaller delay than SNW. Due to the passive waiting strategy, more storage would not help SNW decrease delay. Epidemic routing achieves similar performance with RENA when the storage is bigger than 250 kB. RAPID achieves

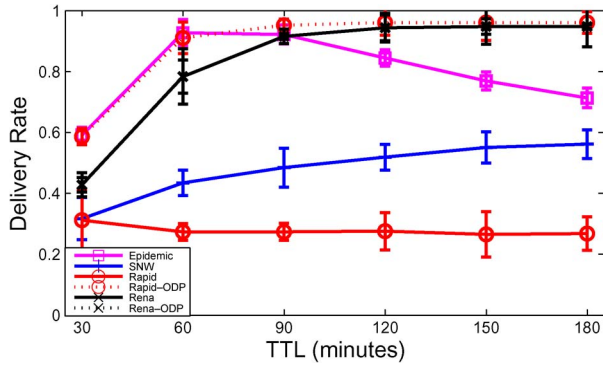


Fig. 7. Bus model: Delivery rate (TTL).

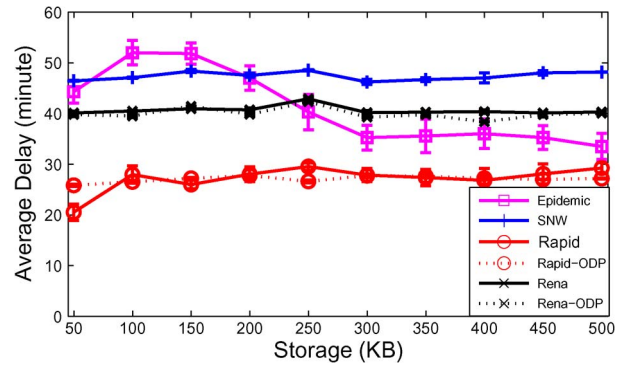


Fig. 10. TVC model: Average delay (storage).

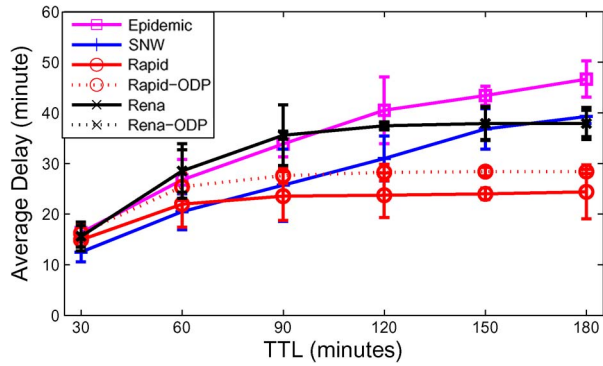


Fig. 8. Bus model: Average delay (TTL).

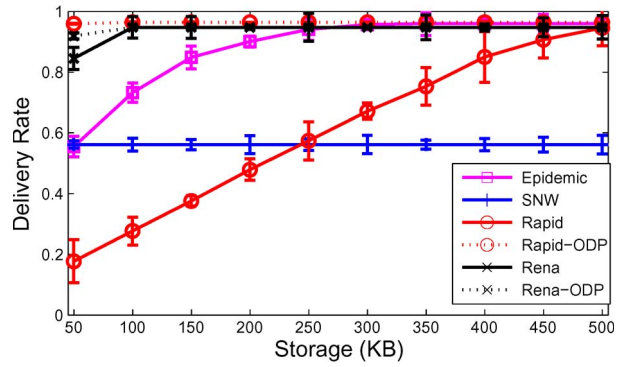


Fig. 11. Bus model: Delivery rate (storage).

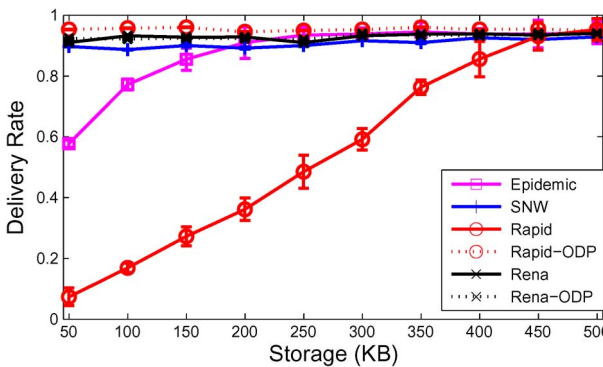


Fig. 9. TVC model: Delivery rate (storage).

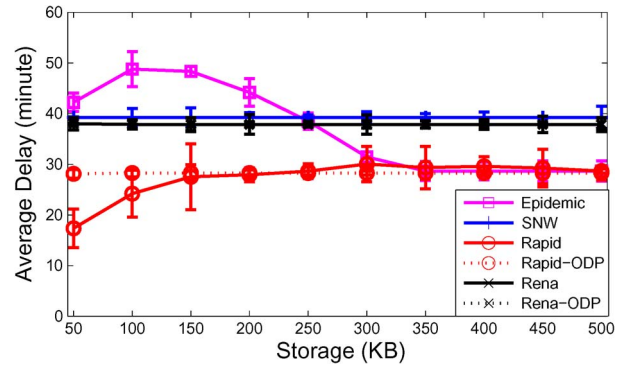


Fig. 12. Bus model: Average delay (storage).

similar reliability with RAPID-ODP and RENA only when the storage is bigger than 500 kB. However, note that, for epidemic routing and RAPID, the improvement is at the cost of higher overhead; with big-enough storage, such as 500 kB, epidemic routing even behaves as well as RAPID.

Figs. 11 and 12 again show the storage friendliness of RENA and the storage sensitivity of RAPID and epidemic routing. Because we have 16 regions in this setup, the control packets, in total, consume about 10 kB, and the average storage per region is about 650 B. Only in the 50-k storage condition is there an obvious difference between RENA and RENA-ODP. Although SNW is also storage friendly, its performance is much worse than RENA due to its passive waiting. Unlike RENA and SNW, RAPID and epidemic routing are markedly sensitive to storage restrictions. Probably due to the periodic nature of the bus network, we observe that most results are more stable than in scenario 1.

3) *Impact of the Number of Nodes:* In Figs. 13 and 14, we present the results as the number of nodes increases from 10 to 100 in the TVC mobility model. Note that all protocols show a decreasing trend on the average delay with an increase in the number of nodes (see Fig. 14), whereas they much differently behave in terms of the delivery rate (see Fig. 13). Epidemic routing first increases and then slowly decreases to a stable value, RAPID quickly decreases due to the fast expansion of control packets, and RAPID-ODP achieves the best performance (around 97% when $n > 70$), because it does not count the storage of control packets. RENA and SNW gradually increase to stable values of 96% and 94%, respectively, when $n > 80$.

The results clearly reflect the tradeoff between mobility benefit and overhead brought by the large number of nodes. Unlike RAPID and epidemic routing, RENA and SNW make good use of mobility due to restricted replications, which consume much less storage and bandwidth.

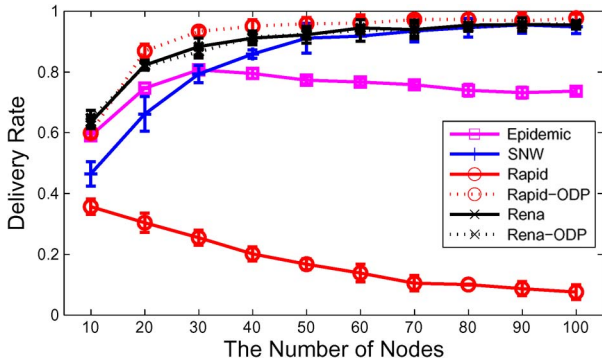


Fig. 13. TVC model: Delivery rate (node).

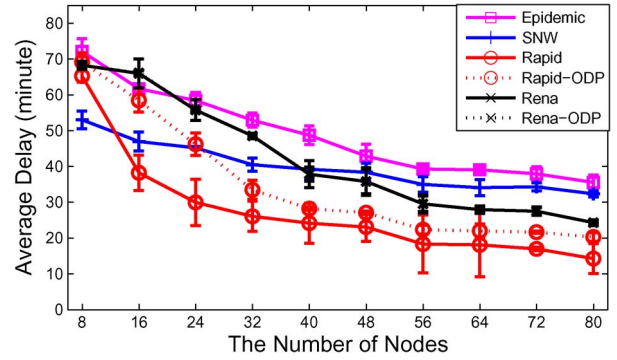


Fig. 16. Bus model: Average delay (node).

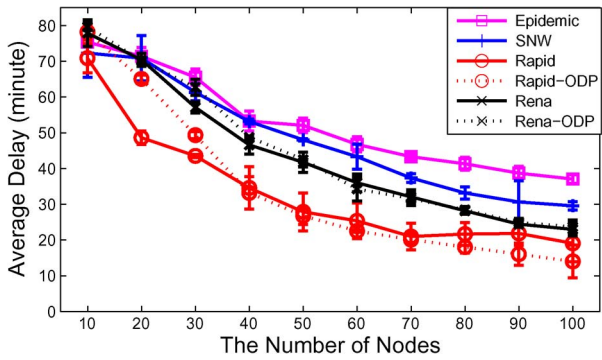


Fig. 14. TVC model: Average delay (node).

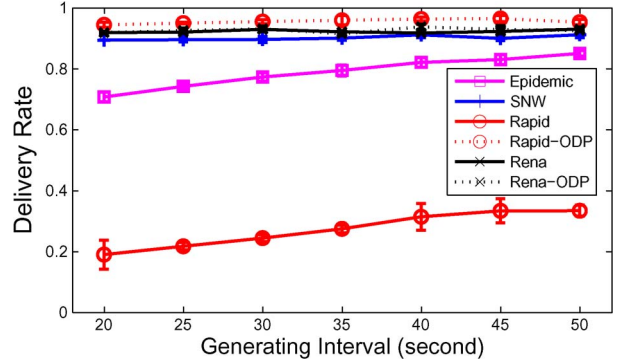


Fig. 17. TVC model: Delivery rate (traffic).

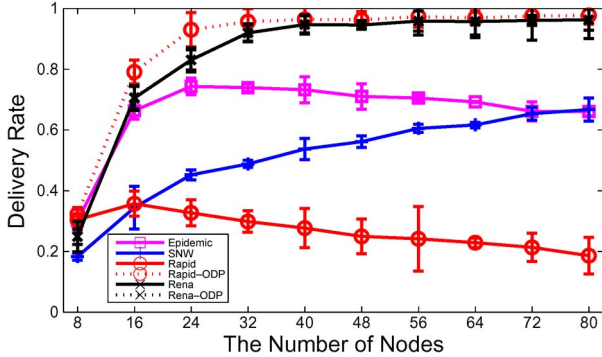


Fig. 15. Bus model: Delivery rate (node).

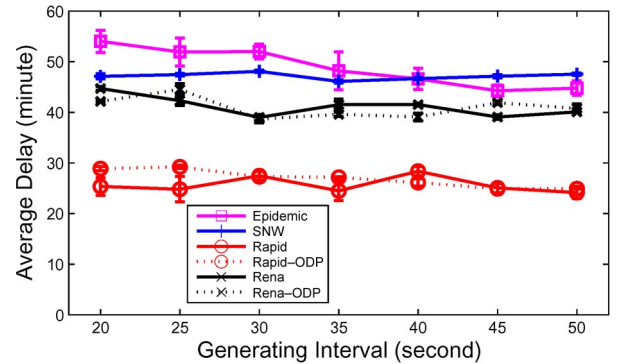


Fig. 18. TVC model: Average delay (traffic).

In the bus network, we evenly increase the number of nodes in eight bus routes. The results in Figs. 15 and 16 are similar to scenario 1. For the storage-friendly protocols RENA and SNW, the increase of nodes is positive for their delivery performance. With regard to the storage-sensitive protocols RAPID and epidemic routing, the increase of nodes improves the delivery performance only in an extremely sparse network ($n < 16$ for RAPID and $n < 24$ for epidemic routing) and finally worsens the delivery performance. Therefore, we conclude that the increase of network size may have an adverse effect on RAPID and epidemic routing in the storage-limited conditions.

4) *Impact of Traffic Load:* When we vary the packet-generating interval time from 20 s (high traffic) to 50 s (low traffic) in both scenarios, it is observed in Figs. 17–20 that epidemic routing and RAPID are obviously sensitive to traffic load, whereas RENA and SNW keep providing a stable performance. Again, RENA not only achieves a delivery rate that

is close to RAPID-ODP but has a smaller delivery delay than SNW and epidemic routing as well, whereas the delivery rate of RAPID still has significant differences from RAPID-ODP.

5) *Overhead:* We further define *delivery overhead* as the total number of redundant packets transmitted, normalized by the number of delivered packets, i.e.,

$$\frac{\text{relayed packets} - \text{delivered packets}}{\text{delivered packets}}$$

Figs. 21 and 22 clearly reflect the overhead of different protocols when the network size increases. In both scenarios, epidemic routing causes numerous transmissions, particularly in the bus network. Although SNW consumes the least overhead as a result of passive waiting, its delivery performance is clearly unsatisfactory. RENA always consumes less overhead than RAPID and RAPID-ODP, which means that RENA is more efficient in delivering packets.

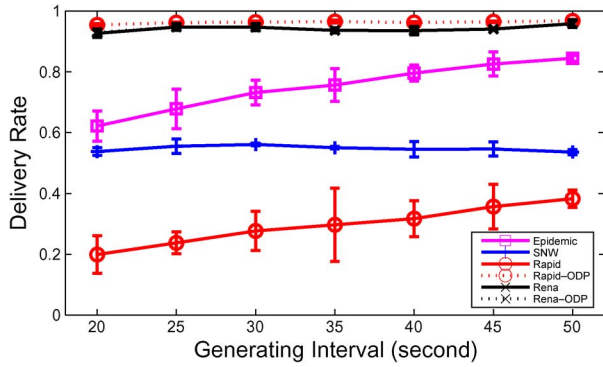


Fig. 19. Bus model: Delivery rate (traffic).

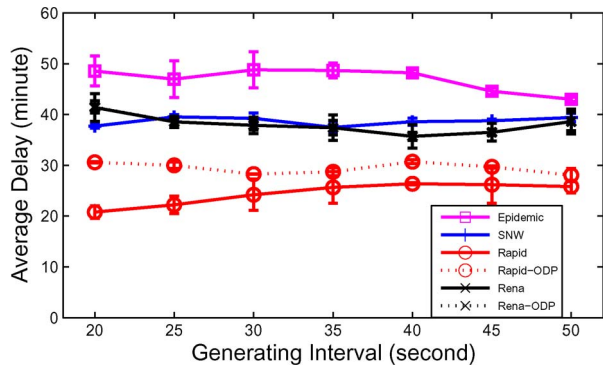


Fig. 20. Bus model: Average delay (traffic).

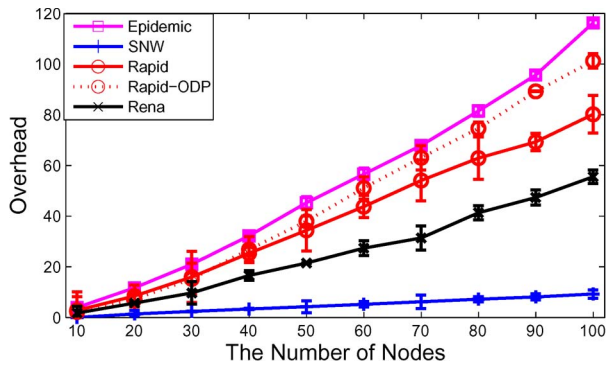


Fig. 21. TVC model: Overhead (node).

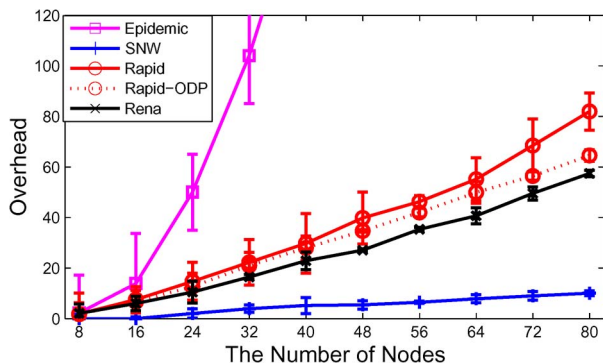


Fig. 22. Bus model: Overhead (node).

Based on the aforementioned simulation results, we observe that RENA achieves much stable performance than RAPID, only at the cost of a relatively low storage and overhead.

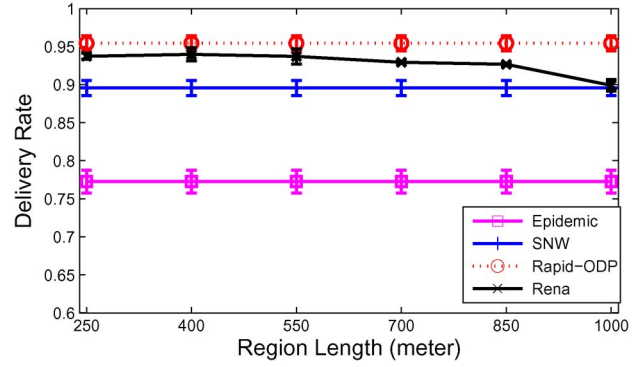


Fig. 23. TVC model: Delivery rate (size).

VI. DISCUSSION

When we apply RENA, it is crucial to answer how we can choose an appropriate region size. In fact, different choices depend on the specific *granularity* of the movement in different applications. In addition, we should consider the storage occupied by control packets. In RENA, the average storage per region consumed by the control overhead is about 650 B. Then, we can restrict the total control overhead by controlling region size. In this paper, our empirical analysis suggests that it is better to keep control packets under 20% of the whole storage.

Therefore, a reasonable approach for selecting a suitable region is to maximize the benefit of macro-level spatial information without having to track the specific locations and consuming excessive storage. We can imagine that when the granularity of region partition is more fine grained than the granularity of the movement, more regions may not bring much benefit.

Fig. 23 shows the results as the length of each region increases in the TVC model, where the number of regions decreases from 16 to 1. (Because the delivery rates of protocols are much different, it is unfair to compare their delivery delays here.) It is observed that the influence of region length in the large-scale bus network is more apparent than in the small-scale WLAN case. These simulation results verify the value of the region-related pattern, and further investigation about the region will be our future work.

Although RENA is proposed based on a Markov mobility model, it can achieve feasibility in most scenarios. When mobility conditions of users do not follow the Markov property, RENA could still work well if there are spatial-related characteristics that widely exist in real-life mobility. Even in the extreme case, without any location preference property, such as random mobility models, RENA could naturally transfer to SNW. In other words, the delivery performance of SNW is the *lower bound* of RENA. The results in Fig. 23 have directly justified this conclusion.

VII. CONCLUSION

In this paper, we have introduced a novel concept of region-related patterns extracted from the spatial property in ICMN. Then, we took advantage of this macromobility information and proposed a region-based routing protocol RENA. Compared with epidemic routing and the restricted-replication protocol SNW, RENA achieves not only higher delivery rates but shorter

delivery delays as well, only at the cost of a relatively low storage consumed by control packets. In terms of storage and feasibility, RENA is more storage friendly and adaptive than encounter-based RAPID. Therefore, we conclude that RENA may be a better choice for resource-constrained devices in ICMN.

REFERENCES

- [1] P. Juang, H. Oki, Y. Wang, M. Martonosi, L. S. Peh, and D. Rubenstein, "Energy-efficient computing for wildlife tracking: Design tradeoffs and early experiences with ZebraNet," in *Proc. ASPLOS-X*, 2002, pp. 96–107.
- [2] A. Balasubramanian, B. N. Levine, and A. Venkataramani, "DTN routing as a resource allocation problem," in *Proc. ACM SIGCOMM*, Japan, Jan. 2007, pp. 373–384.
- [3] P. Hui, J. Crowcroft, and E. Yoneki, "Bubble rap: Social-based forwarding in delay-tolerant networks," in *Proc. ACM MobiHOC*, Hong Kong, May 2008, pp. 241–250.
- [4] K. Fall, "A delay-tolerant network architecture for challenged Internets," in *Proc. SIGCOMM*, Germany, Jan. 2003, pp. 27–34.
- [5] S. Jain, K. Fall, and R. Patra, "Routing in a delay-tolerant networking," in *Proc. SIGCOMM*, Portland, OR, Jan. 2004, pp. 145–158.
- [6] H. Dubois-Ferriere, M. Grossglauser, and M. Vetterli, "Age matters: Efficient route discovery in mobile ad hoc networks using encounter ages," in *Proc. ACM MobiHOC*, Annapolis, MD, Jun. 2003, pp. 257–266.
- [7] J. Burgess, B. Gallagher, D. Jensen, and B. N. Levine, "Maxprop: Routing for vehicle-based disruption-tolerant networks," in *Proc. IEEE INFOCOM*, Barcelona, Spain, 2006, pp. 1–11.
- [8] T. Karagiannis, J. L. Boudec, and M. Vojnovic, "Power law and exponential decay of intercontact times between mobile devices," in *Proc. ACM MobiCom*, Montréal, QC, Canada, Sep. 2007, pp. 183–194.
- [9] W. Hsu, A. Spyropoulos, K. Psounis, and A. Helmy, "Modeling time-variant user mobility in wireless mobile networks," in *Proc. IEEE Infocom*, Anchorage, AK, May 2007, pp. 758–766.
- [10] A. Boukerche, *Algorithms and Protocols for Wireless Mobile Ad Hoc Networks*. Hoboken, NJ: Wiley, 2008.
- [11] A. Vahdat and D. Becker, "Epidemic routing for partially connected ad hoc networks," Duke Univ., Durham, NC, Tech. Rep., 2000.
- [12] T. Spyropoulos, K. Psounis, and C. Raghavendra, "Spray and wait: An efficient routing scheme for intermittently connected mobile networks," in *Proc. ACM WDTN*, Aug. 2005, pp. 252–259.
- [13] Y. Wang, S. Jain, M. Martonosi, and K. Fall, "Erasure-coding-based routing for opportunistic networks," in *Proc. ACM WDTN*, Aug. 2005, pp. 229–236.
- [14] A. Lindgren, A. Doria, and O. Schelen, "Probabilistic routing in intermittently connected networks," in *Proc. SAPIR*, vol. 3126, LNCS, 2004, pp. 239–254.
- [15] T. Spyropoulos, K. Psounis, and C. Raghavendra, "Efficient routing in intermittently connected mobile networks: The multiple-copy case," *IEEE/ACM Trans. Netw.*, vol. 16, no. 1, pp. 77–90, Feb. 2008.
- [16] J. Karvo and J. Ott, "Time scales and delay-tolerant routing protocols," in *Proc. ACM Chants Workshop*, San Francisco, CA, Sep. 2008, pp. 33–40.
- [17] J. Leguay, T. Friedman, and V. Conan, "DTN routing in a mobility pattern space," in *Proc. ACM Chants Workshop*, Philadelphia, PA, Aug. 2005, pp. 276–283.
- [18] W. Hsu, D. Dutta, and A. Helmy, "Profile-cast: Behavior-aware mobile networking," *ACM SIGMOBILE Mobile Comput. Commun. Rev.*, vol. 12, no. 1, pp. 52–54, Jan. 2008.
- [19] J. Ghosh, S. J. Philip, and C. Qiao, "Sociological-orbit-aware location approximation and routing (SOLAR) in MANET," in *Proc. IEEE Broadnets*, Boston, MA, Oct. 2005, pp. 641–650.
- [20] H. Wen, J. Liu, C. Lin, F. Ren, P. Li, and Y. Fang, "Rena: Region-based routing in intermittently connected mobile network," in *Proc. ACM MSWiM*, Canary Islands, Spain, 2009, pp. 280–287.
- [21] J. Broch, D. A. Maltz, D. B. Johnson, Y. C. Hu, and J. Jetcheva, "A performance comparison of multihop wireless ad hoc network routing protocols," in *Proc. MobiCom*, 1998, pp. 85–97.
- [22] M. Musolesi and C. Mascolo, "A community-based mobility model for ad hoc network research," in *Proc. REALMAN*, Florence, Italy, May 2006, pp. 31–38.
- [23] X. Zhang, J. Kurose, B. N. Levine, D. Towsley, N. Banerjee, and H. Zhang, "Study of a bus-based disruption-tolerant network: Mobility modeling and impact on routing," in *Proc. ACM MobiCom*, Montreal, QC, Canada, Sep. 2007, pp. 195–206.
- [24] A. Keranen, J. Ott, and T. Karkkainen, "The one simulator for DTN protocol evaluation," presented at the SIMUTools Conf., Rome, Italy, 2009.



Hao Wen is currently pursuing the Ph.D. degree with the Department of Computer Science and Technology, Tsinghua University, Beijing, China.

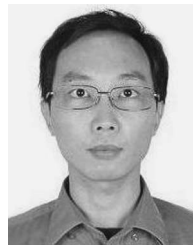
His research interests include analysis and design for wireless sensor networks and delay tolerant networks.

Mr. Wen received the Microsoft Asia Scholarship.



Fengyuan Ren received the Ph.D. degree in computer science from the Northwestern Polytechnical University, Xi'an, China, in December 1999.

He is currently a Professor with the Department of Computer Science and Technology, Tsinghua University, Beijing, China. His research interests include network traffic management and control, control in/over computer networks, wireless networks, and wireless sensor networks.



Jia Liu is currently pursuing the Ph.D. degree with the Department of Computer Science and Technology, Tsinghua University, Beijing, China.

He is especially interested in broadband wireless communications, computer network performance evaluation, etc.



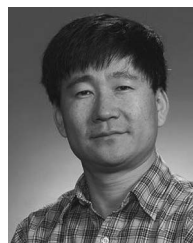
Chuang Lin received the Ph.D. degree in computer science from Tsinghua University, Beijing, China, in 1994.

He is currently a Professor and the Head of the Department of Computer Science and Technology, Tsinghua University. His research interests include computer networks, performance evaluation, network security analysis, and the Petri net theory and its applications.



Pan Li received the Ph.D. degree in electrical and computer engineering from the University of Florida, Gainesville, in August 2009.

He is currently an Assistant Professor with the Department of Electrical and Computer Engineering, Mississippi State University.



Yuguang Fang (F'08) received the Ph.D. degree in systems engineering from Case Western Reserve University, Cleveland, OH, in January 1994 and the Ph.D. degree in electrical engineering from Boston University, Boston, MA, in May 1997.

He is currently with the University of Florida, Gainesville.

Dr. Fang is a member of the Association for Computing Machinery.