Research Article

RHPMAN: Replication in Highly Partitioned Mobile Ad Hoc Networks

Ke Shi and Hongsheng Chen

School of Computer Science and Technology, Huazhong University of Science and Technology, Wuhan, Hubei 430074, China

Correspondence should be addressed to Ke Shi; keshi@mail.hust.edu.cn

Received 25 December 2013; Accepted 23 May 2014; Published 30 June 2014

Academic Editor: Shihong Zou

Copyright © 2014 K. Shi and H. Chen. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Accessing data in mobile ad hoc networks is a challenging problem, which is caused by frequent network partitions due to node mobility and due to the impairments of wireless communications. The partitioning pattern is studied by examining the statistics of network partitions for a number of mobility models. Then the relation between the network partitioning pattern and the effectiveness of the data replication scheme is established. Based on these results, a novel replication scheme, RHPMAN (replication in highly partitioned mobile ad hoc network), taking into account the fact that the network is often partitioned in smaller portions, enjoying only intermittent connectivity thanks to mobile nodes traveling across partition, is proposed. In RHPMAN, data items are replicated to the nodes with rather stable neighboring topology and with enough resources. A semiprobabilistic data disseminating protocol is employed to distribute the replicas and propagate the updates, which can identify the potential mobile nodes traveling across partitions to maximize data delivery. To maintain replica consistency, a weak consistency model is utilized to ensure that all updates eventually propagate to all replicas in a finite delay. Simulation results demonstrate that RHPMAN can achieve high data availability with low overhead.

1. Introduction

Nodes in a wireless and mobile ad hoc network (MANET) are connected by wireless links [1] and function not only as end-systems but also as routers, forwarding transmissions of other nodes. If two nodes are sufficiently close to each other, they can communicate directly; otherwise, the nodes can communicate through other intermediate nodes, provided that there is a path between the two nodes; that is, there is no network partition which prevents establishment of such a path. The protocol reception model is often employed to model the direct communication between nodes: if the nodes are within some distance from each other, referred to as the wireless transmission range, a direct link can be established between them.

Accessing remote data has been a fundamental application in both fixed and mobile networks. However, distributed data access in MANETs has proved to be a much more difficult problem than in fixed networks, since, due to node mobility and due to impairments of wireless transmission, network partitions occur frequently. Such network partitions prevent access to data objects residing in a network partition by nodes which are currently placed outside this partition. Thus, data accessibility and availability in MANETs can be significantly lower than in fixed networks. In this work, we define data accessibility as the ability to obtain information about the data from other nodes and to access those data when needed.

A possible solution to the low accessibility of data in MANETs is to replicate data items on several mobile nodes. As depicted in the example in Figure 1, replicas of data items $D_1$ and $D_2$, which are replicated on nodes in the two network partitions before network partitioning caused by the movement of node Q happened, allow every mobile node in the network to access the data objects, in spite of the network partitioning.

Of course, the extreme solution of replicating every data item on every node in MANETs is impractical due to limited resources of the nodes in MANETs. However, limited replication is a widely used redundancy scheme for improving the reliability of data access in distributed systems at the cost of increased storage space and communication
overhead. Suppose that each data item is replicated into \( k \) copies and stored on different network nodes. Of course, to obtain the data item, one needs access only to one out of the \( k \) replicas. In fixed networks, where network connectivity is almost 100\%, we usually assume that availability of nodes is independent and identically distributed (i.i.d.) random variable. The value of \( k \) should be set based on the target \( \varepsilon \) availability and depending on the average node availability \( \alpha \):

\[
k = \frac{\log(1 - \varepsilon)}{\log(1 - \alpha)}.
\]

Based on (1), if the availability of network nodes is high (i.e., node failures are infrequent), it is usually sufficient to create only a few replicas for achieving high level of data accessibility. Of course, (1) does not hold in MANETs, since the availability of nodes is not i.i.d. anymore, and it depends on the location of the nodes. To demonstrate the dependence of nodes’ availabilities, consider a node with a single neighbor. Failure of the link to this neighbor inhibits communication of the node with every replica of a data object in the system. Furthermore, the average node availability cannot be assumed anymore to be high. Consequently, the solution of limited replication needs to be reexamined for the MANET case. In particular, the solution should allow efficient update propagation among replicas based on the intermittently connected network where connectivity is not 100\% anymore. Considering the example in Figure 1, if data item \( D_1 \) is updated, this update needs to be propagated to its replicas in the other partition as soon as possible; otherwise the nodes in the other partition may get stale data. In this example, whether node Q or other nodes will move into the previous locating area of Q to connect two separate partitions again in a finite delay has a significant influence on the performance of replication-based data access.

In this paper, we present a novel replication scheme, RHPMAN (replication in highly partitioned mobile ad hoc network), to address this issue. In particular, the paper studies network partitioning pattern by examining the detailed statistics of network partitions, including partition size and interconnecting time, and then attempts to design an efficient partition-aware data disseminating method for replica distribution and update propagation based on obtained partitioning pattern data. The contributions of this study are as follows:

1. characterizing the statistics of network partitions,
2. maximizing data delivery despite network partitions and intermittent connectivity by designing a partition-aware semiprobabilistic data disseminating protocol,
3. proposing a novel replication scheme, RHPMAN, to improve the performance of data access in MANETs.

RHPMAN can also be applied in mobile sensor networks that emerge as a new class of sensor networks with the recent advances in distributed robotics and low power embedded systems. Mobile sensor networks can be used in land, ocean, and air exploration and monitoring, automobile applications, habitat monitoring, and a wide range of other scenarios. A typical example is the participating sensing system that collects information from a plurality of mobile devices, such as cellular telephones, using sensors, such as microphones, cameras, accelerometers, and biometric readers, to detect the sensory information.

Mobile sensor networks do not have a stationary network infrastructure, which makes continuous data acquisition to some sink point a nonintuitive task. Data acquisition needs to be succeeded by in-network storage. Data generated inside the network is first stored in the network and then retrieved by the user or uploaded to the faraway base station via different means. These uploading opportunities could be periodic visit by human operator or data mule. Despite advances in large lower-power flash memory, storage is not an “infinite” resource in mobile sensor networks. How to store the data in the finite storage space to ensure that the users can retrieve the data becomes a very important problem to address.

Using RHPMAN, important data can be replicated on the sensor nodes with larger storage space and higher residual power and disseminated through the sensor nodes with higher mobility. The successful rate of data retrieval or uploading can be improved just like the data accessibility can be improved in mobile ad hoc network.

The rest of the paper is organized as follows. An overview of related work is provided in Section 2. Section 3 discusses the network partitioning pattern based on simulating method and derives an analytical model relating the network partitioning patterns to the data replication effectiveness. The details of RHPMAN, including partition-aware semiprobabilistic data disseminating protocol, are given in Section 4. Section 5 presents the results of performance evaluation. Conclusions and future work are considered in Section 6.

2. Related Work

Data replication is a traditional technique in distributed systems, which offers the benefits of improving the data availability and shortening the response time for data read/write operations at the cost of increased storage space and communication overhead [2]. Many strategies (e.g., [3–5]) have been proposed in the technical literature. However, these studies
relate to wired network, where network partitions seldom occur and where the assumption that the availability of nodes and the corresponding network connections are independent is reasonable.

In the field of mobile computing, several strategies (e.g., [6, 7]) taking the intermittent connectivity of nodes explicitly into account for data replication have been proposed. The goal of these strategies is to reduce the cost of wireless communication and to compensate for its reduced reliability, by trading off availability, consistency, and communication costs. However, these studies assume only one-hop wireless communication and, therefore, are unsuitable for evaluation of data replication in the multihop ad hoc networking environment.

Hara [8–10] proposed a series of data replication schemes in ad hoc networks. In order to guarantee data accessibility upon network partitioning, these works focus on optimizing the location of data replicas within a network and are based on the assumption that access frequencies to data items from each node are known and are fixed. This assumption limits the applicability of the schemes in practical systems.

Similarly to Hara’s work, Wang and Li [11], Huang et al. [12], and Derhab and Badache [13] considered the problem of replica allocation. Their approach takes into consideration topological information when replicating data and data replication occurs only when necessary according to certain partition detection schemes. However, in these works, partition detection depends on the mobility model. Furthermore, it is assumed that the locations and velocities of all mobile nodes are known. These assumptions are too restrictive to capture the reality of mobile ad hoc networks. A survey of data replication in mobile ad hoc network was given by Derhab et al. [14].

In delay-tolerant network, replication is widely used to improve the performance of data delivery [15]. However, instead of data access, most methods focus on data routing. Zhuo et al. [16] proposed a contact duration aware data replication scheme in delay-tolerant network, which optimized the data availability based on the detail information of node contact. The number of contacts and the time of each contact lasting are needed to decide the location of each replica.

The above works on data replications in ad hoc networks try to improve data accessibility by investigating the problem of dynamic replica allocation. However, these works do not study the characteristics of network partitioning, characteristics which have critical influence on the performance of data replication in MANETs. Our scheme is based on these characteristics.

### 3. Partitioning Pattern Analysis

#### 3.1. Definition

We study the characteristics of network partitioning by statistical analysis of simulation results. In our simulation, the reception range of the node is indicated by statistical analysis of simulation results. In our work, partition size is scale-free power-law like

$$p(k) = P(S_{NP} = k) = b \times k^{-\alpha} + c.$$  \hspace{1cm} (5)

An ad hoc network is modeled as a graph $G(V, E)$, where $V$ vertices (nodes) that form the set $V$ of vertices are uniformly placed within the finite area $A$, and any two vertices with distance of $r$ or less between them are connected by an edge (a direct communication link) that forms the set $E$ of edges. A network partition is a subset $NP \subseteq V$ where

1. a path exists between all pairs of vertices $n_i, n_j \in NP$,
2. no path exists between any pair of vertices $n_i \in NP$, $n_k \in V - NP$.

The degree of a node $v$, denoted as $d(v)$, represents the number of neighbors of $v$, that is, the number of links. A node of degree $d = 0$ is isolated; that is, it has no neighbors. The average (or mean) node degree is defined as

$$d = \frac{1}{n} \sum_{v \in G} d(v).$$  \hspace{1cm} (2)

In a $G(n, r, A)$, average node degree is

$$d = (n - 1) \times \frac{\pi r^2}{A}.$$  \hspace{1cm} (3)

Node mobility is based on random-trip models on wrapping and nonwrapping two-dimensional surfaces.

#### 3.2. Partition Size

We seek to obtain the distribution of partition size to describe the characteristics of network partitioning in MANETs. Since partition size is a discrete value with finite range from 1 to $n (n = \|N\|)$, the probability mass function (PMF) of the partition size, $p(k) = P(\|NP\| = k)$, is utilized to describe this distribution.

Given the above PMF, the average size of network partitions in the network $S_{NP}$ ($S_{NP}^{a} = \|NP\|$) is calculated as

$$S_{NP}^{a} = \sum_{k=1}^{n} p(k) \times k.$$  \hspace{1cm} (4)

We observed multimodal distributions of partition size from the results of our simulation. For different values of average node degree $d$, partition size can be approximated by different distributions. Figure 2 displays the PMF of partition size for random walk mobility model on a nonwrapping 1000 [m] x 1000 [m] square area and for nine different average node degrees. The radio communication range of each mobile node is a circle with the radius of $r$, 100 [m]. The node’s speed $v$ is selected uniformly from the range of 1–10 [m/s] every 100 [s].

We observe that if $d$ is greater than $d_0$, around 6, the partition size has a bimodal distribution: the probability that the partitions of large size or small size exist is high, and the probability that the partitions of medium size exist is low. This result also demonstrates that the network is almost surely connected when $d_0$ is high.

If $d$ is around or below $d_0$, we observe that the distribution of partition size is scale-free power-law like

$$p(k) = P(S_{NP} = k) = b \times k^{-\alpha} + c.$$  \hspace{1cm} (5)

In delay-tolerant network, replication is widely used to improve the performance of data delivery [15]. However, instead of data access, most methods focus on data routing.
Partitions of all sizes can be encountered. When $d$ is decreased significantly below $d_c$, the probability that the partitions of large size exist becomes very small. There exist a large number of small partitions composed of few nodes. When $d$ is small, the node only has few neighbors. Therefore, the possibility that many nodes form a large partition is small. When $d$ is less than 1, we find that the distribution of partition size can be approximated by the geometric distribution:

$$p(k) = P(S_{NP} = k) = p(1 - p)^{k - 1},$$

$$p = \sqrt{e^{-\lambda}}, \quad \lambda = (n - 1) \times q, \quad q = \frac{\pi r^2}{A}.\quad (6)$$

The distribution of partition size is determined by the spatial distribution of nodes. If the node mobility does not change the spatial distribution of nodes, the distribution of partition size does not change as well. Among random-trip models, random walk mobility, random waypoint model on a wrapping surface [17], and community based model [18] do not change the spatial distribution.

We can get some intuitive knowledge of data replication effectiveness in different cases from the results of our simulation. When average node degree is greater than $d_c$, only a few replicas suffice to achieve large data accessibility because most nodes belong to the largest partition. For the same reason, increasing the number of replicas cannot significantly further improve the data accessibility in this situation. When average node degree is less than $d_c$, the distribution of partition size is scale-free power-like, and there exist a few larger partitions containing most nodes in the network. Large data accessibility can be achieved through allocating enough replicas to these partitions. Replication is then an effective approach to improve data accessibility. If average node degree is less than 1, the size of most partitions is very small and the relationship between the number of replicas and data accessibility is approximately linear. This implies that replication is not an efficient approach to improve data accessibility.

### 3.3. Interconnecting Time

Based on previous discussion, replication may be an efficient solution to improve data accessibility in a MANET whose average node degree falls into certain value area. However, this kind of MANET is highly partitioned, and the mobile nodes in such a MANET are intermittently connected. The knowledge about intermittent connectivity is very helpful to design an efficient data disseminating protocol to distribute replicas or propagate updates to the replicating nodes located in the different network partitions, which has a significant influence on the performance of corresponding replication strategy.

We are interested in the characteristics of connection opportunities of intermittent connection that has the most
significant impact on the feasibility of delivering data across different network partitions. We choose to characterize these opportunities in terms of the interconnecting time that is defined as the time gap separating two connections of the same pair of nodes. There exists a connection between two nodes if and only if these two nodes are in the same network partition. The interconnecting time indicates how long the disconnection between two nodes lasts. This parameter strongly affects the feasibility of opportunistic networking and has rarely been studied in the literature. The nature of the distribution will affect the choice of suitable data delivery strategies to be used to maximize the successful data delivery in a bounded delay.

The left part of Figure 3 displays cumulative distribution function (CDF) of the intermeeting time between two nodes for three different average node degrees (data replication may be an efficient solution under these circumstances according to the previous analysis). The rest of simulation settings are the same with the settings adopted in Figure 2.

We observe that the distributions of interconnecting time show the same behavior, an approximate exponential, as evidenced by the right part of Figure 3, in which the curves of the complementary CDF for the intermeeting time on a log-scale are almost straight.

The exponential distribution of interconnecting time implies that data items can be delivered from one node to another node in a bounded delay with very high probability despite network partitions. Therefore, replicas can be distributed to the replicating nodes in a bounded delay with high probability, and updates can be propagated to corresponding replicas in a bounded delay with high probability too. It makes designing a feasible data replication strategy that can adapt highly partitioned MANETs and optimize the trade-off among storage, data accessibility, delay, consistency level, and overhead depending on the current network operational conditions possible. The next section will discuss such a replication strategy, called RHPMAN, replication in highly partitioned MANETs.

3.4. Analytical Model of Replication Effectiveness. Consider a mobile wireless ad hoc network modeled by a \( G(n, r, A) \) and nodes that attempt to access data objects each represented by \( n_r \) replicas uniformly distributed in the nodes within the network. Data accessibility \( \alpha_d \) can be considered as the probability of the successful data access, \( P_{da} \).

The sufficient condition for a successful data access is that the node initiating data access request has the replicas of requested data object or that the node initiating data access request is located in the same partition with at least one node that has the replicas of the requested data object. Therefore,

\[
P_{da} = P_r + (1 - P_r) \times P_{NP}
\]  

(7)

where \( P_r \) is the probability that the node initiating data access request has the replicas of requested data object; \( P_{NP} \) is the probability that the node initiating data access request is located in the same partition as at least one node that has the replicas of the requested data object.

If data replicas are randomly distributed, \( Pr = n_r/n \). Given the distribution of partition size, \( P_{NP} \) can be computed as follows:

\[
P_{NP} = \sum_{i=1}^{N_P} P(a, NP_i) \times P(r, NP_i),
\]  

(8)

where \( P(a, NP_i) \) is the probability that the node initiating data access request is in a certain partition \( NP_i \) and \( P(r, NP_i) \) is the probability that this partition contains at least one node that has the replicas of the requested data object,

\[
P(a, NP_i) = \frac{S_{NP_i}}{n},
\]  

\[
P(r, NP_i) = 1 - \left( \frac{n_r - 1}{S_{NP_i} - 1} \right) \frac{n - 1}{\pi - 1}
\]  

(9)
Finally, data accessibility is

$$\alpha_d = P_{da} = \frac{n_x}{n} + \left(1 - \frac{n_x}{n}\right) \times \sum_{i=1}^{n_{NP}} \frac{S_{NP_i}}{n} \times \left(1 - \left(1 - \frac{n - n_x - 1}{S_{NP_i} - 1}\right)^{n_x / n-1}\right),$$

where $N_{NP}$ and $S_{NP}$ can be obtained from the approximated distribution discussed before.

If we assume that data replicas are optimally distributed, that is, the ideal replica allocating algorithm can get all necessary information and allocate data replicas to the partitions with the largest size, data accessibility is

$$P_{da} = \frac{n_x}{n} + \left(1 - \frac{n_x}{n}\right) \times \frac{S_{NP}}{n} \geq S_{NP},$$

which can be seen as the upper bound of the data accessibility.

The above analytical model gives us a standard to evaluate how well the data replication schemes do for the read-only data access. If data update exists, successful data access also means getting the latest data, which means that the successful data access rate would be affected by the replica update delay decided by intermeeting time.

4. RHPMAN Scheme

Based on the above section, a good replication algorithm should allocate data replicas to the nodes in the larger partitions and use the nodes with high mobility to deliver the updates. Fortunately, this kind of scenario exists in the real applications where the mobile devices such as laptops, PDAs, and smart phones carried by humans construct MANETs by the needs of humans to socialize/cooperate or mobile sensor network to collect sensory information, for example, disaster relief teams in the emergency, platoons of soldiers in the battlefield, and groups of people in the party.

The partition results in these real networks may be not exact with that observed in the homogeneous networks considered in the last section. However, we can gain some insight into the partitioning behavior of actual networks from these results obtained in the last section. First, network partitioning always exists. Nonuniform node distribution, complicated propagation environments, and other constraints imposed by the real application scenarios make network partitioning occur more frequently and make the possibility that large partitions appear higher. Second, the movement based on human decisions and socialization behaviors improve connecting opportunities between the pair of nodes in the intermittently connected network.

Consider a party scenario consisting of a lot of guests. The guests with the same interesting topic form relatively stable group. The rest of guests frequently roam between different groups. If we make the powerful devices carried by the guests in stable group store information and make the devices carried by the guests roaming between different groups spread information, all the guests may get their wanted information through their carried device with high probability. This idea is also applied to the platoons of soldiers in the battlefield. The commander and his soldiers form a stable group, and some special soldiers such as signalman and scout roam between these groups. The commander may be the good choice for replicating node, and the signalman and scout may be the good choice for delivering data among replicating nodes.

The basic idea of RHPMAN is to replicate desired data items to the nodes with rather stable neighboring topology and with enough resources such as energy and storage space and then employ a semiprobabilistic data disseminating protocol to identify the potential mobile nodes traveling across partitions to distribute the replicas and propagate the updates despite network partitions and intermittent connectivity.

In RHPMAN, we assume that the nodes present in the system cooperate to share the data with each other, including hosting the replicas and delivering the data. In other words, we do not consider the case of nodes that may refuse to store a replica or deliver a data item.

4.1. Semiprobabilistic Data Dissemination Protocol. In the semiprobabilistic data dissemination protocol, a mobile node maintains a disseminating zone of $h$ hops from itself. If the receiving node is present in the disseminating zone of the sending node, data is delivered synchronously based on local zone information. If data cannot be delivered synchronously, data are forwarded towards a subset of nodes in the disseminating zone that have the higher chance of successful delivery. The chances of successful delivery are synthesized from locally available zone information such as the rate of change of connectivity of a node (the likelihood of it connecting other nodes) and its current energy level (the likelihood of it remaining alive to deliver the message). Therefore, this approach performs data disseminating by combining probabilistic and deterministic decisions: the former is resilient to change and therefore addresses dynamicity, while the latter reduces indiscriminate propagation by guiding messages towards the receivers.

We define profile as the set of attributes that describe the aspects of the system that can be used to optimize the process of data dissemination. Every node periodically exchanges the information related to both the deterministic data dissemination and the probabilistic data dissemination. For deterministic data dissemination, every node maintains a routing table to all the nodes in its disseminating zone (we use DSDV [14]; the entries in the routing tables include distances and next hop node identifier). For probabilistic data dissemination, every node maintains a list containing its delivery probabilities for the data receivers. The propagation of such profile information only within a neighborhood of $h$ hops provides reasonable accuracy while keeping the overhead under control, since $h$ can be adjusted accordingly to the network operation conditions.

Each node periodically calculates its delivery probabilities based on locally available profile information. The delivery
probability of a node $i$ for disseminating data $j$ can be computed as

$$P_{ij} = \begin{cases} 1 & \text{if node } i \text{ is the receiver of data } j \\ p_{ij}^d & \text{otherwise.} \end{cases}$$  \hspace{1cm} (12)

If node $i$ is a receiver for data $j$, its delivery probability is the highest possible, equal to 1. If node $i$ is not a receiver, it can still act as an intermediate forwarder or carrier in the disseminating process, with $p_{ij}^d$ representing its effectiveness either as a forwarder or as a carrier of disseminating data $j$ to its corresponding receivers.

Many parameters affect the ability of a node to be a good forwarder or a carrier in the data disseminating process. For instance, a node with a high change degree of connectivity frequently changes its set of neighbors and therefore has more options for forwarding. Instead, the probability of receiver colocation can be used as a direct indicator of the likelihood of a node to connect a receiver, therefore enabling direct data delivery. The node residual energy determines whether it has enough power to stay alive long enough to meet other hosts, possibly after traveling from one partition to the other, and disseminate the data further. Finally, the node free buffer space is a direct measure of the ability of the node to carry the data altogether. In our work, the profile only includes the first two attributes mentioned above, that is, the change degree of connectivity and the probability of receiver colocation. These are indeed the most important attributes for our purposes, and by limiting ourselves to their description we keep our treatment simpler. However, the profile is general and open to inclusion of any other attribute. Formally, we define $p_{ij}^d$, as

$$p_{ij}^d = w_{col}U_{col} + w_{col}U_{col}$$  \hspace{1cm} (13)

$U_{col}$ represents the estimated value of the change degree of connectivity of node $i$, that is, the number of connections and disconnections that the node has experienced over the last $T$ seconds. A high value of this estimate means that node $i$ is likely to be in reach of a large number of different nodes. Instead, $U_{col}$ summarizes the history of colocation of $i$ with receivers of data $j$ (in the i’s disseminating zone). A high value of $U_{col}$ means that $i$ is likely to become colocated with one or more receivers of $j$.

Let $n(t)$ be the set of the neighbors of $i$ at time $t$. $U_{col,i}$ at time $t$ is computed by comparing the members of $n(t)$ against those at the beginning of the previous period,

$$U_{col} = \frac{|n(t-T) \cup n(t)| - |n(t-T) \cap n(t)|}{|n(t-T) \cup n(t)|}. \hspace{1cm} (14)$$

Intuitively, the formula above yields the number of nodes which became neighbors or disappeared in the time interval $[t-T, t]$, normalized by the total number of nodes met in the same time interval. The computation of $U_{col}$ is much simpler: $U_{col} = 1$ if a $j$’s receiver is in the i’s disseminating zone, and $U_{col} = 0$, otherwise. The computation of delivery probability implies the assumption that the only information a node has about its position is logical connectivity information. In particular, we assume that a node is not aware of its absolute geographical location or of the location of those to whom it might deliver the data. Although this information could potentially be useful, it is currently unreasonable to assume the existence of GPS for all potential application domains for this technology.

The data disseminating process is as follows.

1. **Deterministic information is exploited, if available.** Upon receiving data $j$, node $i$ always sends the received data to all the receivers of data $j$ in its disseminating zone by utilizing the underlying synchronous routing mechanism.

2. **Probabilistic data disseminating is divided into two parts: forwarding and carrying.** Forwarding process makes the data spread to all the receivers in the same network partition quickly. The data $j$ is sent to the nodes in the disseminating zone of node $i$ whose delivery probabilities are larger than or equal to forwarding threshold $\sigma$. The value of $\sigma$ indicates the trade-off among resource consumption, coverage, and delay under certain network conditions. This forwarding process is complemented by a store-and-forward approach to deliver data across different network partitions by utilizing the intermittent connectivity. After the forwarding of data $j$ has occurred at node $i$, $j$ is also stored in $i$’s buffer if the delivery probability $P_{ij}$ is larger than or equal to carrying threshold $\tau$. If the carrier, while moving, meets a receiver, the data is transferred to the receiver. The optional rule is that when the carrier meets a node with a higher delivery probability, the data is transferred to the node with the higher delivery probability. With a proper $\tau$ value, data can be delivered to the nodes in the different network partitions in a bounded delay with reasonable overhead (i.e., buffer space and number of transmissions) with high probability.

To avoid duplicated data transferring, a forwarding node forwards the same data only once and discards the data already received. If the node has relayed the data to one destination using the deterministic information, the probabilistic forwarding to the same destination will be cancelled. The node never forwards the data into the node it came from. For the carrying nodes, the carried data is transferred only once to the same receivers or the same nodes with higher delivery probability. A first come first served (FCFS) buffer management scheme is utilized to manage the buffer space on the carrying nodes.

### 4.2. Replicating Nodes Election

In RHPMAN, a distributed replicating nodes election scheme is employed to find the nodes with rather stable neighboring topology and with enough resources such as energy and storage space acting as replicating nodes.

This is an initial setup stage for replicating nodes election when the network is constructed. A node may become a replicating node if it is found to be the most stable and powerful node among its neighborhood of $h$ hops. Otherwise, it is in client state. Every node calculates a hosting fitness value to
indicate the potentials of this node to be a replicating node. The hosting fitness is computed as

$$f = \frac{C}{M} = \frac{w_s S + w_p P + w_e E}{U_{calc}},$$

(15)

where $C$ is the capacity of node defined as the weighted sum of normalized storage space, energy, and process power and $M$ is the mobility of node defined as the change degree of connectivity of node. The node with larger capacity and less mobility has higher $f$ value.

All nodes broadcast their hosting fitness among their neighborhoods of $h$, hops. After an election period, $T_A$, each node compares its own hosting fitness value with those of its neighbors of $h$, hops. If a node has the highest value, it enters into the status of a replicating node. All nodes maintain a list containing the information of elected replicating nodes and periodically exchange the information about this list. Therefore, every node can find all electing replicating nodes with a bounded delay with high probability. In RHPMAN, semiprobabilistic data disseminating protocol regards the replicating nodes in this list as the receivers for replicas or updates.

This distributed replicating nodes election process can guarantee that there is at least one replicating node in every network partition. In large partition, there may be several replicating nodes every at least $h$, hops.

4.3. Replica Distribution. Replica distribution is responsible for the dissemination of replicas on the elected replicating nodes. The owner of the data initiates the replicating distributing process to disseminate the data to all the nodes in its replicating nodes list by employing the semiprobabilistic data disseminating protocol discussed in Section 4.1. Small $\sigma$ and $\tau$ values are preferred in this process to make the data spread to all the replicating nodes quickly.

The RHPMAN is a stateless replicating scheme. The data owner does not try to maintain the status of its replicas through some kind of data structures such as tables because maintaining the state information may cause too much overhead that is unacceptable in MANETs and because this information may not be accurate due to frequent partitions.

Figure 4 illustrates an example of partitioned MANET where semiprobabilistic data dissemination protocol is exploited to distribute replicas.

![Figure 4: An example of partitioned MANET where semiprobabilistic data dissemination protocol is exploited to distribute replicas.](image)

is larger than $\tau$, Q carries the data. When Q moves into the other partitions, it transfers the data to T through S.

4.4. Update Propagation and Consistency. There is a consistency issue in our replicating scheme. Due to impairments of wireless transmission and frequent network partitions, it is not practical to maintain strong consistency among replicas. In our scheme, a weak consistency model called eventual consistency model is adopted, which stipulates that all replicas eventually converge to the same state.

From an implementation point of view, the issues to solve in order to guarantee eventual consistency are (1) update propagation: each update must eventually reach all replicas and (2) update ordering: all updates must be eventually applied in the same order at each replica to ensure that the last update is the same at all replicas.

In RAPMAN scheme, we assume that only the owner of original copy could update the data items. The owner of original copy propagates the updates to all the replicating nodes in its replicating nodes list by utilizing semiprobabilistic data disseminating protocol. The $\sigma$ and $\tau$ values can be adjusted according to the urgent degree of updates and the network operational conditions. Since the semiprobabilistic data disseminating protocol is a best-effort protocol, the updates may reach the replicating nodes in different orders.

To solve this issue, the owner of original copy timestamps the updates by an incremental version number, whose original value is zero and will add one for every update. The replicating nodes apply the received updates in the ascending order of their version numbers. Since the updates with small version number may be delayed in the disseminating process and arrive later than the updates with large version numbers, the replicating nodes manage a small buffer where out-of-sequence updates are held temporarily awaiting the filling-in of gaps in the sequence when delayed updates turn up.

The node requesting the data saves the version number of the data got in the last successful data access. If the version number of the data got in the current data access is smaller than the saved value, the node knows that it gets stale data.
4.5. Replica Retrieval. Replica retrieval aims at enabling nodes to effectively find their requested data at replicating nodes. When a node wants to access a special data item, it first checks the availability of replicating nodes in its disseminating zone. If there is a replicating node in its disseminating zone, the node can access the data item through underlying synchronous routing mechanism directly.

If not, the node initiates a data lookup process by exploiting semiprobabilistic data disseminating protocol to disseminate data requests. The $r$ value is set larger than 1, which indicates that no node has the chance to carry the data requests. The data requests are disseminated only in the network partition where the requested node is. It is because our scheme ensures the availability of replicating nodes in most partitions. Spreading data requests across different partitions may cause unnecessary overhead. The $\sigma$ value is set equal to the delivery probability of the requesting node.

The nodes receiving the data requests check the availability of replicating nodes in their disseminating zone. Upon finding a replicating node, a data reply (including route reply) is sent back to the data requestor indicating the route to the replicating nodes. The retrieval process may return multiple replies. The requesting node only responds to the first reply to achieve minimal delay. The route between the data requestor and replicating node can be established through route accumulation.

The replica retrieval here is distinguished from standard flooding-based protocols by exploiting both deterministic and probabilistic information. The disseminating zones increase the probability that a node can get its data directly. More importantly, knowledge of the zone topology allows a node to efficiently continue the dissemination of a data request in the more likely case that destination can be found. In the example in Figure 4, all the nodes besides K can get the data directly using deterministic information. K can get the data by disseminating its request to nodes F and I. Since F has relayed the request to I, it does not forward the request. I finds the replicating node A and sends the reply back.

4.6. Replica Degree Maintenance. After the initial replica distribution, replica degree maintenance aims at maintaining the replication degree relatively stable by reacting to node exiting such as node failures and the fundamental change of node moving patterns that may degrade the performance of data access seriously.

When a replicating node realizes that it is going to exit the network (e.g., power-off), it can autonomously offload its shared data on the nodes with the highest $f$ value in its neighbors of $h$ hops and propagate the change of replicating nodes list to all the other nodes by utilizing semiprobabilistic dissemination protocol.

However, a replicating node does not succeed in foreseeing its exit from the network in many cases. If a node finds that it cannot access any replicating node for a certain period, it triggers a replicating node election process among the nodes in its neighbors of $h$ hops.

5. Performance Evaluation of RHPMAN

In this section, we compare our approach against others and assess the impact of protocol parameters through simulation.

5.1. Simulation Setting. We evaluated the performance of our protocol using the OPNET simulator. Our simulated network consists of 160 mobile nodes in a nonwrapping 1000 [m] × 1000 [m] square area.

Since RHPMAN focuses on the application scenarios where the mobile devices such as laptops, PDAs, and smart phones carried by humans construct MANETs by the needs of humans to socialize or cooperate, a community based mobility model [18] is utilized to describe the moving behaviors of mobile nodes. The simulation area is divided into a grid—4 × 4 in our experiments. The randomly chosen 8 nodes are then placed in one of these squares. Each node moves following the random walk model inside each square. The rest of the 32 nodes act as traveler nodes moving following the random walk model inside the whole simulation area to improve the connection opportunities. The radio communication range of each mobile node is a circle with the radius of $r$, 100 [m]. The node's speed $v$ is selected uniformly from the range of 1–10 [m/s] every 100 [s]. The speed of traveler nodes is set to 20 [m/s] to differentiate their roles in the simulation.

A 10% of the nodes are chosen randomly to be the owners of original data items. They are responsible for distributing the data items and corresponding updates to the elected replicating nodes. We assume that the replicating nodes have enough capacity to store all these data items. The updating interval follows exponential distribution with mean value, 120 [s]. Each node generates a single stream of data requests following a Poisson point process. The rate of request generating is 30 [s]. We also assume that the network topology remains unchanged during transferring one data item from one node to the other node synchronously.

The interval $T$, which determines the frequency of the nodes exchanging the profile information and computing delivery probability, is inversely proportional to the node speed, so networks with different mobility experience the same acceptable accuracy level of information. Here $T$ is set to 6 [s]. The weights for computing delivery probability are $w_{cdp} = w_{col} = 0.5$. The default values for semiprobabilistic data dissemination protocol are $\sigma = 0.4$, $\tau = 0.6$, $h = 2$, and $h = 4$.

The simulation time is set to 2400 s. No data was collected for the first 600 s of the simulations while the initial replicating nodes election process finished.

5.2. Simulation Results. In this section we present the results of our simulations. We mainly concentrate our analysis on the ratio of successful data access and network traffic (in terms of the number of transmissions).

5.2.1. Performance Comparison. To demonstrate the effectiveness of RHPMAN, we first compare it with the ideal approach, putting the $k$ replicas into the $k$ largest partitions, and the random approach, putting the $k$ replicas randomly.
The results are shown in Figure 5. When the number of nodes is small, the successful data access rates of these three approaches are all low since there are a lot of small partitions. As the number of nodes increases, the RHPMAN and ideal approach achieve higher successful data access rate than the random approach since larger partition emerges. RHPMAN is only 10%–15% lower than the ideal approach and has a 30% gain compared with the random approach. When the number of nodes reaches 200, the difference among these three approaches becomes smaller since the network is almost connected.

We also compare our scheme with the E-DAFN and E-DCG approaches proposed by Hara and Madria [9]. Both these approaches consider the data access pattern and replicate the data items with the higher access frequency with higher priority. To make the comparison fair, the data access frequency is distributed uniformly.

As shown in Figure 5, the result of E-DAFN is lower than the random approach since it only maintains one-hop neighbors’ information and guarantees only one replica distributed in the one-hop neighboring area. The replicas may not be distributed to a wide area. E-DCG constructs a biconnected component of the node hosting the original data and distributes one replica in this biconnected component. The performance is better than random approach. However, E-DCG relocates the replicas periodically and does not use the nodes with high mobility to deliver the replicas, which makes it difficult to disseminate the replica to the other partitions. This is the reason why the performance of E-DCG is lower than our approach. Constructing a biconnected component also needs to maintain the topological information of the whole network, which leads to a very high $h$ value. The network traffic is very high as we discuss in the following section about the impact of parameter $h$.

5.2.2. Semiprobabilistic Data Dissemination. We compare the performance of our scheme against the replication methods using the synchronous routing and pure probabilistic routing to disseminate replicas and updates. Since the number of replicating nodes is 20–25 in our simulation, we randomly choose 25 nodes as the replicating nodes in these two replicating methods and assume that the replica distribution has finished. As illustrated in Figures 6 and 7, RHPMAN outperforms these two methods in most cases. Synchronous routing (we use AODV) does not have the ability to bypass partitions and therefore leads to low successful data access ratio, 0.39. The network traffic is 291,020. With $\sigma = 0.4$ and $\tau = 0.6$, the successful data access ratio of RHPMAN reaches 0.79, an almost 100% improvement, with only 20% extra network traffic, from 291,020 to 351,078.

Figures 6 and 7 also show the performance results of the replicating method using probabilistic routing, in which each node forwards and stores the data at a certain probability ($1 - \sigma$ or $1 - \tau$ for the convenience of comparing), reforwarding the data when meeting new neighbors. This method may achieve high successful data access ratio at the cost of very heavy network traffic. Compared with RHPMAN, this method causes twice the network traffic to get the same successful data access ratio.

5.2.3. Impacts of Parameters $\sigma$ and $\tau$. The impacts of parameters $\sigma$ and $\tau$ on the performance are also shown in Figure 6 and Figure 7, respectively. As shown in Figure 6, if $\sigma$ is too large, our scheme fails to find appropriate subset of nodes to forward data, successful data access ratio is low, and network traffic is also low. If $\sigma$ is too small, each node forwards the data, which make unnecessary network traffic helpless to improve successful data access ratio. In our simulation, the value of $\sigma$ which leads to good trade-off between successful data access ratio and network traffic is around 0.4.

We can observe the same phenomena in Figure 7. $\tau$ should be set to a proper value to ensure that our scheme can find the appropriate subset of nodes to carry data across different network partitions. Small $\tau$ causes too much network traffic while large $\tau$ leads to low successful data access ratio. Under the simulated network conditions, $\tau$ should be around 0.6.

5.2.4. Impact of Parameter $h$. Parameter $h$ determines the scope of disseminating zone and directly controls the amount of deterministic information maintained in the network. When $h$ is small, the nodes only have very limited deterministic information to guide data lookup or update propagation, which leads to large transmission number. When $h$ is large, we have more deterministic information that can speed data disseminating. However, the network traffic caused by zone maintenance becomes very large, and the information updates may not arrive in each node in time, which leads to inaccurate zone information and degrades the performance. Figure 8 shows that $h$ should be set to 2 in the simulated network. Under this circumstance, the effect of the increased deterministic information on reducing extra forwarding is more significant than the effect of increasing network traffic.
Figure 6: Successful data access ratio and network traffic versus forwarding threshold $\sigma$.

Figure 7: Successful data access ratio and network traffic versus carrying threshold $\tau$.

Figure 8: Successful data access ratio and network traffic versus disseminating radius $h$. 
5.2.5. Update Propagating Delay. As shown in Figure 9, update propagating delay is more sensible to parameter $\tau$. If $\tau$ is too large, our scheme fails to find appropriate subset of nodes to carry the data across different partitions; update propagating delay will be very large since forwarding process can only spread the data in the same partitions no matter what $\sigma$ is. On the contrary, with appropriate $\tau$, our scheme always can find some suitable nodes carrying the data and disseminating the data to the replicating nodes. Of course, large $\sigma$ makes the data spread to the carrying nodes fast.

The knowledge about updating propagation delay is helpful to manage the buffer space on the carrying nodes. For example, we can discard the received data after a certain period to recycle buffer space. The length of period can be determined by exploiting the knowledge of the updating propagation delay that makes the probability of the data needed to be retransferred in the future very low. Our scheme uses a simpler policy, FCFS, which still tries to lower the probability of the data needed to be retransferred in the future. The knowledge about updating propagation delay is also helpful to determine the size of the buffer temporarily storing out-of-sequence updates on the replicating nodes. Given the updating propagation delay $T_{ud}$ and updating interval $T_{ui}$, the size of buffer $b$ can be computed as $b = \text{MAX}(T_{ud})/\text{MIN}(T_{ui})$. Considering the uncertainty of these values, a more complicated formula incorporated with probability distribution may be also obtained. However, the discussion of this formula exceeds the scope of this paper.

5.2.6. Impact of Mobility. Thus far we discussed the effect on performance of protocol parameters. Now, instead, we vary some scenario parameters. The first one is mobility model. Our approach relies on carriers that are chosen for their relative mobility and colocation with replicating nodes. In other words, disseminated data are stored only on potential carriers. Using homogeneous random walk mobility model where no traveler node exists, the nodes have the same value of the change degree of connectivity, which makes identifying potential forwarders and carriers difficult. Successful data access ratio decreases. With a large $\tau$, we may not find any carrying nodes, which leads to a very low successful data access ratio as shown in Figure 10(a). In this case, lower $\sigma$ and $\tau$ are preferred.

We also vary the number of traveler nodes in the network. As already pointed out, travelers are key to disseminate data in remote portions of the network and a peculiarity of our approach is the ability to identify these special nodes. As shown in Figure 10, successful data access ratio increases, and network traffic also increases a little since there are more carrying nodes. In this case, higher $\sigma$ and $\tau$ are preferred since the change degree of connectivity increased due to more travelers.

6. Conclusion and Future Direction

In this paper, a novel replication scheme, RHPMAN, is proposed to improve the performance of data access in highly partitioned MANETs, which employs a semiprobabilistic data disseminating protocol to distribute the replicas and propagate the updates despite network partitions and intermittent connectivity. A weak consistency model, eventual consistency model, is utilized to ensure that all updates eventually propagate to all replicas in a finite delay. Simulation results demonstrate that RHPMAN scheme leads to good successful data access ratios and latencies with small overheads.

Future work will address the inclusion of additional profile information and corresponding analyzing techniques in identifying potential carriers, for example, residual energy in the profile, using some kind of prediction technique to predict the colocation. Buffer management policy for utilizing the limited buffer on the carrying node will be also studied.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.
**Figure 10:** The impact of mobility.

**Acknowledgment**

This work was supported in part by the Specialized Research Fund for the Doctoral Program of Higher Education under Grant no. 20110142110062.

**References**

[1] Homepage of IETF mobile ad hoc networks (MANET) working group, 2013, http://www.ietf.org/html.charters/manet-charter.html.

[2] D.-W. Sun, G.-R. Chang, S. Gao, L.-Z. Jin, and X.-W. Wang, “Modeling a dynamic data replication strategy to increase system availability in cloud computing environments,” *Journal of Computer Science and Technology*, vol. 27, no. 2, pp. 256–272, 2012.

[3] D. A. Tran, K. Nguyen, and C. Pham, “S-CLONE: socially-aware data replication for social networks,” *Computer Networks*, vol. 56, no. 7, pp. 2001–2013, 2012.

[4] M.-C. Lee, F.-Y. Leu, and Y.-P. Chen, “PFRF: an adaptive data replication algorithm based on star-topology data grids,” *Future Generation Computer Systems*, vol. 28, no. 7, pp. 1045–1057, 2012.

[5] E. Kayaaslan, B. B. Cambazoglu, and C. Aykanat, “Document replication strategies for geographically distributed web search engines,” *Information Processing and Management*, vol. 49, no. 1, pp. 51–66, 2013.

[6] P. Bellavista, A. Corradi, M. Fanelli, and L. Foschini, “A survey of context data distribution for mobile ubiquitous systems,” *ACM Computing Surveys*, vol. 44, no. 4, article 24, 2012.

[7] C.-A. La, P. Michiardi, C. Casetti, C.-F. Chiasserini, and M. Fiore, “Content replication in mobile networks,” *IEEE Journal on Selected Areas in Communications*, vol. 30, no. 9, pp. 1762–1770, 2012.

[8] T. Hara, “Quantifying impact of mobility on data availability in mobile Ad Hoc networks,” *IEEE Transactions on Mobile Computing*, vol. 9, no. 2, pp. 241–258, 2010.

[9] T. Hara and S. Madria, “Data replication for improving data accessibility in Ad Hoc networks,” *IEEE Transactions on Mobile Computing*, vol. 5, no. 11, pp. 1515–1532, 2006.
[10] T. Hara and S. K. Madria, “Consistency management strategies for data replication in mobile ad hoc networks,” *IEEE Transactions on Mobile Computing*, vol. 8, no. 7, pp. 950–967, 2009.

[11] K. H. Wang and B. Li, “Efficient and guaranteed service coverage in partitionable mobile ad-hoc networks,” in *Proceedings of the IEEE 21st Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM ’02)*, pp. 1089–1098, June 2002.

[12] J.-L. Huang, M.-S. Chen, and W.-C. Peng, “Exploring group mobility for replica data allocation in a mobile environment,” in *Proceedings of the 12th ACM International Conference on Information and Knowledge Management*, pp. 161–168, New Orleans, La, USA, November 2003.

[13] A. Derhab and N. Badache, “Data replication protocols for mobile ad-hoc networks: a survey and taxonomy,” *IEEE Communications Surveys and Tutorials*, vol. 11, no. 2, pp. 33–51, 2009.

[14] A. Derhab, N. Badache, and A. Bouabdellah, “A partition prediction algorithm for service replication in mobile Ad Hoc networks,” in *Proceedings of the 2nd Annual Conference on Wireless On-demand Network Systems and Services (WONS ’05)*, St. Moritz, Switzerland, January 2005.

[15] S. Kapadia, B. Krishnamachari, and L. Zhang, “Data delivery in delay tolerant networks: a survey,” in *Mobile Ad-Hoc Networks: Protocol Design*, X. Wang, Ed., chapter 26, 2011.

[16] X. Zhuo, Q. Li, W. Gao, G. Cao, and Y. Dai, “Contact duration aware data replication in delay tolerant networks,” in *Proceedings of the 19th IEEE International Conference on Network Protocols (ICNP ’11)*, pp. 236–245, October 2011.

[17] J.-Y. le Boudec and M. Vojnović, “Perfect simulation and stationarity of a class of mobility models,” in *Proceedings of the IEEE Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM ’05)*, pp. 2743–2754, Miami, Fla, USA, March 2005.

[18] M. Musolesi and C. Mascolo, “A community based mobility model for Ad Hoc network research,” CS-UCL Research Note RN/05/31, Department of Computer Science, University College London, 2005.
