

Motion-MiX DHT for Wireless Mobile Networks

Seungjae Shin, Uichin Lee, *Member, IEEE*,
Falko Dressler, *Senior Member, IEEE*, and Hyunsoo Yoon, *Member, IEEE*

Abstract—Last encounter routing (LER) is an excellent routing paradigm that exploits distributed mobility diffusion to achieve both moving object tracking and packet networking services in dynamic mobile networks. From our observations, we discover that LER can be easily extended to a mobile DHT protocol that introduces excellent performance to high-speed mobility environments. This is of particular interest when higher level of mobility and membership dynamics go hand in hand. In our simple but powerful DHT paradigm, a data publish/look-up process consists of a sequence of spatial *motion tracking* of the rendezvous node that is responsible for the data resource. Thus, we name the protocol MX-DHT (Motion-MiX-DHT). As opposed to existing topology based DHT schemes, MX-DHT does not require additional management of logical or overlying look-up topologies, except for the one-hop encounter records of logical metadata carried by mobile nodes. Therefore, in high-speed mobility and dynamic membership environments, MX-DHT achieves a significant reduction in the communication costs of the publish/look-up and join/leave operations as compared to existing mobile DHT schemes. An extensive set of experiments showed that MX-DHT is a cost-effective solution to providing a content centric networking service in different types of networks with dynamic mobility and membership changes.

Index Terms—Wireless mobile network, ad hoc network, high-speed mobility, dynamic membership change, location service, last encounter routing, distributed hash table, content centric network

1 INTRODUCTION

As content centric networking (CCN) has emerged as an effective solution to the recent traffic explosion problem [1], [2], [3], the distributed hash table (DHT) functionality has now become a significantly more important requirement in the wireless network domain [4], [5]. Especially, because of its autonomous configuration capability [6], mobile DHT can be an attractive CCN framework to offload the concerns of economics, security, and the limited capacity of infrared access. Almost all existing DHT schemes realize publish/look-up rendezvous protocols by uniformly mapping data resources into logical indexing topologies, such as a tree, ring, or coordinate space. In these schemes, node mobility naturally causes a change in the logical indexing topology, which requires communication overheads for topology reconfiguration. Therefore, existing topology-based DHT protocols suffer from a considerable communication burden in high-speed mobility environments.

In our research, we identified that a simple and elegant combination of last encounter routing (LER) [7], [8] and DHT saves us from the large communication burden in high-speed mobility environments. This LER-based mobile DHT leverages *mobility assisted milestone sharing* to reduce the communication cost required for publish/look-up and membership management. In our approach, none of the

nodes is required to manage any logical or network-related topology information, except for the single-hop neighbor lists. Furthermore, each node maintains a memory structure named the extended last encounter table (E-LET) to record the times and locations at which it encounters logical meta-data information carried by mobile nodes. By exploiting these location and time tuples as *milestones*, the LER-based DHT provides an efficient motion tracking-based publish/look-up service for various types of data resources. While the topology reconfiguration overheads of existing mobile DHT protocols for each member join/leave event are substantial, the LER-based DHT does not need topology-related control traffic. This is because motion tracking-based rendezvous is not influenced by who is the carrier of meta-data. If the motion of logical meta-data can be tracked by its continuous trail over time, we can guarantee successful publish/look-up, regardless of whether the meta-data is involved with a member join/leave event or not. In other words, no explicit notification is required for every member join/leave event provided that motion continuity is preserved. Therefore, in order to ensure the motion continuity of logical meta-data, the LER-based DHT operates such that every member join/leave operation is conducted within the *motion-mix* zone where different motions meet together. Thus, we named our LER-based DHT solution Motion-MiX DHT, or MX-DHT for short.

Our analytical and experimental evaluations show that MX-DHT achieves excellent communication efficiency and publish/look-up reliability in high-speed mobility and dynamic membership environments. We expect MX-DHT to become a candidate solution for various CCN services in future mobile networks. Our work can be summarized as follows:

- We present mobile DHT usage scenarios and related work (Section 2) and review the LER concept, which

• S. Shin and H. Yoon are with the School of Computing, Korea Advanced Institute of Science and Technology, Daejeon 305-338, Korea. E-mail: {sjshin, hyoon}@nslab.kaist.ac.kr.

• U. Lee is with the Department of Knowledge Service Engineering, Korea Advanced Institute of Science and Technology, Daejeon 305-338, Korea. E-mail: ucllee@kaist.edu.

• F. Dressler is with the Department of Computer Science, University of Paderborn, Germany. E-mail: dressler@ccs-labs.org.

Manuscript received 16 June 2015; revised 13 Jan. 2016; accepted 19 Jan. 2016. Date of publication 3 Feb. 2016; date of current version 31 Oct. 2016.

For information on obtaining reprints of this article, please send e-mail to: reprints@ieee.org, and reference the Digital Object Identifier below. Digital Object Identifier no. 10.1109/TMC.2016.2524575

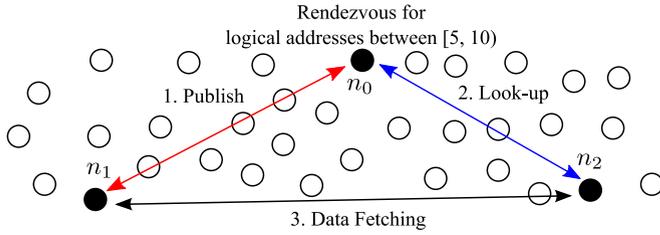


Fig. 1. Mobile DHT concept: Rendezvous network for meta directory service.

particularly exploits node mobility for setting up routing paths in mobile ad hoc networks (see Section 3).

- We propose MX-DHT, an LER-based mobile CCN solution that achieves excellent communication efficiency while maintaining plausible reliability under high-speed mobility and membership dynamics. Especially, the publish/look-up robustness of MX-DHT is also investigated (see Section 4).
- We present an analytical assessment to estimate the communication costs of MX-DHT based on the previous probabilistic evaluation framework of LER [9]. Asymptotic costs are also discussed (see Section 5).
- Through an extensive set of experiments covering different types of networks and different mobility patterns, we evaluate the performance of MX-DHT in comparison to a variety of related solutions, including proactive and reactive flooding, virtual ring routing, virtual cord protocol, and geographic hash tables (see Section 6). Our results clearly demonstrate the outstanding performance of MX-DHT.

2 BACKGROUND AND RELATED WORK

2.1 Mobile DHT Usage: Scenarios and Model

We consider a distributed storage model where a mobile DHT is used to realize a directory service (or meta-data store) as in conventional DHT usage models for P2P file sharing (i.e., storing resource locations throughout the network). We envision three application scenarios of DHTs for publishing and locating various kinds of data in mobile environments (e.g., dashcam video clips, terrain map data, and location-based services).

① *Sharing sensor data via Internet of vehicles.* Vehicles can share sensed data with one another to provide intelligent driver support services (e.g., situation awareness by sharing dashcam video clips, or disseminating real-time road surface conditions to nearby vehicles) [10]. DHT helps vehicles to efficiently locate and access these kinds of sensed data on a mobile vehicular cloud on an on-demand basis [11].

② *Robust mobile content store for urban warfare.* We envision an urban warfare scenario where a group of mobile nodes (e.g., soldiers and armored vehicles equipped with wireless links) share various kinds of situation awareness data. A decentralized content store allows users to reliably distribute mission critical content among mobile nodes (e.g., terrain map, blueprints). This model is applicable to other information-sharing scenarios such as disaster recovery where cellular infrastructure is not available, perhaps because of a power outage or infrastructure failure.

③ *Location based look-up services for opportunistic cooperation.* Mobile DHT can be applied to enable a location-based

look-up service that allows mobile users to opportunistically cooperate with a group of other users with the same interest (e.g., receiving special discount coupons). This mechanism can be used to organize location-based crowdsourcing events [6].

Fig. 1 illustrates our storage model where n_0 is a rendezvous node of resources whose logical address is in the range [5, 10]. If n_1 has a data resource c to publish, it first calculates the logical address, say 7.3 (e.g., by hashing the uniform resource identifier of c), and then publishes the meta-data of c and its physical address to node n_0 . This phase is referred to as the *publish* phase. Later, when node n_2 wants to access c , the node uses the logical address of c to send a query to n_0 for obtaining the physical address of the node that owns the data resource c ; in this case, it is node n_1 . This phase is referred to as the *look-up* phase. After obtaining the physical address of n_1 , n_2 proceeds with fetching the data resource c via conventional routing mechanisms.

2.2 Existing Mobile DHT Schemes

We categorize existing mobile DHT schemes into the following four groups:

Flooding. The most naive approach for wireless DHT is flooding [12]. Flooding is divided into two types: proactive and reactive. In proactive flooding, each node periodically floods the advertisement notifying every other node of the logical addresses for which it is responsible. Hence, every node in the network keeps an almost up-to-date logical-physical address mapping table so that the look-up process is identical to conventional routing processes. Reactive flooding is an AODV-like version of a mobile DHT. If a node wants to involve the rendezvous of a specific data resource, it floods the look-up request message all over the network in order to discover the path toward the rendezvous node. When the request message arrives at the rendezvous node, the path is established. According to [12], although flooding is naive and easy to implement, the communication networks become (over-)loaded, as the network size and look-up request rate increase.

Overlay concepts. In some studies, the logical address is expressed as a topological or geographical index. L+ [13] and TRIBE [14] use a hierarchical tree as an indexing topology, where routing proceeds by repeated tree traversal. These schemes are more compact and bandwidth-efficient solutions than naive overlay techniques. Based on L+ and TRIBE, DART [15], EMP [16], and ATR [17] provide improved indexing topology management exploiting tree-balancing and path diversity gain. When geographical forwarding is possible, GHT, which maps data resources to multi-dimensional geographic coordinates, is an excellent alternative [18], [19], [20].

DHT-inspired routing. A more evolved approach is to combine a logical indexing topology with the routing protocol. By this integration, the routing layer naturally provides conventional source-destination communication as well as a publish/look-up service. Caesar et al. proposed VRR [21] as a novel integration of proactive routing and a logical ring space concept so that the DHT operation is well embedded in the routing process. While the conventional proactive routing has $O(N^2)$ (N is the network size) growth of the routing table, VRR reduces it to $O(1)$ by establishing only

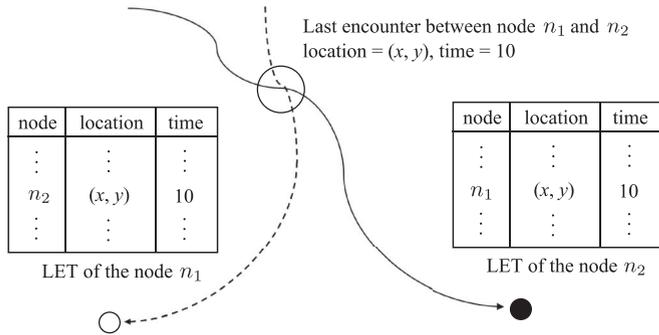


Fig. 2. Mobility diffusion of LER.

the paths with a constant number of neighbors on the ring space topology. Publish/look-up is performed by repeated jumps on the ring space. Scalable source routing (SSR) [22] integrates the ring space concept into dynamic source routing. As in VRR, it adopts incremental traversals on a virtual ring for the publish/look-up process. VRR and SSR show excellent scalability and handle membership dynamics well. However, they still suffer from the path-stretch problem, i.e., they tend to use rather long paths through the network.

DHT-inspired routing with topology awareness. In order to solve the path stretch problem, in recent studies physical topology proximities of nodes for logical address assignment were considered. Awad et al. presented VCP [23] using an advanced ring structure called a virtual cord. VCP ensures that, if the two nodes are one-hop neighbors of each other on the virtual cord, they should be also physical one-hop neighbors. Because of this topology awareness, the presence of a virtual cord between two nodes implies their physical adjacency. Hence, by assigning (removing) a virtual cord to (from) appropriate pairs of nodes, VCP provides high-quality DHT topology in terms of path length. Jain et al. proposed VIRO [24], which associates the logical ring space and hierarchical tree so that the node is ordered by location proximity on the ring space. Mesh-DHT [25] and 3D-RP [26] integrate multi-dimensional coordinates into the routing protocol while providing topology awareness. By deploying logical node objects on the multi-dimensional vector space with a similar distance interval, they provide more uniform distribution of data resources, as well as improved path diversity, while managing a reasonable correspondence between physical and logical topology proximity.

Note that despite above advances of mobile DHT, existing approaches cannot effectively deal with high-speed mobility and membership churning, since they still suffer from large overheads for reconfiguring logical indexing topologies triggered by high-speed mobility and frequent join/leave events. Our goal is to design a solution that can effectively mitigate this problem. Unlike existing DHT protocols that explicitly maintain end-to-end paths among logical neighbors for resource publish/lookup, we exploit motion traces to help mobile nodes to locate metadata by tracking back their encounter histories.

3 REVIEW OF LAST ENCOUNTER ROUTING

We consider a wireless network environment where the node mobility causes continuous changes in the physical network topology. Instead of explicitly maintaining any

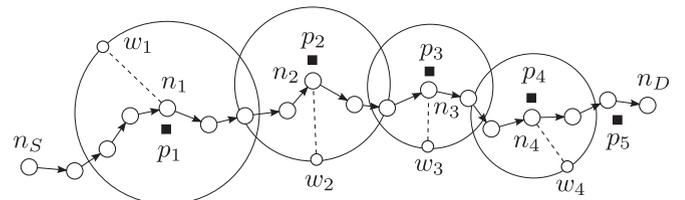


Fig. 3. Routing process of LER.

path, LER exploits on-demand path discovery based on the history records of one-hop encounters. In LER, each node maintains a last encounter table (LET) to record the times and locations at which it encounters every other node in the network. Here, the term *encounter* means that two nodes are within a one-hop communication range. Thus, the LET can be easily maintained by simply monitoring periodic hello messages exchanged by the layer-two or three protocols.

Fig. 2 shows how two nodes update their LET during an encounter. As n_1 and n_2 move within the geographical area, an encounter can be observed and both nodes update their LET with the (location, time)-tuples for n_i . These encounters can be used as milestones to approximate the movement trail of the other nodes. In other words, the mobility of n_i is recorded in the network in a distributed manner. This mechanism is referred to as *mobility diffusion* [7].

The last encounter records distributed by *mobility diffusion* play the role of milestones for *piece-meal-wise* path discoveries in an on-demand routing process. Fig. 3 illustrates the routing of a packet from n_S to n_D . First, n_S searches its LET, and finds out that it encountered n_D at point p_1 at time t_S . Thereafter, it sends the packet toward p_1 . When n_1 , the closest node to p_1 , receives the packet, it locally floods the search message to find a more recent witness of n_D . Then, the new witness w_1 directs it to the more recent encounter point p_2 , where it meets n_D at $t_1 (> t_S)$. When the packet is delivered to n_2 near p_2 , n_2 again starts local flooding so that the w_2 reports a new milestone to the next tracking point p_3 . Eventually, the local flooding and forwarding are repeated until the packet is routed to n_D .¹

This is similar to AODV routing [27]. However, there is an important difference. AODV requires network-wide flooding for path discovery, because it uses a single discovery-and-forward strategy. On the other hand, LER takes a *piece-meal-wise* discovery-and-forward strategy, where not only the source but also several intermediate relays emit a search message to find a relay closer to the destination. Nicely, each intermediate search message is locally flooded to a small area that is in fact sufficient to discover milestones for each next path. In particular, the higher the node mobility, the smaller the local flooding area can be by virtue of the *mobility capacity effect*, where the encounter-probability of an arbitrary node pair is increased as the node mobility increases [28], [29]. As illustrated in Fig. 4, LER achieves a substantial reduction in communication costs as compared to AODV because of the incremental path discovery with small area flooding.

1. For convenience, our example uses a geographic LER based on (coordinate, time)-milestones. However, LER can also be implemented well with a non-geographic approach using (witness, time)-milestones [8].

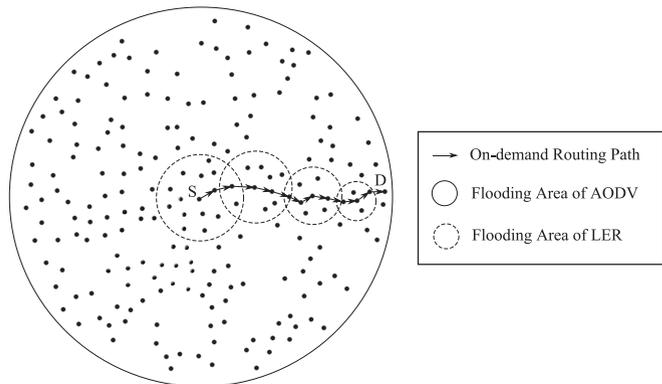


Fig. 4. Comparison of route discovery overheads: LER and AODV.

By decomposing the single path discovery process into several incremental discovery sub-processes, LER takes a novel advantage of the *mobility capacity effect* to realize resource-efficient on-demand routing where nodes are moving with high-speed mobility. According to previous studies, the routing path established by LER is asymptotically bound to the optimal [7], [9], [30]. It is recognized as a powerful layer-3 protocol for ubiquitous services in human or vehicular-carried device networks [31], [32], [33]. Moreover, LER is also one possible routing solution for a delay tolerant network (DTN) because it can also reap the benefit of *mobility-assisted data delivery*. The basic strategy of DTN is known as *carry-and-forward* where an intermediate relay carries the packet until it encounters the destination. Several works used the incremental *search-and-forward* strategy of LER in DTN routing to improve end-to-end responsiveness [34], [35], [36].

4 MOTION-MIX-DHT

4.1 Basic Concept: DHT-Capable Extension of LER

We augmented LER to support DHT functionality by extending the LET, as depicted in Fig. 5. In the figure, each node has a new single-column table, which stores the *logical address interval* (LAI) associated with the original LET. The logical address space is the same hash-based ring space as in existing DHTs [21], [22], [23], [24]. That is, the logical address of a data resource is generated by applying the hash function to the identifier of the data resource (e.g., URI). While existing DHT schemes make one-to-one correspondences between nodes and LAIs, in our LER-based DHT each node can take several LAIs, which do not have to be adjacent to each other.

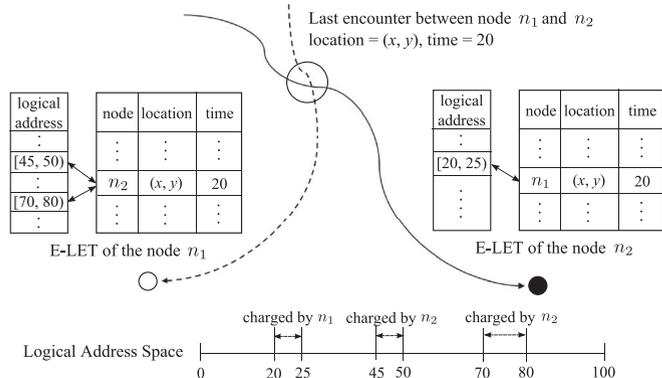


Fig. 5. Toward MX-DHT: DHT-capable LER.

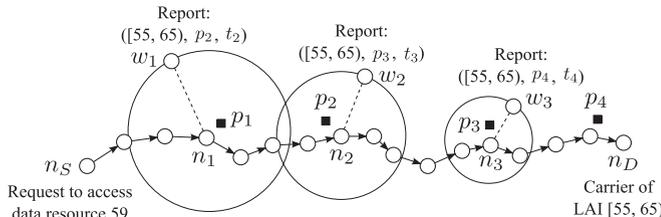


Fig. 6. Look-up of DHT-capable LER.

In our example, node n_1 is responsible for the meta-data whose logical address is within LAI [20, 25] (cf., Fig. 5). In other words, n_1 is the rendezvous point for the data resources whose logical address is equal to or greater than 20 and less than 25. Similarly, n_2 maintains two address intervals, [45, 50] and [70, 80]. When n_1 and n_2 encounter each other at location (x, y) , each node updates its extended LET (E-LET), as indicated in the figure. In particular, since n_2 maintains two LAIs, n_1 makes two separate associations of the $[n_2, (x, y), 20]$ tuple, with [45, 50] and [70, 80], in its E-LET. Accordingly, in this extended LER, the node diffuses the mobility of LAIs for which it is responsible, as well as its own mobility. To implement such an augmented diffusion, we have only to add an LAI indication field to the hello message exchanged between one-hop neighbor nodes.²

The look-up (publish) of DHT-capable LER is similar to the routing of the original LER.³ Fig. 6 illustrates an example of a data look-up where node n_S wants to look-up the data resource whose logical address is 59. Initially, n_S searches its E-LET, and then, finds the record that gives the information that it encountered the LAI [55, 65] carried by n_D at location p_1 at time t_1 . Then, n_S sends the look-up request message toward p_1 . When the request message arrives at n_1 , which is the node closest to p_1 , n_1 starts local flood searching to query more recent witness nodes for the logical address 59. w_1 reports it encountered [55, 65] at p_2 at t_2 . Thereafter, the request message is forwarded toward p_2 so that it is delivered to n_2 . Upon receiving the message, n_2 also performs local flood searching, and then, discovers the next tracking point p_3 reported by w_2 . As this *piece-meal-wise* path discovery and forwarding is repeated, the look-up request message approaches n_D , the rendezvous place of the targeted data resource. The complete algorithm for the data look-up protocol of DHT-capable LER is described in the form of pseudo-code in Algorithm 1.

The routing feasibility of LER strongly depends on whether the witness nodes provide encounter points that are not continuous but sufficiently close to each other. Provided that this *approximate motion continuity* is satisfied, LER is effective, because the motion of each node is well approximated with sufficient accuracy. In other words, LER exploits the sequence of encounter reports to approximate

2. Similar to the previous section, we assume that MX-DHT uses (LAI, coordinate, time)-milestones in order to adopt the geographic forwarding [7], [37]. Of course, MX-DHT can be also easily extended from non-geographic LER [8]. In such an extension, we need to use (LAI, witness, time)-milestones.

3. In MX-DHT, the *publish* and *look-up* processes are conceptually same as each other; a message exchange between data owner (or customer) and rendezvous. Therefore, we only explain the *look-up* process throughout this section for simplicity.

the continuous motion of the destination. A comparison of Figs. 3 and 6 shows that this property is well inherited by the DHT-capable extension. The only difference is that the data look-up tracks the motion of the LAIs, as well as the node. This means that the feasibility of the original LER implies the feasibility of DHT-capable LER.

Algorithm 1. Data Look-up Protocol

Notations:

t_C : current time, n_S : source node.

c : the logical address of the data resource n_S wants to access.

m : the content request message starting from n_S .

$(p_{[\alpha,\beta],n}, t_{[\alpha,\beta],n})$: the tuple of location and time at which the node n last encountered the LAI $[\alpha, \beta]$.

- 1: $n = n_S$.
 - 2: $(p, t) = (p_{[\alpha,\beta],n_S}, t_{[\alpha,\beta],n_S})$ such that $\alpha \leq c < \beta$.
 - 3: Forward m to p .
 - 4: n = the node where m is reached by forwarding.
 - 5: **while** n is not responsible for c **do**
 - 6: Search more recent witness node of c by local flooding.
 - 7: Let W be the set of witness node w such that
 - 8: $t \leq t_{[\alpha_w,\beta_w],w}$ and $\alpha_w \leq c < \beta_w$.
 - 9: **if** $W \neq \emptyset$ **then**
 - 10: $(p, t) = (p_{[\alpha_w,\beta_w],w}, t_{[\alpha_w,\beta_w],w})$ such that
 - 11: $\arg \min_{w \in W} t_C - t_{[\alpha_w,\beta_w],w}$.
 - 12: **else**
 - 13: Search wider flood area until W is not empty.
 - 14: Go to line 9.
 - 15: **end if**
 - 16: Forward m to p .
 - 17: n = the node to which m is delivered by forwarding.
 - 18: **end while**
-

4.2 Membership Management Protocols

In a DHT network, LAIs of data resources are distributed throughout the network. The membership management protocol is necessary for re-assigning LAIs for each node join/leave event. Almost all conventional DHT protocols require that the joining (leaving) node receive (give) the LAI information from (to) its neighbor (predecessor or successor) node on the logical indexing topology. In other words, for every join/leave event, the donor or recipient of the LAI should be a neighbor on the logical ring or coordinate spaces. To maintain the logical topologies, existing DHT approaches pay significant management overheads, in particular in dynamic membership and high mobility scenarios.

In contrast, our MX-DHT is no longer tied to these logical indexing topologies. Instead, it passively records the approximate motions of nodes and their LAIs that virtually float on the geographic map. In our LER-based DHT, for each join/leave event, a donor (recipient) node is selected from the physical one-hop neighbors of the joining (leaving) node. This is to maintain *approximate motion continuity*, which is important for the data look-up feasibility. If the LAI transfer is conducted within the one-hop communication range, the approximate motion continuity of the transferred LAI is not broken. Here, when node n_1 transfers the LAI $[\alpha_1, \beta_1]$ to its physical one-hop neighbor n_2 , which originally has the LAI $[\alpha_2, \beta_2]$ at time t , their motions are merged to a single motion so after that both the LAIs move on the same trail. This

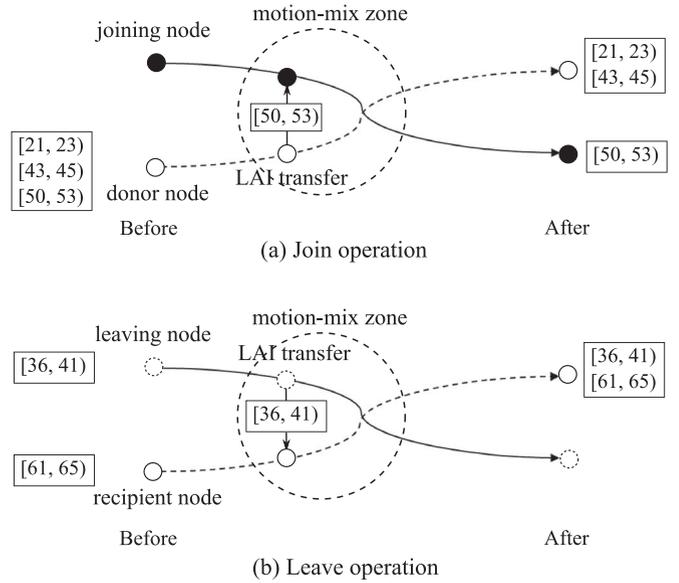


Fig. 7. Join/leave operations of MX-DHT.

feature is referred to as the *motion-mix* property [38]. For this reason, for each membership change joining or leaving nodes select any node as a donor or recipient for the LAI relocation from within the so-called *motion-mix zone*, i.e., one-hop communication range, or encounter zone. Therefore, we call our scheme *Motion-MiX-DHT*. In MX-DHT, since every membership management operation consists of simply passing LAI information from a donor to a recipient within the motion-mix zone, no further control and notification are required.

Node join protocol. When a new node joins the network, the joining node monitors the hello messages exchanged by the physical neighbor nodes within the motion-mix zone, i.e., the one-hop communication range. Then, it sends a join request message to the neighbor node that has the largest amount of LAIs. The joining node becomes the LAI recipient, and the node receiving the join request message becomes the LAI donor. The donor transfers a part of its LAIs to the joining node. If it has a single LAI, the LAI is divided into two fragments, and then, one of these is given to the joining node. Fig. 7a shows an example of the node join operation. The joining node receives the LAI [50, 53] from the donor node. Note that LAI [50, 53] maintains *approximate motion continuity*.

Node leave protocol. When an existing node leaves the network, it sends the leave request message to the node that has the smallest amount of LAIs among those within the motion-mix zone. In this case, unlike in the join operation, the leaving node becomes the LAI donor and the recipient takes the LAIs from the donor. Fig. 7b shows an example of the node leave protocol.

4.3 Look-Up Robustness under LAI—Node Inconsistency

In conventional DHT schemes, when a membership change occurs, the system has to notify this change to the neighbors on the logical address space. This is because the join/leave event changes the correspondence LAI—node, which has to be consistently identified for accurate data look-up. Furthermore, the signaling overheads become large if the physical

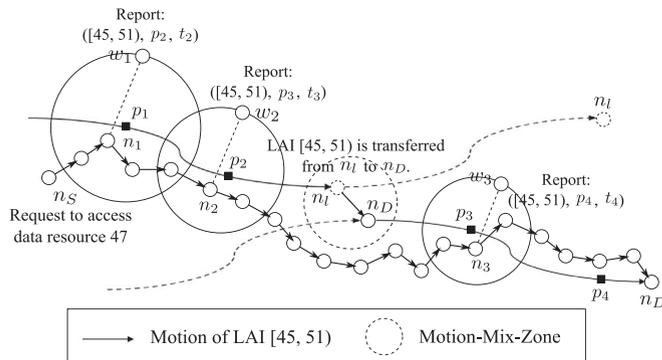


Fig. 8. Data look-up under LAI—node inconsistency.

path distance between two logical neighbors is stretched. Now, MX-DHT does not have to notify any other node of this change, except for the LAI donor and recipient. Even if the LAI-node consistency is destroyed temporarily, it is automatically fixed during the data look-up process, provided that the trail of the relocated LAI maintains *approximate motion continuity*.

Fig. 8 illustrates the manner in which the data look-up remains still effective under the LAI—node inconsistency. In the figure, the source node n_S wants to look-up the data resource whose logical address is 47. Initially, n_S searches its E-LET and then finds out that it encountered the LAI [45, 51] carried by n_i at location p_1 and time t_1 . However, the current rendezvous of [45, 51] is node n_D , because n_i leaves the network between time t_2 and t_3 . That is, n_S does not know about this change of LAI-node correspondence. Anyhow, n_S sends the look-up request message toward p_1 . When the request message arrives at node n_1 at around p_1 , n_1 performs local flood searching and then obtains the next tracking point p_2 from witness w_1 . The message is then delivered to n_2 at around p_2 . When n_2 performs local flood searching, it receives the report from w_2 , which encountered n_D , the new carrier of [45, 51], at p_3 at t_3 . Hereafter, the LAI—node correspondence is automatically updated, and the look-up proceeds well according to the motion of [45, 51] diffused by n_D . The motion of LAI [45, 51] follows the motion of n_i until the node leaves, and thereafter, follows the motion of n_D . This example shows that LAI—node inconsistency does not damage the look-up process. The inconsistency is even automatically corrected as the look-up proceeds.

For every membership change, provided that the LAI transfer is performed within the motion-mix zone, the *approximate motion continuity* of the relocated LAI is maintained so that the look-up remains effective. Consequently, the look-up process of MX-DHT is influenced not by the membership change itself, but by whether the motion of the rendezvous for the data resource is tracked well or not.

Theorem 1. *In an MX-DHT network, if the meta-information of a data resource is relocated because of a join or leave operation, data look-up is still possible without additional notification.*

Sketch of Proof This can be proved in two steps. The first step is to define a look-up process model to generalize the example in Fig. 6. The second step is to construct another look-up process model to generalize the example in Fig. 8 by inserting into the first model some additional assumptions about LAI transfer. By showing that the look-up

processes and costs of the two models are exactly same, we can validate our claim. Rather than a full proof, we present only a sketch of the proof to show the rationale behind it.

Step 1. Basic model construction. Suppose a node n_S wants to look-up the resource whose logical address is γ and its meta-data is carried by node n_D , which moves along the trajectory f ; further, assume n_D maintains $[\alpha_0, \beta_0)$ ($\alpha_0 \leq \gamma < \beta_0$). According to the look-up protocol of MX-DHT, the look-up message must be delivered to n_D by repeating the searching-and-forward process k times ($k \geq 0$). Here, let us denote the relays and witnesses that are related to local flood searching by $R = \{r_1, r_2, \dots, r_k\}$ and $W = \{w_1, w_2, \dots, w_k\}$, respectively. From the above formulation, we can express the look-up cost of MX-DHT as

$$\sum_{i=1}^k c(r_i, w_i) + d(n_S, r_1) + \sum_{i=1}^{k-1} d(r_i, r_{i+1}) + d(r_k, n_D), \quad (1)$$

where $c(r, w)$ is the cost required to discover the witness w through local flood searching from r , and $d(r_i, r_j)$ is the cost required to forward a look-up message from r_i to r_j (where $r_i \neq r_j$).

Step 2. Modified model construction and comparison. The second model generalizes the situation where the look-up begins with LAI—node inconsistency. Let us add to the basic model a new assumption that the meta-data of γ is relocated l times at locations $T = \{\tau_1, \tau_2, \dots, \tau_l\}$ (w.l.o.g. $l \leq k$). The other assumptions are the same as in the basic model. Here, provided that f is still the same as in the basic model, it is obvious that relocations of γ at $\tau (\in T)$ do not change W and R , and thus, Eq. (1) yields the same results as in the basic model. Therefore, we can see that the look-up process and cost remain the same as those of the basic model. \square

In conventional DHT schemes, the more dynamic the membership, the more control messages are required. With high-speed mobility, in particular, such overheads may be heavier because of frequent calibration of logical indexing topologies. In contrast, MX-DHT is absolutely free of such burdens. By virtue of this salient advantage, MX-DHT outperforms existing mobile DHT schemes in terms of communication costs under dynamic membership and high-speed mobility environments.

5 ANALYTICAL PERFORMANCE OF MX-DHT

In this section, we analytically assess the communication costs of MX-DHT. Since the publish/look-up of MX-DHT is an operation naturally inherited from the end-to-end communication of LER, we extend the previous analytical study on LER for assessing the communication cost of MX-DHT. Based on the probabilistic analysis of LER proposed in [9], we approximate the number of message transmissions per unit time required for (a) publish/look-up and (b) management operations of MX-DHT.

5.1 Preliminaries

We define mathematical terms and present some preliminary results for our analytical evaluation. For the convenience of reference, we use the same notations as in [9] whenever possible.

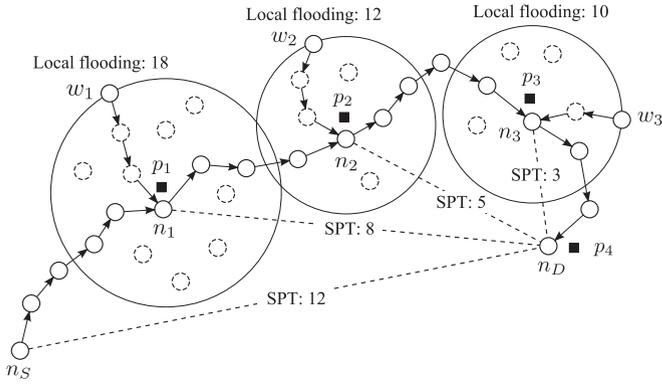


Fig. 9. Example used for the analytical performance evaluation.

① x_m, x_n . $x_m(x_n)$ is the set of nodes that have m -hops (n -hops) shortest path (SPT) distance from the destination in the publish/look-up process of MX-DHT (as in the routing process of LER). In Fig. 9, n_S, n_1, n_2 , and n_3 are the anchor nodes that perform local flood searching for each discovery-and-forward step. Their SPT distances from n_D are 12-hops, 8-hops, 5-hops, and 3-hops, respectively. We can see $n_S \in x_{12}, n_1 \in x_8, n_2 \in x_5$, and $n_3 \in x_3$.

② $p(x_m, x_n)$. The probability that an anchor node $n_\alpha (\in x_m)$ determines the next anchor node as $n_\beta (\in x_n)$ by local flood searching during an arbitrary publish/look-up process. In Fig. 9, for example, the consecutive anchor point pairs $(n_S, n_1), (n_1, n_2), (n_1, n_2)$, and (n_3, n_D) are the samples corresponding to $p(x_{12}, x_8), p(x_8, x_5), p(x_5, x_3)$, and $p(x_3, x_0)$, respectively.

③ $\bar{d}(x_m, x_n)$. The average hop count distance between two consecutive anchor nodes $n_\alpha (\in x_m, \in x_n)$ that the packet is forwarded during the publish/look-up. In Fig. 9, the forwarding distances between anchor nodes are 5, 4, 5, and 3. They are the samples for calculating $\bar{d}(x_{12}, x_8), \bar{d}(x_8, x_5), \bar{d}(x_5, x_3)$, and $\bar{d}(x_3, x_0)$.

④ $\bar{c}(x_m, x_n)$. The average local flood searching cost where an anchor node $n_\alpha (\in x_m)$ finds out the milestone toward the next anchor node as $n_\beta (\in x_n)$ during an arbitrary publish/look-up process. In Fig. 9, 0, 18, 12, and 10 are the samples for calculating $\bar{c}(x_{12}, x_8), \bar{c}(x_8, x_5), \bar{c}(x_5, x_3)$, and $\bar{c}(x_3, x_0)$. Note that first local flood searching cost is 0 in the example because the source node sends the message to the point where its local E-LET directs at the beginning of publish or look-up. On the other hand, second local flood searching cost 18 (= 15 + 3): 15 nodes participate the flood searching, and the milestone report from w_1 is delivered to n_1 via three-hop transmissions.

⑤ $p_l(x_m)$. The probability that the SPT distance from source to destination of an arbitrary publish/look-up is m . The example in Fig. 9 shows an event corresponding to $p_l(x_{12})$.

⑥ $\overline{PL}(x_m)$. The average path distance a message is forwarded during the publish/look-up, where the shortest path distance between the source and destination is m .

⑦ $\overline{SC}(x_m)$. The average cost of local flood searching during the publish/look-up, where the shortest path distance between the source and destination is m .

According to [9], $\overline{PL}(x_m)$ and $\overline{SC}(x_m)$ can be expressed as the following recursive equations:

$$\overline{PL}(x_m) = \frac{\sum_{n=1}^D \bar{d}(x_m, x_n) p(x_m, x_n) + \sum_{n=1}^{m-1} \overline{PL}(x_n) p(x_m, x_n)}{1 - p(x_m, x_m) - \sum_{n=m+1}^D \frac{n}{m} \cdot p(x_m, x_n)},$$

with $\overline{PL}(x_1) = 1, \overline{PL}(x_0) = 0$,

(2)

$$\overline{SC}(x_m) = \frac{\sum_{n=1}^D \bar{c}(x_m, x_n) p(x_m, x_n) + \sum_{n=1}^{m-1} \overline{SC}(x_n) p(x_m, x_n)}{1 - p(x_m, x_m) - \sum_{n=m+1}^D \frac{2n^3 - 3n^2 + n}{2m^3 - 3m^2 + m} \cdot p(x_m, x_n)},$$

with $\overline{SC}(x_1) = 1, \overline{SC}(x_0) = 0$,

(3)

where D is the network diameter. $p(x_m, x_n), \bar{d}(x_m, x_n), \bar{c}(x_m, x_n)$, and $p_l(x_m)$ are discrete histogram functions that are computed by a pre-evaluation called age gradient tree (AGT) analysis when a target mobility trace is given. For more details about Eq. (2), (3), and AGT analysis, please refer to [9].

5.2 Publish/Look-Up Cost

We derive the mean publish/look-up cost from $p_l(x_m), \overline{PL}(x_m)$, and $\overline{SC}(x_m)$. The publish or look-up of MX-DHT consists of a request and response phase: (a) the request message is routed from the source node to the rendezvous node that is responsible for the targeted data resource; and (b) the response message is delivered from the rendezvous node to the source. If the SPT distance between the source and rendezvous node is m , the forwarding cost is proportional to $2 \cdot \overline{PL}(x_m)$. The local flood searching cost required to route the request message is proportional to $\overline{SC}(x_m)$. Same as AODV and LER, the response message can be forwarded in reverse along the path of the request message. This is because the publish or look-up process can be completed within a relatively short time (generally, order of ten milliseconds). Hence, local flooding is not required for the response phase. Therefore, by averaging $2 \cdot \overline{PL}(x_m) + \overline{SC}(x_m)$ with $p_l(x_m)$, the approximated mean publish/look-up cost of MX-DHT is computed as

$$C_{PL} = \lambda_d \cdot \sum_{m=1}^D [2 \cdot \overline{PL}(x_m) + \gamma_s \cdot \overline{SC}(x_m)] \cdot p_l(x_m), \quad (4)$$

where D is the network diameter, λ_d denotes the publish/look-up request rate of the network, and γ_s denotes the message size of the local flood searching message normalized by the size of the publish/look-up request message.

5.3 Management Cost

The mean management cost consists of two components. The first component is the hello messaging cost involved in advertising a local LAI set to one-hop neighbors. The second component is the cost required for join and leave operations. It is trivial that they are determined by the hello message interval μ_h , and join (leave) rate λ_j (λ_l). Hence, we express the management cost as

$$C_M = \frac{N}{\mu_h} \cdot \gamma_h + \lambda_j \cdot \gamma_j + \lambda_l \cdot \gamma_l, \quad (5)$$

where N is the number of nodes in the network and γ_h, γ_j , and γ_l are the message sizes of the hello, join, and leave

protocol, respectively, normalized by the size of the publish/look-up message. In particular, if the network size is stable in the steady state, $\lambda_j = \lambda_l$. Thereby, Eq. (5) becomes $C_M = \frac{N}{\mu_h} \cdot \gamma_h + \lambda_j \cdot (\gamma_j + \gamma_l)$.

5.4 Overall Communication Cost

The mean overall communication cost of MX-DHT is the sum of the publish/look-up and management costs:

$$C_T = C_{PL} + C_M. \quad (6)$$

5.5 Discussion of Asymptotic Costs

Besides the above probabilistic modeling, it is possible to perform an asymptotic cost analysis of the key operations (i.e., publish, look-up, and management) of different approaches to show the comparative benefit of MX-DHT. We can use an existing asymptotic cost model [12], where there are N nodes with the radio range $r = \Theta(\sqrt{\log N / N})$, and their average velocity is proportional to the radio range. [12] showed that the expected hop length is given as $\Theta(1/r) = \Theta(\sqrt{N / \log N})$, and [39] showed the flooding costs to be $\Theta(1/r^2) = \Theta(N / \log N)$.

Floodings. Proactive and reactive flooding require network wide broadcast ($= \Theta(N / \log N)$) for advertisement and look-up, respectively. In contrast, proactive and reactive flooding do not incur any cost for look-up and advertisement, respectively.

VRR. Each node must maintain paths to its logical successor and predecessor nodes, and thus, its management cost is $\Theta(\sqrt{N / \log N})$. For publish and look-up, because VRR incurs hop stretch of $\Theta(\log N)$, the cost increases to $\Theta(\sqrt{N \log N})$ [40].

VCP. Due to the topology-awareness, two logical neighbors are also physical neighbors to each other. Hence, the management cost is $\Theta(1)$. The cost of publish/look-up is bound to the average hop length ($= \Theta(\sqrt{N / \log N})$) [23].

MX-DHT. Management only involves single-hop broadcasts with a cost of $\Theta(1)$. Publish and look-up require the cost of the average hop length, i.e., $\Theta(\sqrt{N / \log N})$. This is because the cost of local flooding is insignificant compared to the average hop length as proved in [7].

MX-DHT requires management cost that is an order of magnitude lower compared to that of floodings and VRR. It exhibits asymptotic costs similar to that of VCP. However, as we will see in the next section, MX-DHT provides much better service reliability compared to VCP in dynamic mobility environments.

6 SIMULATION RESULTS

We conducted an extensive set of simulations to evaluate the effect of the data publish/look-up rate, membership dynamics, and mobility on the performance of MX-DHT.

6.1 Simulation Setup

We used the OMNeT++ simulator and inet-2.1.0 framework that support various packet networking simulations, such as the TCP/IP suite and IEEE 802 technologies [41]. We considered IEEE 802.11b in ad hoc mode at 11 Mb/s on the MAC-PHY. The receiver sensitivity and pass loss exponent were calibrated to set a communication range of 125 m.

We implemented MX-DHT and five existing mobile DHT schemes: proactive flooding, reactive flooding, GHT, VRR, and VCP. We chose floodings to show an excessive communication burden of network-wide broadcast. GHT, VRR, and VCP were selected as representatives of overlay on wireless, DHT-inspired routing, and topology awareness DHT-inspired routing, respectively (see Section 2.2). To provide a better understanding of the simulation results, we note some details of each protocol's implementation.

MX-DHT. It was implemented as described in Section 4. During the motion tracking process, we used 2-hop flooding for witness search. If the search fails, 4-hop, 8-hop, and 16-hop floodings are repeated again until a next milestone was reported. If 16-hop search fails, the packet is dropped.

Proactive flooding. Each node floods an LAI advertisement message every 4s. (We originally used 1s. However, because the layer-2 buffer grows rapidly in simulations, we used 4s instead.) Upon receiving an advertisement, each node records the received LAI–node correspondence in the local resolution table, and thus, a look-up request is directly routed by the underlying routing protocol. As in MX-DHT, when a node joins (leaves) the network, one of its physical neighbors is selected as an LAI donor (recipient).

Reactive flooding. For every look-up, the request packet is flooded throughout the network. It can be seen as an AODV-like version of mobile DHT. The membership change operations are the same as in proactive flooding.

GHT. We divided the network area into 4×4 geographic clusters. One-to-one correspondence is made between the 16 clusters and 16 LAIs of which the logical address space consisted. The logical address is locally resolved by a geographic hash function, and thus, a publish/look-up request message is directly routed by the underlying routing protocol. It does not need join (leave) operations. Instead, nodes in a same cluster have to synchronize the LAI information. We had each node conduct in-cluster flooding every 1s.

VRR. It establishes end-to-end connection for every c -hop neighbor pair over the virtual ring. We set $c = 2$ in our simulations. For example, assume nodes n_7, n_9, n_{10}, n_{14} , and n_{17} are ordered on the virtual ring. Node n_{10} keeps the end-to-end connections to n_7, n_9, n_{14} , and n_{17} . It has [10, 14] as its LAI. When a new node n_{11} joins the network, node 10 establishes the connection to the joining node and breaks the connection to n_{17} . LAI [11, 14] is transferred from n_{10} to n_{11} . Recall that virtual neighbors can be far away from each other in the underlying network.

VCP. The publish/look-up and LAI management policy is the same as in VRR. One of the most important differences is that VCP reflects physical node proximities for the virtual address assignment.

Commonly, in every protocol implementation each node broadcasts a one-hop hello message every 1 s (with random jitters to prevent synchronization). We set the TTL (Time-To-Live) of a message as 32 hops. Floodings and GHT adopt greedy perimeter stateless routing (GPSR) as their underlying routing protocol [37]. All measurements were conducted within layer-3 components to capture only the behaviors of mobile DHT implementations. Unless otherwise mentioned, we followed the default settings listed in Table 1.

TABLE 1
Default Parameters

Parameter Name	Value
Simulation time	30 minutes
Node number (N)	200
Network area size	700 m \times 700 m
Data publish/look-up rate (λ_d)	50 requests/minute
Member join/leave rate (λ_j)	50 members/minute
Hello message interval (μ_h)	1 seconds
Transmission power	20 mW
Receiver sensitivity	-85 dbm
Pass loss exponent	2.76
Mobility model	Random waypoint model
Node velocity	20 m/s

6.2 Validation of Analytical Model

In this section, we validate the analytical communication costs of MX-DHT using the simulation results. In other words, Eq. (4), (5), and (6) are examined in terms of estimation accuracy.

First, the publish/look-up cost is validated. For the default mobility setting (random waypoint with 20 m/s), we computed $\bar{d}(x_m, x_n)$, $\bar{v}(x_m, x_n)$, and $p_l(x_m)$ by using the AGT analysis described in [9]. We compared the analytical results of Eq. (4) with the simulation results by varying the publish/look-up rate (λ_d) from 10 to 200 requests/minute. Considering the packet formats in the simulations, we set $\gamma_s = 1.0345$. Fig. 10a shows the results. As expected, the mean publish/look-up cost increases with respect to λ_d . We see that Eq. (4) achieves excellent estimation accuracy. For example, when $\lambda_d = 50$ requests/minute, the difference between the analytical and simulation result is only 1.58 percent (normalized by the simulation result).

We validate the mean management cost with various join/leave rates: $\lambda_j = \lambda_l = 10, 20, 50, 100$, and 200 members/minute. As in Table 1, $N = 200$ and $\mu_h = 1$ s. We set $\gamma_h = 0.9138$, $\gamma_j = 1.8621$, and $\gamma_l = 1.3103$ based on the packet formats in the simulations. Fig. 10b shows the results. In both the analytical and simulation results, the management cost slowly increases as the join/leave rate increases. This is because the cost of hello message transmission dominates the cost of join/leave operations. In short, $\frac{N}{\mu_h} \cdot \gamma_h \gg \lambda_j \cdot \gamma_j + \lambda_l \cdot \gamma_l$ in Eq. (5). For instance, the cost of a join/leave operation is only 2.81 percent of the management cost when $\lambda_j = 50$ members/minute. As we discuss below, this is a salient feature of MX-DHT compared to other existing mobile DHTs that pay a significant cost for join/leave operations.

The estimation accuracy is also good. For every λ_j , the difference between the analytical and simulation results is less than 4.5 percent.

Finally, we validate the mean overall communication cost while the mean node velocity v is increased from 10 to 50 m/s in increments of 10 m/s. Fig. 10c illustrates the results. For every node velocity, the estimation difference is within ± 3 percent. This excellent estimation accuracy is natural considering that the estimations of publish/look-up and management costs are sufficiently accurate [see Eq. (6)]. We can see that the mean overall communication cost of MX-DHT is slightly increased with respect to v if λ_j and λ_d are fixed. In Eq. (4), the terms $\overline{PL}(x_m)$ and $\overline{SC}(x_m)$ are mainly influenced by the node mobility pattern; however, the effect of velocity is not significant. Furthermore, Eq. (5) is independent of node velocity. This is in agreement with the study of [9], which revealed that the routing cost of LER is not sensitive to node velocity.

6.3 Impact of Join/Leave Rate

Existing mobile DHT protocols pay a significant communication cost to maintain consistent LAI-node correspondence for membership changes. In addition, the higher the membership dynamic, the lower the link availability, and thus, the publish/look-up reliability can also decrease. In terms of communication cost and service reliability, we now evaluate the extent to which MX-DHT outperforms the existing mobile DHTs under membership dynamics. To investigate the effect of the join/leave rate on the total communication cost and publish/look-up success ratio, we varied the join/leave rate (λ_j): 10, 20, 50, 100, and 200 members/minute.

Fig. 11a shows the total communication costs of MX-DHT and the existing mobile DHT with varying λ_j . As expected, MX-DHT achieves much better communication efficiency than the other schemes. For example, at a join/leave rate of 50 members/minute, the communication cost of VCP, reactive flooding, VRR, GHT, and proactive flooding is 170, 188, 290, 371, and 456 percent, respectively, of that of MX-DHT. As we explain in Section 4.2, MX-DHT does not require any signaling overhead, except for the transfer of LAI information within a motion-mix zone, i.e., the one-hop communication range. Therefore, the cost of hello messaging dominates the cost of join/leave operations, and thus, the total communication cost increases very slowly with respect to λ_j . Like MX-DHT, reactive flooding also uses only one-hop signaling for each join/leave event, and thus, it has somewhat good communication efficiency. The most part of

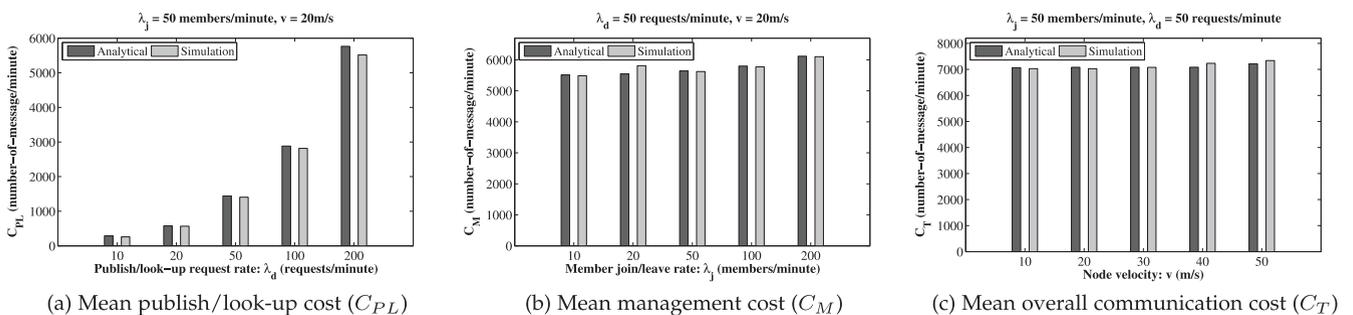


Fig. 10. Comparison of analytical and experimental performance.

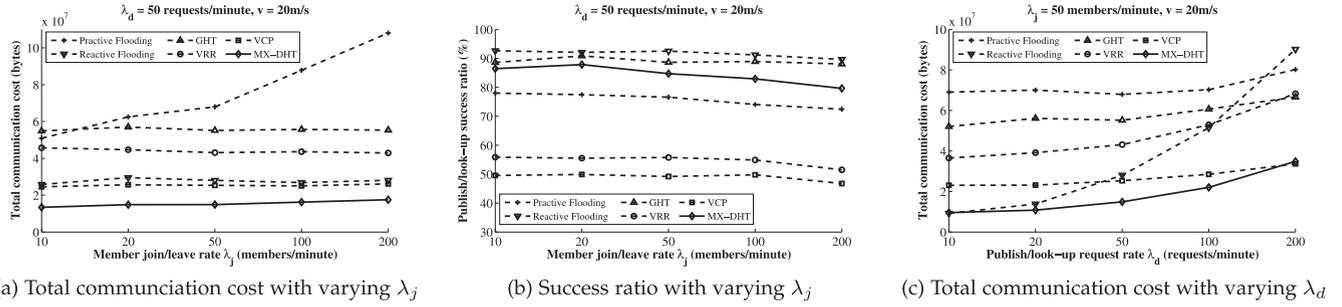


Fig. 11. Performance with respect to join/leave and publish/look-up rate.

the communication cost is spent on the network-wide broadcast for look-up operations. Recall that we used $\lambda_d = 50$ requests/minute as the default in our simulations. GHT and proactive flooding generate a significant amount of communications because of their network-wide and in-cluster advertisements. For proactive flooding, note that we used an advertisement interval of 4 s because of experimental feasibility. It is clear that a shorter advertisement interval results in an explosive increase in communication cost. For VRR and VCP, an interesting result is that the communication cost does not seem to be influenced by the increase in the join/leave rate. This is because we consider a default node mobility with $v = 20$ m/s. In fact, both membership dynamics and node mobility cause frequent changes in the virtual topology, and thus, the communication cost increases. However, if they appear together, their impact is partially cancelled out. In other words, if node mobility already exists in the network, the communication cost is significantly less sensitive to an increase in the join/leave rate. The opposite situation also holds true.

We plot the publish/look-up success ratios of MX-DHT and other mobile DHTs in Fig. 11b. For every protocol, as the join/leave rate increases, the success ratio gradually decreases. Reactive flooding and GHT achieve a success ratio between 88 and 92 percent by virtue of the network-wide and in-cluster flooding. Proactive flooding degraded the performance a little (72 percent \sim 78 percent), because we set an advertisement interval of 4 s for the sake of experimental feasibility. A shorter interval will result in a success ratio comparable to that of reactive flooding. VRR and VCP have a significantly more degraded success ratio (45 percent \sim 55 percent), because they experience frequent path disconnection of the virtual topology, although the physical topology is still connected. In the case of MX-DHT, as the join/leave rate increases, the accuracy of the encounter records decreases, and thus, the success ratio is degraded a little; however, it still retains a plausible reliability between 79 and 88 percent.

6.4 Impact of Publish/Look-Up Rate

To investigate the effect of the publish/look-up rate, we measured the total communication costs of MX-DHT and existing mobile DHT schemes by varying the publish/look-up rate: $\lambda_d = 10, 20, 50, 100,$ and 200 requests/minute. Fig. 11c illustrates the results. As we expected, the communication cost increases linearly with respect to the publish/look-up rate. An examination of the increasing rate shows that proactive flooding, VCP, GHT, MX-DHT, VRR, and reactive flooding require on average 1.753, 1.874, 2.202,

4.316, 5.130, and 12.967 KB/request, respectively, for each publish or look-up process. This suggests that reactive flooding, MX-DHT, and VRR require more look-up costs because they adopt an in-network path discovery strategy. Conversely, proactive flooding and GHT require only a forwarding cost, since path discovery is locally completed before the request packet departs. However, their underlying overhead for LAI advertisement or synchronization is significant, and thus, their total communication costs are still burdensome. Although VCP uses in-network path discovery similarly to VRR, it requires almost no overhead for path discovery, exploiting the benefit of topology-aware DHT routing. Nonetheless, it has to bear frequent path failure of the virtual topology due to the mobility dynamics. Thus, it experiences a success ratio of around 50 percent (see Fig. 11b). The results in Fig. 11 show that MX-DHT succeeds in reducing the communication cost significantly, while still retaining plausible reliability.

6.5 Behavior Under Synthetic Mobility

In this section, we discuss the influence of the node mobility pattern and velocity on the behavior of MX-DHT and other mobile DHT schemes. We consider two synthetic mobility: the random waypoint and Manhattan models. For each of the mobility models, we measured the total communication cost and publish/look-up success ratio of MX-DHT and the others, while increasing the node velocity v from 10 to 50 m/s in increments of 10 m/s. Each mobility was generated by MobiSim, an open synthetic mobility simulator [42].

Random waypoint. Fig. 12a shows the effect on the publish/look-up reliability when we increase the node velocity v from 10 to 50 m/s. In DHT-inspired routing approaches, such as VRR and VCP, the virtual topology is frequently disconnected as the node velocity increases. Therefore, the publish/look-up ratio significantly decreases. Reactive flooding, GHT, and MX-DHT succeed in retaining an acceptable level of reliability (74 percent \sim 94 percent). Proactive flooding has exceptionally lower results than expected. Recall that we used a LAI advertisement interval of 4 s. It would be better if the advertisement interval were shorter. If we consider that reactive flooding has a practical maximum of the success ratio (it uses network-wide broadcast for data look-up), MX-DHT achieves a bound result near to the maximum. For example, a success ratio of 94.09 percent versus the maximum at $v = 30$ m/s.

Fig. 12d shows the results for the total communication cost. The total communication cost is significantly less influenced by the increase in node velocity. This is because the

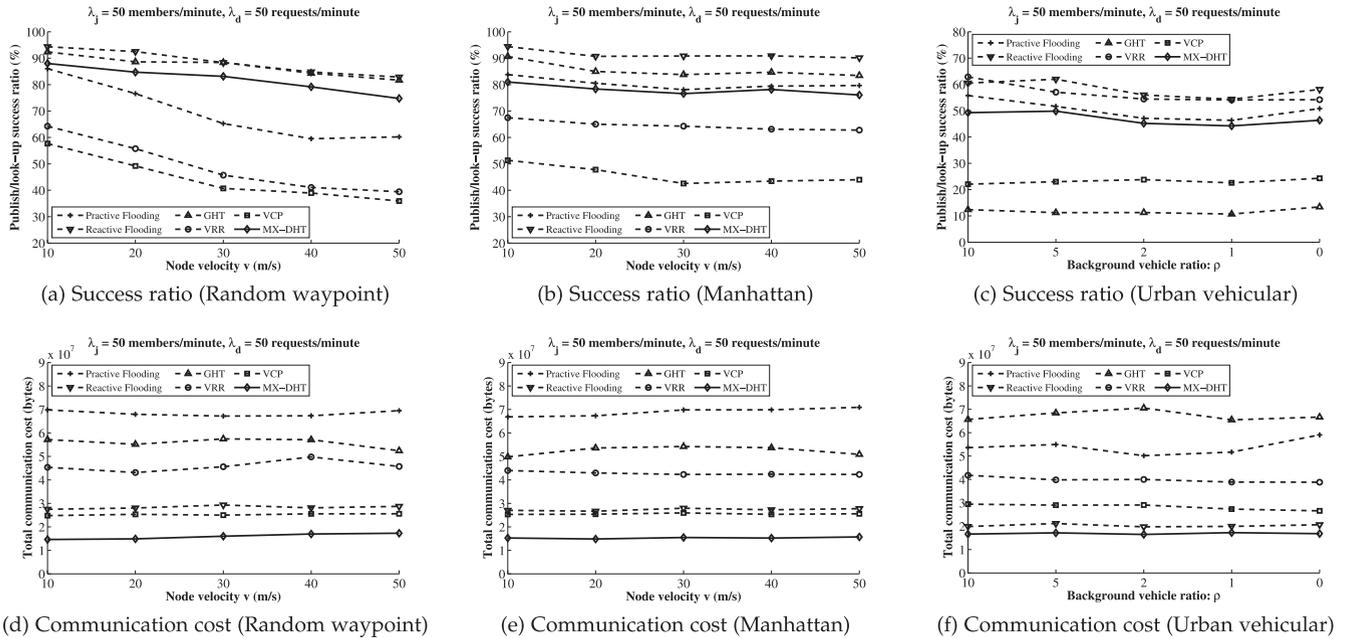


Fig. 12. Performance under random waypoint, Manhattan, and urban vehicular mobility.

membership dynamics also appears with $\lambda_j = 50$ members/minute as the default setting. Recall that the effects of the membership and mobility dynamics on the communication cost are counterbalanced, as shown in Section 6.3. The figure shows that MX-DHT has better communication efficiency than the other mobile DHTs. For example, at $v = 50$ m/s, the communication cost of VCP, reactive flooding, VRR, GHT, and proactive flooding is 156, 183, 285, 359, and 419 percent, respectively, of that of MX-DHT.

Manhattan. In the Manhattan scenario, we divided the network area into $100 \text{ m} \times 100 \text{ m}$ rectangular sub-areas. Each node moves only along the 2-lane roads among the sub-areas. Therefore, nodes are distributed on a much smaller space than in the random waypoint scenario. This increases the node density, and thus, the success ratio is improved as compared to that of the random waypoint mobility (see Figs. 12a and 12b). For instance, under random waypoint mobility, the success ratio of MX-DHT decreases by 5.57 percent from 40 to 50 m/s node velocity. In the Manhattan scenario, it decreases by 2.67 percent from 40 to 50 m/s. The results for the total communication cost show almost the same trend as in the random waypoint scenario (see Fig. 12e).

We find that MX-DHT succeeds in balancing the trade-off between the reliability and communication efficiency where the membership and mobility dynamic appear simultaneously. In particular, for both the random waypoint and Manhattan mobility, the communication cost is insensitive to the increase in the node velocity. This is in agreement with the results presented in [9].

6.6 Behavior Under Urban Vehicular Mobility

In the real world, user mobility is influenced by various factors, such as traffic regulations and the level of crowdedness in the street. Therefore, we examined the behavior of MX-DHT and other mobile DHT schemes in the urban vehicular mobility model. To generate vehicular mobility,

we used SUMO (Simulation of Urban Mobility), a road traffic simulation suite [43], [44]. We employed real world geographic data of the town of Galma in the vicinity of KAIST, Daejeon, South Korea from the OpenStreetMap.⁴ $1350 \text{ m} \times 1250 \text{ m}$ area, latitude: $36.3511^\circ \text{N} \sim 36.3581^\circ \text{N}$, longitude: $127.3741^\circ \text{E} \sim 127.3832^\circ \text{E}$ (see Fig. 13a).

In keeping with the traffic regulations in South Korea, we also imported traffic light signals and driving restrictions into SUMO by using JOSM and NETCONVERT utilities. We had 200 VoIs (*Vehicle-of-Interest* = node) take repeated random trips according to city section movement pattern proposed in [45]. The repeated trips were implemented via TraCI4MATLAB [46], an open library that provides a programmed control of SUMO. Unlike in the random waypoint and Manhattan model, we did not artificially control the node velocity. Instead, we injected background vehicles into SUMO in order to control the mobility dynamics in a realistic manner. The background vehicles did not involve DHT networking. We conducted the simulation with different numbers of background vehicles: 0, 200, 400, 1000, and 2000.

We plot the result of the publish/look-up success ratio and communication cost in Figs. 12c and 12f. The number of background vehicles is denoted by ρ , the proportion to the number of VoIs. For example, if $\rho = 2$, the number of background vehicles is 400. Note that we varied ρ in descending order, since the higher the number of background vehicles, the lower the mobility dynamics. In an urban vehicular mobility scenario, the most salient feature is that nodes frequently form a long tailed queue waiting for a green traffic light signal at the main crossroads (see the blue rectangles in Fig. 13b). The spatial distribution of the nodes is much less uniform, and thus, network partitions frequently appear. Due to such the unconnectivity, physical network reachability is highly limited. Fig. 12c shows that the success ratios of every mobile DHTs are significantly degraded because of frequent network

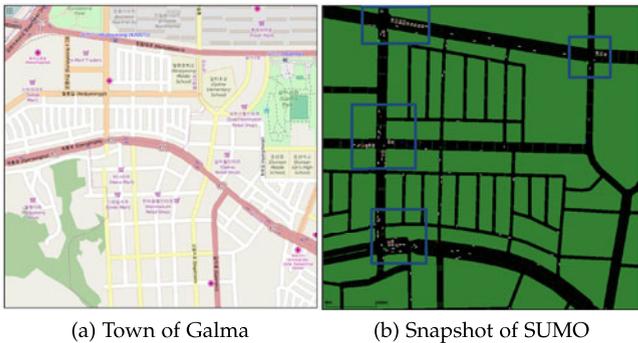


Fig. 13. Urban vehicular mobility in the town of Galma.

partitions. Therefore, the actual maxima of success ratio are at around 60 percent (reactive flooding in Fig. 12c). An encouraging result is that MX-DHT still nearly follows the actual maximum, for instance, 80.73 percent of the actual maximum at $\rho = 2$. GHT and VCP suffer a serious degradation of the reliability. It is obvious that geographic or topology awareness look-up approaches are vulnerable to network partitions. An interesting result is that the success ratio of VRR is less decreased as than that of other schemes. This is because VRR naturally provides a reactive LAI replication for network partition as explained in Section 3.3.4 of [21]. When the network is partitioned, each node attempts to expand its LAIs so that separate virtual rings are formed for each connected component. In fact, if this LAI replication is effective, the reliability of VRR should approach close to 100 percent. Yet, its reliability is not better than that of reactive flooding that does not provide replication. The reason is that VRR always suffers from frequent intermittent disconnections of the virtual ring if mobility and membership dynamics appear together. This vulnerability to mobility/membership dynamics reduces the efficacy of the LAI replication.

Fig. 12f shows the results for the total communication cost. MX-DHT has the best performance, as expected. For example, at $\rho = 2$, the communication cost of reactive flooding, VCP, VRR, proactive flooding, and GHT are 120, 176, 243, 305, and 428 percent, respectively, of that of MX-DHT. The cost of flooding somewhat decreases because broadcast packets are not disseminated well due to the partitions. Meanwhile, the communication cost of GHT is increased. This is why the number of in-cluster floodings increases because of the many signal-waiting nodes at main crossroads (see Fig. 13b). For VRR and VCP, the communication cost is less sensitive to network partitions, since each node manages a constant number of paths to its neighbor on the virtual topology.

6.7 Discussion

Using the simulations, we evaluated the effect of the publish/look-up rate, join/leave rate, and node mobility on the performance of MX-DHT and other mobile DHT schemes. Under realistic environments where the node mobility and membership dynamic appear simultaneously, MX-DHT achieves excellent communication efficiency while retaining good reliability. This is because every membership change operation is conducted within a motion-mix zone, i.e., a one-hop communication range, and hence, the management cost is significantly reduced. In particular, it is encouraging

that the *piece-meal-wise* publish/look-up is effective under frequent membership changes. However, if a node suddenly turns off or a link fails in the middle of a join/leave operation, some LAIs may be lost in the network. In addition, as described in Section 6.6, network partitions cause intermittent loss of LAIs. To improve its practicality, in our opinion MX-DHT should be used in combination with LAI loss recovery schemes.

Proactive loss recovery. A considerable amount of work has been reported on data replication schemes for providing a reliable mobile content disseminations service [47], [48], [49]. We can integrate the MX-DHT to prepare for the unexpected loss of LAIs.

Lazy loss recovery. In MX-DHT, if a node fails because of a turn off or network partition, the motion of the node is halted at the failure point. Assume there is a publish/look-up for an LAI of the lost node. When the request packet arrives at around the failure point, it cannot find a more recent encounter of the targeted LAI. In this case, we can choose a new carrier of the requested LAI from among the one-hop neighbor nodes. By simply re-assigning the lost LAI to a one-hop neighbor node, the *approximate motion continuity* of the recovered LAI is preserved.

For both the proactive and lazy approach, there is no doubt that the reliability improves up to nearly 100 percent. The adoption of proactive recovery will result in good recovery latency. Instead, we have to consider a substantial amount of communication overheads. On the other hand, if we use lazy recovery, the communication cost will be relatively low. However, the lost LAI will not be recovered until a publish or look-up is requested for the LAI. We leave a comparative study of the above loss recovery approaches as a future work.

7 CONCLUSION

This paper proposes Motion-Mix DHT, an LER-based mobile DHT protocol for wireless mobile environments where a high level of membership and mobility dynamics are considered. In the original LER, the network stores the motions of every node as distributed encounter milestones in a cooperative manner. Simply by extending the *last encounter table*, in MX-DHT, the network stores the motions of not only the nodes but also the LAIs for which the nodes are responsible. From the perspective of an individual node, the network provides a shared virtual geographic map, which provides the approximate motion trails of every node and the LAIs for which it is responsible. By exploiting *mobility assisted milestone sharing*, each node receives the location, routing, and data sharing services from the virtual map by using inexpensive local flooded queries. In addition, as described in Section 4.2 and 4.3, due to the *motion-mix* property, all join and leave operations require only one-hop message exchanges. By virtue of such simplicity, MX-DHT achieves excellent communication and energy efficiency as compared to existing mobile DHT protocols that require considerable communication cost to maintain logical ring or tree structures. To the best of our knowledge, MX-DHT is the first motion tracking-based mobile DHT paradigm that is especially targeting highly dynamic membership and high-speed mobility environments. The simulation results indicate that we successfully transferred the

bandwidth and energy efficiency of LER from the domain of ordinary end-to-end communications to that of content centric communications.

ACKNOWLEDGMENTS

This work was supported by Institute for Information & Communications Technology Promotion (IITP) Grants funded by the Korea government (MSIP) (No. B0101-15-1272, Development of Device Collaborative Giga-Level Smart Cloudlet Technology, and No. 10041244, SmartTV 2.0 Software Platform). Uichin Lee is the corresponding author of this paper.

REFERENCES

- [1] V. Jacobson, D. K. Smetters, J. D. Thornton, M. F. Plass, N. H. Briggs, and R. L. Braynard, "Networking named content," in *Proc. 5th ACM Int. Conf. Emerging Netw. Exp. Technol.*, Rome, Italy, Dec. 2009, pp. 1–12.
- [2] S. K. Fayazbakhsh, Y. Lin, A. Tootoonchian, A. Ghodsi, T. Koponen, B. Maggs, K. Ng, V. Sekar, and S. Shenker, "Less pain, most of the gain: Incrementally deployable ICN," in *Proc. ACM Int. Conf. Appl., Technol., Archit., Protocols Comput. Commun.*, Hong Kong, China, 2013, pp. 147–158.
- [3] G. Xylomenos, C. Ververidis, V. Siris, N. Fotiou, C. Tsilopoulos, X. Vasilakos, K. Katsaros, and G. Polyzos, "A survey of Information-centric networking research," *IEEE Commun. Surv. Tuts.*, vol. 16, no. 2, pp. 1024–1049, Apr.–Jun. 2014.
- [4] G. Fersi, W. Louati, and M. Ben Jemaa, "Distributed hash Table-based routing and data management in wireless sensor networks: A survey," *Wireless Netw.*, vol. 19, no. 2, pp. 219–236, 2013.
- [5] S. A. Abid, M. Othman, and N. Shah, "A survey on DHT-based routing for Large-scale mobile ad hoc networks," *ACM Comput. Surv.*, vol. 47, no. 2, pp. 20:1–20:46, Aug. 2014.
- [6] G. S. Tuncay, G. Benincasa, and A. Helmy, "Autonomous and distributed recruitment and data collection framework for opportunistic sensing," in *Proc. ACM 18th Annu. Int. Conf. Mobile Comput. Netw.*, New York, NY, USA, 2012, pp. 407–410.
- [7] M. Grossglauser and M. Vetterli, "Locating mobile nodes with EASE: Learning efficient routes from encounter histories alone," *IEEE/ACM Trans. Netw.*, vol. 14, no. 3, pp. 457–469, Jun. 2006.
- [8] H. Dubois-Ferriere, M. Grossglauser, and M. Vetterli, "Age matters: Efficient route discovery in mobile ad hoc networks using encounter ages," in *Proc. 4th ACM Int. Symp. Mobile Ad hoc Netw. Comput.*, Annapolis, MA, USA, Jun. 2003, pp. 257–266.
- [9] F. Bai and A. Helmy, "Impact of mobility on last encounter routing protocols," in *Proc. 4th Annu. IEEE Commun. Soc. Conf. Sensor, Mesh, Ad Hoc Commun. Netw.*, Jun. 2007, pp. 461–470.
- [10] M. Gerla, C. Wu, G. Pau, and X. Zhu, "Content distribution in VANETs," *Veh. Commun.*, vol. 1, no. 1, pp. 3–12, Jan. 2014.
- [11] F. Dressler, P. Handle, and C. Sommer, "Towards a vehicular Cloud—Using parked vehicles as a temporary network and storage infrastructure," in *Proc. ACM Int. Workshop Wireless Mobile Technol. Smart Cities*. New York, NY, USA, 2014, pp. 11–18.
- [12] M. Varvello, I. Rimac, U. Lee, L. Greenwald, and V. Hilt, "On the design of content-centric MANETs," in *Proc. 8th Int. Conf. Wireless On-Demand Netw. Syst. Services*, Jan. 2011, pp. 1–8.
- [13] B. Chen and R. Morris, "L+: Scalable landmark routing and address lookup for multi-hop wireless networks," *Lab. Comput. Sci.*, Massachusetts Inst. Technol., Cambridge, MA, USA, Tech. Rep. MIT-LCS-TR-837, 2002.
- [14] A. C. Viana, M. D. de Amorim, S. Fdida, and J. F. de Rezende, "An underlay strategy for indirect routing," *Wireless Netw.*, vol. 10, no. 6, pp. 747–758, Nov. 2004.
- [15] J. Eriksson, M. Faloutsos, and S. Krishnamurthy, "DART: Dynamic address routing for scalable ad hoc and mesh networks," *IEEE/ACM Trans. Netw.*, vol. 15, no. 1, pp. 119–132, Feb. 2007.
- [16] M. Caleffi, G. Ferraiuolo, and L. Paura, "Augmented tree-based routing protocol for scalable ad hoc networks," in *Proc. IEEE Int. Conf. Mobile Adhoc Sensor Syst.*, Oct. 2007, pp. 1–6.
- [17] S. Jha, B. Jouaber, and K. Ahmed, "Dynamic-address-allocation based scalable routing protocol in context of node mobility," in *Proc. IEEE Wireless Hive Networks Conf.*, Aug. 2008, pp. 1–6.
- [18] S. Ratnasamy, B. Karp, L. Yin, F. Yu, D. Estrin, R. Govindan, and S. Shenker, "GHT: A geographic hash table for data-centric storage," in *Proc. Int. Workshop Wireless Sens. Netw. Appl.*, Atlanta, GA, USA, 2002, pp. 78–87.
- [19] Y.-T. Yu, T. Punihaoale, M. Gerla, and M. Sanadidi, "Content routing in the vehicle cloud," in *Proc. IEEE Mil. Commun. Conf.*, Oct./Nov. 2012, pp. 1–6.
- [20] A. Attwood, D. Lamb, and O. Abuelmaatti, "Position-relative identities in the internet of things: An evolutionary GHT approach," *IEEE Internet Things J.*, vol. 1, no. 5, pp. 497–507, Oct. 2014.
- [21] M. Caesar, M. Castro, E. B. Nightingale, G. O'Shea, and A. Rowstron, "Virtual ring routing: Network routing inspired by DHTs," in *Proc. ACM Int. Conf. Appl., Technol., Archit., Protocols Comput. Commun.*, Pisa, Italy, 2006, pp. 351–362.
- [22] T. Fuhrmann, "scalable routing in sensor actuator networks with churn," in *Proc. IEEE 3rd Annu. Conf. Sens. Ad Hoc Commun. Netw.*, Sep. 2006, pp. 30–39.
- [23] A. Awad, R. German, and F. Dressler, "exploiting virtual coordinates for improved routing performance in sensor networks," *IEEE Trans. Mobile Comput.*, vol. 10, no. 9, pp. 1214–1226, Sep. 2011.
- [24] S. Jain, Y. Chen, and Z.-L. Zhang, "VIRO: A scalable, robust and namespace independent virtual id routing for future networks," in *Proc. IEEE INFOCOM*, Apr. 2011, pp. 2381–2389.
- [25] H. Wirtz, T. Heer, R. Hummen, and K. Wehrle, "Mesh-DHT: A locality-based distributed look-up structure for wireless mesh networks," in *Proc. IEEE Int. Conf. Commun.*, Jun. 2012, pp. 653–658.
- [26] S. Abid, M. Othman, N. Shah, M. Ali, and A. Khan, "3D-RP: A DHT-based routing protocol for MANETs," *The Comput. J.*, vol. 58, no. 2, pp. 258–279, Feb. 2015.
- [27] C. Perkins, E. Belding-Royer, and J. Das, "Ad hoc on-demand distance vector (AODV) routing," *Network Working Group IETF RFC 3561*, Jul. 2003.
- [28] M. Grossglauser and D. Tse, "Mobility increases the capacity of ad hoc wireless networks," *IEEE/ACM Trans. Netw.*, vol. 10, no. 4, pp. 477–486, Aug. 2002.
- [29] N. Lu and X. Shen, "Scaling laws for throughput capacity and delay in wireless networks: A survey," *IEEE Commun. Surv. Tuts.*, vol. 16, no. 2, pp. 642–657, Apr.–Jun. 2014.
- [30] S. Ioannidis and P. Marbach, "A Brownian motion model for last encounter routing," in *Proc. 25th IEEE Int. Conf. Comput. Commun.*, Apr. 2006, pp. 1–12.
- [31] X. Yang, X. Fan, W. Yu, X. Fu, and S. Yang, "HLLS: A history information based light location service for MANETs," *Comput. Netw.*, vol. 56, no. 2, pp. 731–744, 2012.
- [32] F. Ducatelle, G. A. Di Caro, A. Forster, M. Bonani, M. Dorigo, S. Magnenat, F. Mondada, R. O'Grady, C. Pinciroli, P. Retornaz, V. Trianni, and L. Gambardella, "Cooperative navigation in robotic swarms," *Swarm Intell.*, vol. 8, no. 1, pp. 1–33, 2014.
- [33] M. Pascoe-Chalke, J. Gomez-Castellanos, V. Bonilla-Gonzalez, M. Lopez-Guerrero, V. Rangel-Licea, and E. Rodriguez-Colina, "Direction of encounter (DoE): A Mobility-based location method for wireless Networks," *IEEE Trans. Mobile Comput.*, vol. 13, no. 11, pp. 2524–2536, Nov. 2014.
- [34] T. Spyropoulos and A. Picu, *Opportunistic Routing*. Hoboken, NJ, USA: Wiley, Mar. 2013, ch. 11, pp. 419–452.
- [35] T. V. Spyropoulos, R. N. B. Rais, T. Turletti, K. Obraczka, and A. Vasilakos, *Delay Tolerant Networking: Routing Taxonomy and Design*. New York, NY, USA: Taylor & Francis, Dec. 2013, ch. 39, pp. 431–449.
- [36] Y. Lu, M. Gerla, T. Le, V. Rabsatt, and H. Kalantarian, "Community aware content retrieval in disruption-tolerant networks," in *Proc. 13th Annu. Mediterranean Ad Hoc Netw. Workshop.*, Jun. 2014, pp. 172–179.
- [37] B. Karp and H. T. Kung, "GPSR: Greedy perimeter stateless routing for wireless networks," in *Proc. 6th Annu. Int. Conf. Mobile Comput. Netw.*, Boston, MA, USA, 2000, pp. 243–254.
- [38] J. Kong, D. Wu, X. Hong, and M. Gerla, "Mobile traffic sensor network versus motion-MIX: Tracing and protecting mobile wireless nodes," in *Proc. 3rd ACM Workshop Security Ad Hoc Sensor Netw.*, Alexandria, VA, USA, 2005, pp. 97–106.
- [39] S. Shakkottai, X. Liu, and R. Srikant, "The multicast capacity of large multihop wireless networks," *IEEE/ACM Trans. Netw.*, vol. 18, no. 6, pp. 1691–1700, Dec. 2010.
- [40] D. Malkhi, S. Sen, K. Talwar, R. Werneck, and U. Wieder, "Virtual ring routing trends," in *Proc. Int. Symp. Distrib. Comput.*, 2009, pp. 392–406.

- [41] A. Varga and R. Hornig, "An overview of the OMNeT++ simulation environment," presented at the 1st ACM/ICST Int. Conf. Simulation Tools Techniques Communication, Networks and Systems, Marseille, France, Mar. 2008.
- [42] S. Mousavi, H. Rabiee, M. Moshref, and A. Dabirmoghaddam, "MobiSim: A framework for simulation of mobility models in mobile ad-hoc networks," in *Proc. 3rd IEEE Int. Conf. Wireless Mobile Comput., Netw., Commun.*, Oct. 2007, pp. 82–82.
- [43] D. Krajewicz, J. Erdmann, M. Behrisch, and L. Bieker, "Recent development and applications of SUMO—Simulation of urban mobility," *Int. J. Adv. Syst. Meas.*, vol. 5, nos. 3/4, pp. 128–138, 2012.
- [44] C. Sommer, R. German, and F. Dressler, "Bidirectionally coupled network and road traffic simulation for improved IVC analysis," *IEEE Trans. Mobile Comput.*, vol. 10, no. 1, pp. 3–15, Jan. 2011.
- [45] V. A. Davies, "Evaluating mobility models within an ad hoc network," Masters thesis, Colorado School of Mines, Golden, CO, USA 2000.
- [46] A. F. Acosta and J. Espinosa, "TraCI4MATLAB: Re-engineering the python implementation of the TraCI interface," in *Proc. 2nd SUMO User Conf. Model. Mobility Open Data*, May 2014, pp. 155–170.
- [47] A. Derhab and N. Badache, "Data replication protocols for mobile ad-hoc networks: A survey and taxonomy," *IEEE Commun. Surv. Tuts.*, vol. 11, no. 2, pp. 33–51, Apr.–Jun. 2009.
- [48] J. Garcia-Reinoso, N. Fernández, I. Vidal, and J. A. Fisteus, "Scalable data replication in content-centric networking based on alias names," *J. Netw. Comput. Appl.*, vol. 47, pp. 85–98, Jan. 2015.
- [49] E. Spaho, A. Barolli, F. Xhafa, and L. Barolli, "P2P data replication: Techniques and applications," in *Modeling and Processing for Next-Generation Big-Data Technologies* (ser. Modeling and Optimization in Science and Technologies), vol. 4, F. Xhafa, L. Barolli, A. Barolli, and P. Papajorgji, Eds. New York, NY, USA: Springer, 2015, pp. 145–166.



Seungjae Shin received the BS degree in electrical and computer engineering from Chung-Nam National University, Daejeon, Korea, in 2007, and the MS degree in computer science from the Korea Advanced Institute of Science and Technology (KAIST), Daejeon, in 2009. He is currently working toward the PhD degree at KAIST. His research interests include mobile communication and networking.



Uichin Lee received the BS degree in computer engineering from Chonbuk National University, Jeonju, Korea, in 2001, the MS degree in computer science from the Korea Advanced Institute of Science and Technology (KAIST), Daejeon, Korea, in 2003, and the PhD degree in computer science from the University of California (UCLA), Los Angeles, CA, USA, in 2008. He continued his studies at UCLA as a postdoctoral research scientist, from 2008 to 2009, and then was with Alcatel-Lucent Bell Labs as a member of technical staff till

2010. He is currently an associate professor in the Department of Knowledge Service Engineering, KAIST. His research interests include social computing systems and mobile/pervasive computing. He is a member of the IEEE.



Falko Dressler received the MSc and PhD degrees from the Department of Computer Science, University of Erlangen, Erlangen, Germany, in 1998 and 2003, respectively. He is currently a full professor of computer science and the head of the Distributed Embedded Systems Group, Department of Computer Science, University of Paderborn, Paderborn, Germany. He is an editor for journals such as *IEEE Transactions on Mobile Computing*, *Elsevier Ad Hoc Networks*, *Elsevier Computer Communications*, and *Elsevier Nano*

Communication Networks. He was the guest editor of special issues on self-organization, autonomic networking, vehicular networks, and bio-inspired communication for *IEEE Journal on Selected Areas in Communications*, *Elsevier Ad Hoc Networks*, and others. He was the general chair of IEEE/ACM BIONETICS in 2007, IEEE/IFIP WONS in 2011, IEEE VNC in 2014, and ACM MobiHoc 2016, TPC cochair for IEEE INFOCOM, IEEE VNC, IEEE VTC, IEEE GLOBECOM, and ACM MSWiM, and Poster/Demo Chair for ACM MobiCom. He regularly serves in the program committee of leading IEEE and ACM conferences. He authored the textbooks *Self-Organization in Sensor and Actor Networks* published by Wiley in 2007 and *Vehicular Networking* published by Cambridge University Press in 2014. He has been an IEEE distinguished lecturer as well as an ACM distinguished speaker in the fields of inter-vehicular communication, self-organization, and bio-inspired and nano-networking. His research objectives include adaptive wireless networking, self-organization techniques, and embedded system design with applications in ad hoc and sensor networks, vehicular networks, industrial wireless networks, and nano-networking. He is a senior member of the IEEE as well as a senior member of ACM.



Hyunsoo Yoon received the bachelor's degree from the Seoul National University, Seoul, Korea, in 1979, the MS degree in computer science from the Korea Advanced Institute of Science and Technology (KAIST), Daejeon, Korea, in 1981 and the PhD degree from the Ohio State University, Columbus, OH, USA, in 1998. He was a member of researcher at Tong-Yang Broadcasting Company during 1978–1981 and Samsung Electronics during 1981–1984. During 1988–1989, he was a member of technical staff at AT&T Bell

Labs. Since 1989, he has been a professor at the Department of Computer Science, KAIST. His research interests include computer network and security. He is a member of the IEEE.

▷ For more information on this or any other computing topic, please visit our Digital Library at www.computer.org/publications/dlib.