Mobility-aware Data Management on Mobile Wireless Networks

Akiko Yamasaki*, Hirozumi Yamaguchi*, Shinji Kusumoto*, and Teruo Higashino*
* Graduate School of Information Science and Technology, Osaka University

***Abstract*— We design a Mobility-aware DAta management (MoDA) scheme for mobile ad hoc networks (MANETs) composed by mobile nodes such as urban pedestrians and vehicles. By fully utilizing the knowledge about the trajectories of mobile nodes, MoDA determines how replicas of data are copied and transferred among mobile nodes to provide the required data accessibility. Experimental results have shown that MoDA could achieve the small number of data transfers among mobile nodes while keeping reasonable accessibility.**

I. INTRODUCTION

In future, mobile nodes will have enough processing power and storage space. Then they will collaboratively be able to maintain copies of location-dependent data, and to provide the data to nearby nodes through MANET or to the nodes in remote regions via gateways (see Fig. 1). Such architecture can save wide-area wireless resources by localizing traffic and can facilitate to discover location-dependent information. However, to manage data on MANETs is not so easy because of node mobility. Since mobile nodes are usually cars and pedestrians, the network topology and its constituents change from time to time. This means that data holders may move within the region, or to make matters worse, may disappear. In such an environment, keeping the copies of data on appropriate nodes, avoiding frequent data transfers between nodes is a challenging task.

To cope with this problem, we propose a Mobility-aware DAta management (MoDA) scheme for MANETs. MoDA places replicas (copies) of data on mobile nodes so that any node can reach one of those replica holders within a designated hop distance. To maintain replicas in such a way on mobile nodes, MoDA determines a distribution strategy that indicates how these replicas are copied and transferred among mobile nodes, utilizing the knowledge about their moving "flows". To minimize the number of data transfers between mobile nodes, we formulate the problem of deriving the optimal distribution strategy as a set cover problem which is known as NP-hard, and apply a greedy algorithm to obtain feasible solutions. For collecting the knowledge about moving flows of mobile nodes, we may exploit our approach [1] that can estimate moving flows of pedestrians in city sections from a small amount of density observation information, or the other traffic estimation methods which have been extensively studied so far.

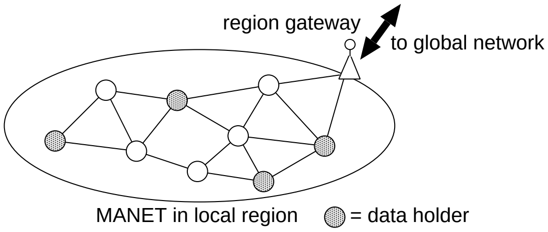


Fig. 1. Data Management Service Supported by MANET

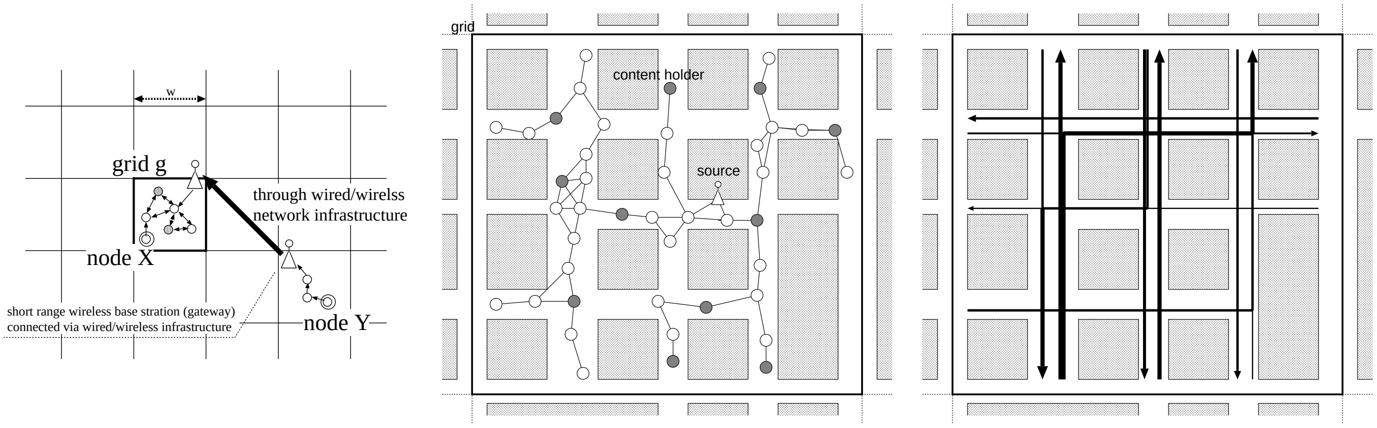
Many techniques have been designed for data management on MANETs that do not utilize the knowledge about mobility [2], [3]. Also, several approaches have recently proposed [4], [5] to schedule the delivery of messages, utilizing the knowledge of movement of nodes. But actually these methods determine the trajectories of messages and do not take the *trajectories of mobile nodes* into consideration. There is a technique where the trajectories of vehicles are utilized for efficient message delivery on VANETs [6], but its goal is to minimize the delivery time of the messages. Different from all the approaches above, we deal with the problem to maintain the accessibility to information on MANET with less communication overhead, utilizing the knowledge about the trajectories of mobile nodes.

Experimental results based on our network simulator MobiREAL [1], [7], [8] which can create realistic behavior of pedestrians in cities have shown that the distribution strategies of MoDA could achieve the small number of data transfers while keeping reasonable hop distances from any node to a replica holder, compared with a random-based distribution strategy with variations of parameter values.

II. MOBILITY-AWARE DATA MANAGEMENT

A. System Model and Overview

We assume that each mobile node is equipped with GPS (or an alternative position-measurement device) and a short range wireless device with communication range r. Each node also has storage which stores data (called *contents* hereafter) and is capable of serving the contents to the other nodes. We show the network architecture in Fig. 2(a). We divide geography into square grids, and MoDA tries to maintain the copies of contents associated with a grid g, on the mobile nodes which reside in g. Like node X in Fig. 2(a), a request for accessing a content associated with the same grid g is broadcast on the MANET composed in g, and the requested node will receive a reply from the nearby copy holder (if any). We note that some nodes in other grids like node Y may wish to obtain



(a) Network Architecture

(b) MANET in Grid (expected hop distance $H = 2$)

(c) Movement Flows in Grid

Fig. 2. Example

the contents in grid g. We assume that such a request is forwarded to g by some query forwarding techniques such as geocast [9] through ad hoc networks, mesh networks, or wired infrastructures to the gateway of the MANET in the grid. Since a numerous number of techniques regarding geographic addressing and query forwarding have been investigated so far, it is out of scope of this paper.

Hereafter we denote a content held in grid g by C^g. The proposed method MoDA maintains a content C^g sent from a stationary node in g (called *source node*, which may be the gateway of g) on the MANET so that any mobile node on the MANET can reach C^g within a specified hop distance called *expected hop distance* and denoted by H. Fig. 2(b) shows an ideal case where copies are distributed on nodes so that every node can reach a content holder node within 2 hops. The goal of MoDA is to generate and maintain such placement under mobility of nodes, if $H = 2$ is designated.

MoDA fully utilizes the knowledge about the movement of mobile nodes. In particular, we assume that mobile nodes are cars or pedestrians in city sections and they form flows like Fig. 2(c). Using the information about the moving flows (hereafter simply flows), in MoDA, the source node derives a *distribution strategy*, which specifies at which point each content holder on a flow should copy the content and transfer to the other nodes on other flows, in order to distribute the content to all the streets. To minimize the number of transfers between nodes, an optimal strategy is calculated by the algorithm described in Section II-C.

Since copies of the content held by content holders will be unavailable as these holders move outside the grid g, the source node transmits the content for every $\frac{2 \cdot H \cdot \alpha \cdot r}{V_{ave}}$ time units where α $(0 < \alpha \leq 1)$ denotes the ratio of average one-hop distance to communication r, and V_{ave} denotes the average speed of mobile nodes, to maintain the accessibility to the content. This means that each node between two copies of the content transmitted subsequently is expected to reach the closest copy within H hops. The strategy is attached to every copy of the content so that each content holder can know and follow the strategy.

B. Problem Statement

Here we focus on a content C^g in grid g and formulate the problem to derive an optimal distribution strategy. We assume that the following constants are given; the "time to live" of a copy of content C^g (denoted by $ttl(C^g)$), which is the time duration from the moment when a mobile node receives a copy of C^g till the moment when it is expired (we assume that this value is common for all the nodes and copies), the expected hop distance H, communication range r, the ratio α of the average one-hop distance to the communication range, and the average speed V_{ave} of mobile nodes.

The problem takes two inputs. One is a graph $G = (V, E, pos, v_0)$ that represents the road structure of grid g where V, $E \subseteq V \times V$, pos and $v_0 \in V$ denote a set of vertexes, a set of edges, a coordinate assignment function that assigns to each vertex a unique coordinate and the point of a source node, respectively. The graph is called a *road graph*. To generate a road graph from a street map with the point of a source node we apply the following rules. We let V be the set of all the intersections, the cross points of streets and grid edges, dead ends of streets, and the point of the source node. Also we let E be the set of all the street segments between two vertexes, let v_0 be the point of the source node, and pos be an assignment function that assigns geographic coordinates of these points to the corresponding vertexes. Then we add vertex(es) for each edge longer than $V_{ave} \cdot ttl(C^g)$ to cut it into edges not longer than $V_{ave} \cdot ttl(C^g)$. This is to add all the potential points where data copies and transfers between nodes are performed.

The problem takes another input, a set F of mobile nodes' flows (or simply *flows*) on the road graph G. Each flow is a pair $f_i = (fp_i, fr_i)$ where fp_i is an acyclic undirected path of G

and fr_i is the flow rate of f_i. fp_i is denoted as a sequence of points in V. For simplicity of discussion we assume that each flow $f_i = (fp_i, fr_i)$ consists of two directed flows toward opposite directions whose flow rates are the half of fr_i. This assumption simplifies the algorithm and in fact it is true in the real world. We also assume that each edge in E is contained by at least one flow of F. If this assumption is not satisfied, we just simply remove such an edge from E because no node passes the edge.

Here, for a flow $f \in F$, we let $\overline{f}$ denote a sub-flow of f (*i.e.* a part of f). This sub-flow is called *flow fragment* of f. For example, if f is v_i-v_j-v_k, then v_i-v_j, v_j-v_k and v_i-v_j-v_k are the flow fragments of f. Without confusion, we let $\overline{F}$ denotes a set of flow fragments of flows in F, and $\overline{\mathcal{F}}$ denotes the set of *all* the flow fragments of flows in F.

The output of the problem is a *distribution strategy* $ST = (\overline{F}, IC)$. $\overline{F} \subseteq \overline{\mathcal{F}}$ is a set of flow fragments of flows in F and $IC \subseteq \overline{F} \times V \times \overline{F}$ is a set of tuples of two flow fragments and a vertex which is shared by the two flow fragments. Each tuple $(\overline{f_i}, v, \overline{f_j}) \in IC$ means that a content holder in flow f_i copies and transfers the content to a node in f_j when it arrives at vertex v. This tuple is called an *inter-node copy* hereafter. IC must contain at least one tuple $(-, v_0, \overline{f_k})$ where v_0 is the point of the source node and "$-$" represents that the source node transmits the content.

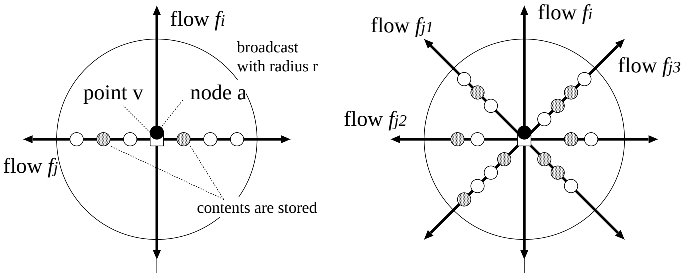
According to a distribution strategy $ST = (\overline{F}, IC)$, for each tuple $(-, v_0, \overline{f_k}) \in IC$, we let the source node at vertex v_0 perform an inter-node copy to a node on f_k, and for each tuple $(\overline{f_i}, v, \overline{f_j}) \in IC$ we let a content holder perform an inter-node copy to a node on flow fragment f_j when the content holder is on flow fragment $\overline{f_i}$ and reaches vertex v. Then C^g transmitted by the source node is copied and delivered through all edges, and finally discarded. If the source node at v_0 transmits C^g for every $\frac{2 \cdot H \cdot \alpha \cdot r}{V_{ave}}$ time units, then the copies of C^g will be placed on G with inter-distance $2 \cdot H \cdot \alpha \cdot r$. Thus any node between the two copies will find a copy within H hops in the ideal case.

We say that $ST = (\overline{F}, IC)$ is *feasible* iff ST satisfies the following two conditions; (i) each edge in E is contained only one flow fragment in $\overline{F}$, and (ii) the flow fragments of $\overline{F}$ connected by inter-node copies of IC do not form a loop. These conditions indicate that in a feasible distribution strategy the flow fragments connected by inter-node copies form a tree on G, rooted at v_0. Here we say that a feasible distribution strategy $ST = (\overline{F}, IC)$ is *optimal* iff $|IC|$ is minimum. The optimal strategy distributes the copies with the least number of inter-node copies.

In the following section we present a heuristic algorithm for the optimal distribution strategy problem.

C. Algorithm

The key idea of our heuristic algorithm is to formulate the problem as a set cover problem, which is known as NP-hard [10]. The set cover problem takes a set of subsets $S = \{S_1, S_2, ..., S_m\}$ of the universal set $U = \{1, ..., n\}$ as an input, and determines the smallest subset $T = \{T_1, ...T_k\}$ of S such that $\bigcup_{i=1}^{k} T_i = U$.



(a) (b)

Fig. 3. Implementation of Inter-node Copy by Broadcast; (a) Single Inter-node Copy and (b) Multiple Inter-node Copies

In our case, the set E of edges corresponds to the universal set U, and the set $\overline{\mathcal{F}}$ of flow fragments corresponds to the set S of subsets of U. We apply a known greedy algorithm of at most m steps where starting with $T = \emptyset$, at each step we add a subset S_i that contains the most items of U that have not yet included in the subsets in T. The algorithm stops when all the items of U are selected. Applying this greedy algorithm, we obtain the set $\overline{F}$ of flow fragments where $|\overline{F}|$ is small enough. Since the greedy algorithm takes the set $\overline{\mathcal{F}}$ of all the possible flow fragments as an input, it can always find a set $\overline{F}$ where each edge is covered by only one flow fragment. This means that for the derived $|\overline{F}|$, we can always find a set IC of inter-node copies by choosing them one by one avoiding a loop until $|IC| = |\overline{F}|$ holds. We note that we need to start the greedy algorithm with a flow fragment $\overline{f}$ where one of its ends is v_0 as the initial subset, $(-, v_0, \overline{f})$ is chosen in IC. As a result, the algorithm can determine a feasible distribution strategy $ST = (\overline{F}, IC)$ where $|IC|$ is as small as possible.

D. Implementation of Inter-node Copy Mechanism

An inter-node copy $(\overline{f_i}, v, \overline{f_j})$ requires each content holder (say a) on flow $\overline{f_i}$ to send the content to another node (say b) on flow $\overline{f_j}$ when node a arrives at point v. Since we assume that each node knows its position, the node can know when it needs to perform the inter-node copy. However it is really unrealistic to assume that each node knows flow information. Therefore, we assume that nodes cannot identify the flows to which their neighboring nodes belong.

Under this assumption, we take a simple 1-hop broadcast approach to implement inter-node copy $(\overline{f_i}, v, \overline{f_j})$. For each inter-node copy $c_l = (\overline{f_i}, v, \overline{f_j})$, the source node, which knows the whole flow information, specifies a value m_l given as follows

$$m_l = fr_j \cdot \frac{d_j}{V_{ave}}$$

where fr_j denotes the flow rate of f_j and d_j denotes the length of the part of flow f_j which is inside the circle of radius r centered at v (for example, in case of Fig. 3(a), $d_j = 2r$). m_l represents the estimated number of nodes which are on flow f_j and which are within the transmission range of node u at point v. Thus we let each node which hears the broadcast of C^g with parameter m_l store the content with probability $\frac{2}{m_l}$ (this probability is called *reception probability*) and follow the distribution strategy if it stores the content. As a result, at least two nodes on flow f_j store the content. Since each flow includes nodes toward opposite directions, we select two nodes per flow. In the example of Fig. 3(a), $m_l = 6$ and as a result two nodes store the content.

If the distribution strategy contains multiple inter-node copies which should be performed by node a on flow f_i at point v like $c_1 = (\overline{f_i}, v, \overline{f_{j_1}})$, $c_2 = (\overline{f_i}, v, \overline{f_{j_2}})$, ... and $c_n = (\overline{f_i}, v, \overline{f_{j_n}})$, node a only performs one broadcast with the smallest m_l for all c_l ($l = 1...n$). By doing this, on each flow at least two nodes store the content and follow the distribution policy. In the case of Fig. 3(b) ($l = 1, 2, 3$), $c_2 = (\overline{f_i}, v, \overline{f_{j_2}})$ has the smallest estimated number "4" of nodes, thus the reception probability set by node a at point v is 0.5.

We note that by this broadcast-based method, some nodes on the flows which are not specified in inter-node copies may store the content. Such a node carries the content, but the content on the node will be expired and discarded after $ttl(C^g)$ unit time passes.

III. EXPERIMENTAL RESULTS

a) Simulation Settings: We have evaluated the performance of MoDA through our network simulator *MobiREAL* [1], [7], [8]. We have used *Manhattan model* [11] of 500m×500m where we generated 13 shortest paths between randomly selected points at the edge of the region, and used them as flows of mobile nodes. Their flow rates were set randomly such that the number of nodes in the field at a time was about 350. We generated new nodes at an end point of each flow, and if the generated nodes reached another end point, they were removed. Due to limitation of space, we omitted the experimental results using *Osaka downtown model*. Interest readers may refer to Ref. [12]. We have focused on single data C^g. A data requests was generated for every five seconds by a randomly selected node. The wireless range r was set to 50m. We have set the other parameters as follows; the expected hop distance $H = 2$; the time to live of a copy of content $ttl(C^g) = 400sec.$; the ratio of average one-hop distance to wireless range $\alpha = 0.7$; simulation time was 1500sec. and the average speed $V_{ave} = 1.6m/s$.

As comparison, we have implemented a distribution method called *Intersection Distribution (ID)* method that does not utilize flow information. In ID method, a mobile node broadcasts the content with probability P_{send} (this probability is called *broadcast probability*) at the first intersection after the reception of the content. Both MoDA and ID methods use the broadcast model described in Section II-D, but ID method sets the reception probability such that at least two nodes store the content for every flow that contains the intersection. This reception probability is referred to as P (the value of P differs for each intersection). Here, one may think that such random-based distribution with carefully tuned parameters such as the broadcast probability and reception probability may outperform MoDA, or perform as well as MoDA. To see the truth, we have used the ID method with combinations of different broadcast probabilities and reception probabilities. Hereafter the ID method with broadcast probability $P_{send} = a$ and reception probability $P_{rcv} = b$ is denoted as $ID[a, b]$. We have used two values, 0.6 and 0.2 for P_{send}, and four parameters, P, $0.75P$, $0.5P$ and 0.2 (static value) for P_{rcv}. We believe that the combinations of these values uniformly cover the possible parameter settings. We have implemented *Direct Source (DS)* method where no copy is placed on ad hoc networks and every request is forwarded to the source node. This is used just to see how many hops are required to reach the source node.

In each simulation scenario, at time 200 sec., the source node started broadcasting the data for every $\frac{2 \cdot H \cdot \alpha \cdot r}{V_{ave}}$ seconds. We have measured the performance from time 500 seconds to time 1500 seconds. We have conducted 5 simulations to obtain average values.

b) Results: We have measured (i) the numbers of hops to reach the closest content holders, (ii) the numbers of content holders, and (iii) the numbers of broadcasts. These values are shown in Fig. 4, Fig. 5 and Fig. 6, respectively. Fig. 4 indicates that except DS method and ID methods with some parameter values, MoDA and several ID methods achieved almost similar hop counts, which are not greater than the expected hop counts $H = 2$ in average. Then we would like to see how many content holders and how many broadcasts are required in order to realize the hop counts shown in Fig. 4. From the results in Fig. 5 and Fig. 6 as well as Fig. 4, there is no ID method that outperformed MoDA in terms of all the three metrics, *e.g.* some need more broadcast messages and some others needs more hops. As a result, we can say that our distribution strategy could achieve the designated hop counts, with smaller numbers of content holders and broadcast messages. This means that our strategy could place the copies of contents in an efficient way, by utilizing flow information of mobile nodes.

IV. CONCLUSION

In this paper, we have shown a new concept to utilize the mobile nodes' moving flows to realize efficient maintenance of content replicas on MANETs. As far as we know, this is the first approach that presents the concept and protocol design.

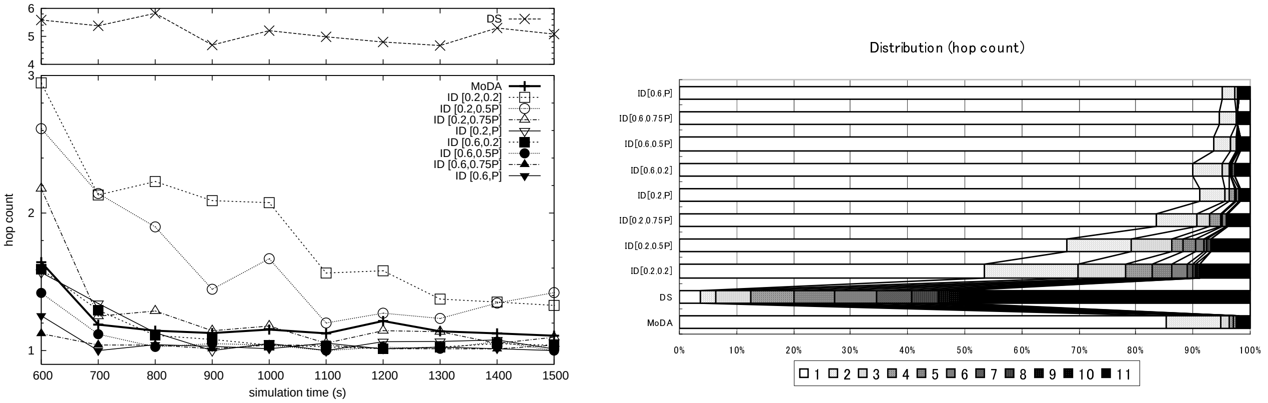
REFERENCES

[1] K. Maeda, K. Sato, K. Konishi, A. Yamasaki, A. Uchiyama, H. Yamaguchi, K. Yasumoto, and T. Higashino, "Getting urban pedestrian flow from simple observation: Realistic mobility generation in wireless network simulation," in *Proc. of ACM/IEEE MSWiM2005*, 2005, pp. 151–158.

[2] G. Karumanchi, S. Muralidharan, and R. Prakash, "Information dissemination in partitionable mobile ad hoc networks," in *Proc. of IEEE Symp. on Reliable Distributed Systems*, 2000.

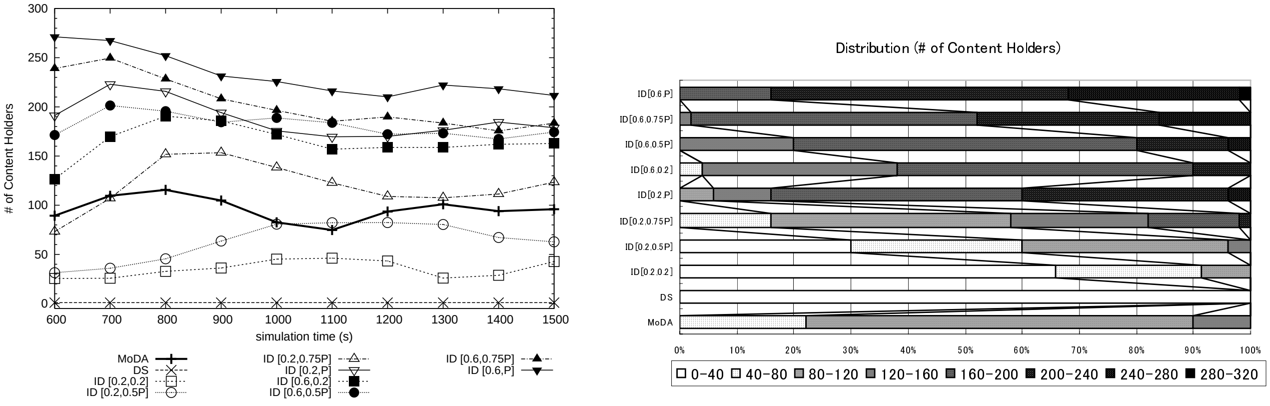
[3] T. Hara, "Replica allocation methods in ad hoc networks with data update," *Mobile Networks and Applications*, vol. 8, no. 4, pp. 343 – 354, 2003.

[4] Q. Li and D. Rus, "Sending messages to mobile users in disconnected ad-hoc wireless networks," in *Proc. of Mobicom*, 2000, pp. 44–55.



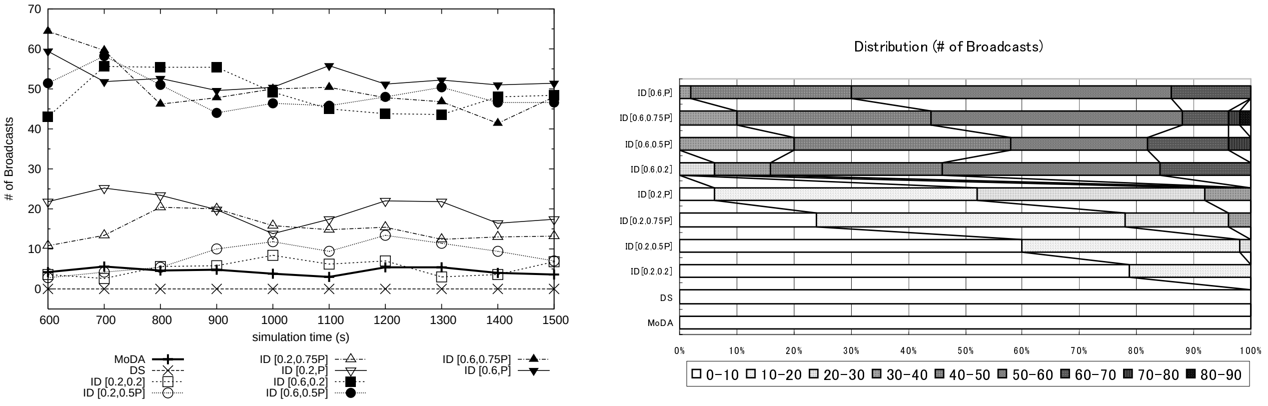
(a) Average Values as Function of Time (b) Distribution

Fig. 4. The Numbers of Hops to Closest Content Holders



(a) Average Values as Function of Time (b) Distribution

Fig. 5. The Numbers of Content Holders



(a) Average Values as Function of Time (b) Distribution

Fig. 6. The Numbers of Broadcasts

[5] D. Niculescu and B. Nath, "Trajectory based forwarding and its applications," in *Proc. of Mobicom*, 2003.

[6] H. Wu, R. Fujimoto, R. Guensler, and M. Hunter, "MDDV: a mobility-centric data dissemination algorithm for vehicular networks," in *Proc. of ACM VANET 2004*, 2004, pp. 47–56.

[7] K. Konishi, K. Maeda, K. Sato, A. Yamasaki, H. Yamaguchi, K. Yasumoto, and T. Higashino, "MobiREAL simulator – evaluating MANET applications in real environments –," in *Proc. of IEEE MASCOTS2005*, 2005, pp. 499–502.

[8] "MobiREAL web page," http://www.mobireal.net.

[9] M. Mauve, J. Widmer, and H. Hartenstein, "A survey on position-based routing in mobile ad hoc networks," *IEEE Network Magazine*, vol. 15, no. 6, pp. 30–39, 2001.

[10] S. S. Skiena, *The Algorithm Design Manual*. Springer-Verlag, 1998.

[11] T. Tracy, B. Jeff, and D. Vanessa, "A survey of mobility models for ad hoc network research," *Wireless Communication & Mobile Computing*, vol. 2, no. 5, pp. 483 – 502, 2002.

[12] A. Yamasaki, H. Yamaguchi, S. Kusumoto, and T. Higashino, "MoDA: Mobility-aware data management on mobile wireless networks," white paper, available online: http://www-sdl.ist.osaka-u.ac.jp/~h-yamagu/techreport/.