Replica arrangement for location dependent data in consideration of network partition in ad hoc networks

Gen Tsuchida* and
Susumu Ishihara

Graduate School of Science and Technology,
Shizuoka University,
3-5-1 Johoku, Naka-ku, Hamamatsu-shi
Shizuoka, 432-8561 Japan
Fax: +81-53-478-1265
E-mail: gen@ishilab.net
E-mail: gen.tsuchida.qc@hitachi.com
E-mail: ishihara@ishilab.net
*Corresponding author

Abstract: We propose link-aware and density based replica arrangement (LDR) method for dynamical arrangement of replicas of location dependent data that addresses the difficulties encountered by mobile hosts in ad hoc networks when trying to access data on other hosts due to the movement of hosts, obstacles, etc. In this LDR method, replicas of location dependent data items are placed on mobile hosts neighbouring branch points in the ad hoc network and the density of replicas is controlled in accordance with the degree of the network topology. Performance evaluation by simulation showed that the LDR method performs better than other methods when the requesting host is distant from the location where the target data item is associated and the host is moving quickly.

Keywords: ad hoc network; location dependent data; geocast; replica arrangement; network partition; connectivity; communication network; distributed system

Reference to this paper should be made as follows: Tsuchida, G. and Ishihara, S. (2009) 'Replica arrangement for location dependent data in consideration of network partition in ad hoc networks', *Int. J. Communication Networks and Distributed Systems*, Vol. 2, No. 4, pp. 401–423.

Biographical Notes: Gen Tsuchida received his BE and ME in System Engineering and his PhD in Information Science and Technology from Shizuoka University in 2004, 2006 and 2009, respectively. From 2007 to 2009, he was a Research Fellow of Japan Society for a Promotion of Science. Since 2009, he has joined Hitachi Ltd. His research interests include mobile ad hoc networks. He is a member of IEICE and IPSJ.

Susumu Ishihara received his BE, ME and PhD in Electronics Engineering from Nagoya University in 1994 and 1999. From 1998 to 1999, he was a JSPS Special Researcher. Since 1999, he has been with Shizuoka University, where he is an Associate Professor of the Graduate School of Science and Technology. He was a Visiting Researcher at the University of California, Irvine in 2008. His research interests are in the areas of mobile TCP/IP networking, mobile ad hoc networking and sensor networks. He is a member of IEEE, IEEE Communication Society, IEEE Computer Society, ACM SIGMOBILE, ACM SIGCOMM, IEICE and IPSJ.

1 Introduction

Systems providing location dependent information are becoming widely deployed. For example, there are pedestrian navigation systems using cell phones and car navigation systems using vehicle information and communication systems (VICS (1996)). Therefore, the management of geographical information has been attracting much interest. Maihofer et al. (2005) proposed a concept, called “abiding geocast”, in which messages are delivered to all hosts within a particular region within a certain period of time. Services like position-based advertising and position-based publish-and-subscribe and many other location-based services may profit from abiding geocast. Yashiro (2006) proposed a system for providing location information using a “nomadic agent (NA)”, which is a mobile agent that stays within its target area with certain information. A user desiring that information retrieves it by using the NA system.

Services sharing location dependent information on wireless ad hoc networks (Broch et al. (1998)) comprising only mobile hosts are expected to become widely available because they do not require large and costly infrastructure. Such services include ones that exchange traffic information, shopping information, or natural disaster information between mobile hosts. We call such systems “systems for sharing objects with location information on ad hoc networks (SOLA)” (Fig. 1). The location dependent data handled by SOLA are generated by mobile hosts and are associated with the locations where they were generated.

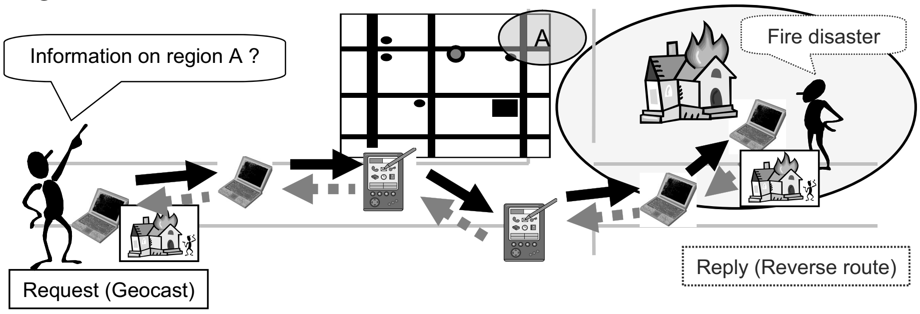


Figure 1 Sharing objects with location information on ad-hoc networks (SOLA)

The sharing of data items between mobile hosts in wireless ad hoc networks is difficult because the movement of the hosts can cause link disconnections, which degrades data accessibility. This hampers the practical use of ad hoc networks. The accessibility to the data objects held by a mobile host from other hosts must be maintained even if link disconnections occur between these hosts. One approach to solving this problem is to distribute replicas of data items to other mobile hosts and to arrange them as necessary following movement of the hosts. For example, Hara proposed a method based on the assumption that all mobile hosts in the network know the probability of access to each data item and that the probabilities do not change (Hara (2001)). Hara extended this method to support data update (Hara (2002)), to support selection of replica-holding hosts using link state information (Hara et al. (2003)), and (Shinohara et al. (2007)). Yin proposed a cache algorithm for data maintained by fixed servers on ad hoc networks (Yin and Cao (2006)).

We proposed the “skip copy (SC)” (Tsuchida et al. (2005)) and “adaptive skip copy (ASC)” (Tsuchida et al. (2006)) methods for maintaining and sharing location dependent data collected by sensors and requested by geocasting (Ko and Vaidya (2002)) request messages on serverless ad hoc networks. Such data would be used in personal and car navigation systems, for cooperative work during disaster relief, etc. These methods are based on the assumption that mobile hosts do not know the IDs of the hosts holding a replica of the requested data item. The mobile hosts send request messages to multiple hosts near the location to which the requested data item is associated in order to access it. The replicas of requested date items are placed on only mobile hosts on the reply route. However, if a mobile host on a previous reply route moves with holding a replica of the data item, the replica may no longer be accessible, which would increase the amount of request traffic and lengthen the request routes.

To overcome this problem, we have now developed a method for dynamically arranging replicas. In this “link-aware and density-based replica arrangement (LDR)” method, replicas are placed on mobile hosts that neighbouring branch points in the ad hoc network. The density of replicas is controlled in accordance with the host density, the number of neighbour hosts, and the number of hops from a replica-holding host on the reply route. Performance evaluation showed that the LDR method has a higher access success rate than other methods especially when the requesting host is moving quickly and is far from the destination host.

The remainder of the paper is organized as follows. Section 2 gives a brief overview of the SC and ASC methods. In section 3, we introduce the LDR method. Section 4 presents the simulation conditions, and in section 5 we present some of the results and discuss the characteristics of the LDR method. Finally, we summarize the key points and mention future work in section 6.

2 Replica Distribution Method for Location Dependent Data

In this description of our previously proposed replica distribution method for location dependent data, we focus on a system in which mobile hosts share location dependent data obtained by cameras, sensors, etc. over wireless ad hoc networks, i.e. a “system for sharing objects with location information on ad-hoc networks (SOLA)” (Fig. 1). We assume the location dependent data (e.g. vehicle traffic information, disaster-related information) handled by SOLA are generated by mobile hosts and associated with the locations at which they were generated. Our SC and ASC methods for distributing replicas were designed for SOLA environments.

2.1 Assumptions

There are six specific assumptions regarding SOLA.

- Mobile hosts collect and share location dependent information on an ad hoc network.
- Each host tracks its location using GPS.
- There is no specific data server in the ad hoc network, so a host does not know which host has a particular data item.

- Each host obtains data items from other mobile hosts when it does not have suitable ones locally. When a host wanting to know information associated with a location does not have a data item about the location, it geocasts a request message to hosts near the location.
- Each host has a limited amount of data storage.
- Each mobile host generates data items associated with its current position, each annotated with the generation time and expiration time.

2.2 Skip Copy (SC) Method

In the SC method, replicas of location dependent data items collected by mobile hosts are distributed as follows. When a data item is generated on a mobile host, a replica of the data item is flooded to mobile hosts near the generator and distributed sparsely in accordance with the hop count c from the generator within the range R of replica distribution from the location where the item was generated, its "birth place". The density of the replicas is controlled using a skip parameter, s. If the hop count satisfies $c \mod s = 0$, the receiver holds it; otherwise, the receiver disposes of it after forwarding it to the neighbouring hosts. Replicas are dynamically arranged when they are forwarded as reply messages to requesting hosts. That is, hosts that forward the reply message hold the replica if hop count c_r between the replying host and the forwarding host satisfies $c_r \mod s_r = 0$ and the forwarding host is within R_r of the birthplace of the data item.

With the SC method, replicas of a data item are distributed sparsely around the location where they are generated, and they stay near that location even if hosts holding them move far away. In this way, location dependent data can be accessed using geocasting even if there are no fixed servers. Figure 2 shows an example of replica arrangement for $s_r = 2$.

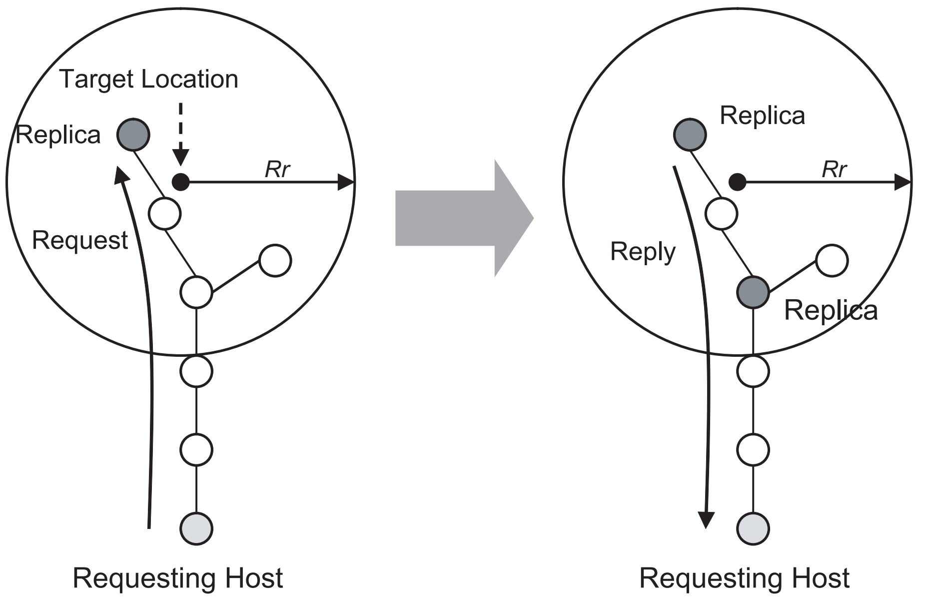


Figure 2 Replica arrangement with SC method (s_r=2)

With the SC method, if the requesting host is at a distance from the receiving host, the request message tends to be lost and the response time tends to be long because the replicas are too far from the requesting host. To solve this problem, we developed the ASC method.

2.3 Adaptive Skip Copy (ASC) Method

The ASC method was designed to reduce the response time and to achieve high accessibility to location dependent data in environments where hosts frequently request data items generated at a point distant from themselves. With the ASC method, replicas of data items are distributed sparsely around the location at which they were generated, in a fashion similar to that with the SC method. Additionally, the replicas are dynamically arranged during a data access in almost the same manner as with the SC method. The ASC method arrangement is based on the hop count between the requesting and replying hosts, while the SC one is based on the hop count between a forwarding host and the replying host and the distance between the forwarding host and the birthplace of the data item. In the ASC method, skip parameter s_r is calculated using the following equation, while the SC method uses a constant value of s_r.

(1) $$s_r = \max\left(\left\lceil \frac{h(P_r, P_a)}{\alpha} \right\rceil, s_{\min}\right),$$

where α is the number of replica-holding hosts including the requesting host and $h(P_r, P_a)$ is the hop count between requesting host P_r and replying host P_a. This equation means that the nearer the replying host to the requesting host, the smaller the arrangement skip parameter. The minimum skip parameter, $s_{\min}$, is used to control the usage of storage on mobile hosts. Figure 3 shows an example of replica arrangement for $\alpha = 2$.

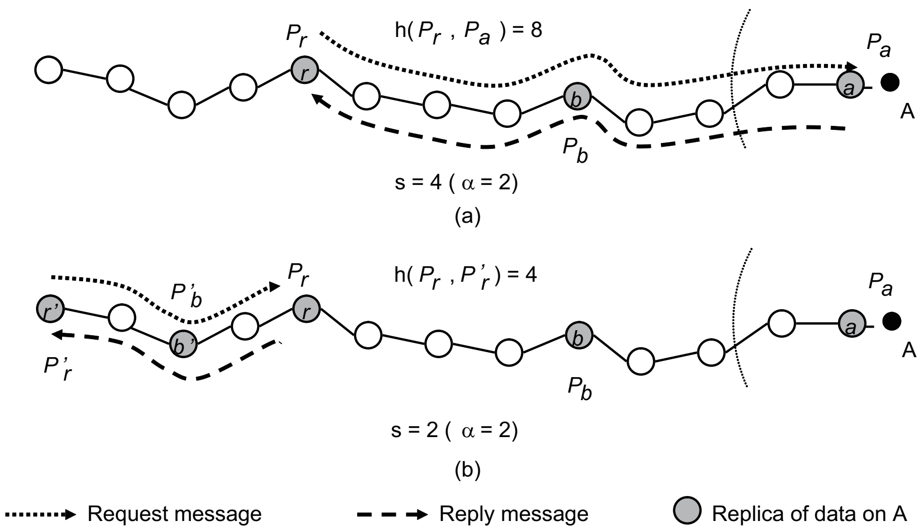


Figure 3 Replica arrangement with ASC method

3 Link-aware and Density-based Replica Arrangement (LDR) Method

In the LDR method, replicas are placed on mobile hosts neighbouring branch points on the reply route when the reply message is forwarded, and the density of replicas is controlled in accordance with the host density, the number of neighbour hosts, and the hop count from a replica-holding host on the reply route. The branch point is a critical host in ad hoc network. The branch point host has at least three neighbour hosts (fig. 4 host *c*). In fig. 4, if the branch point host *c* moves away after the arangement of replica, other hosts cannot use the replica and connect each other. To mitigate the impact of the movement of a critical host, replicas are arranged on mobile hosts neighbouring branch points in the LDR method. Figure 5 shows an example arrangement of replicas.

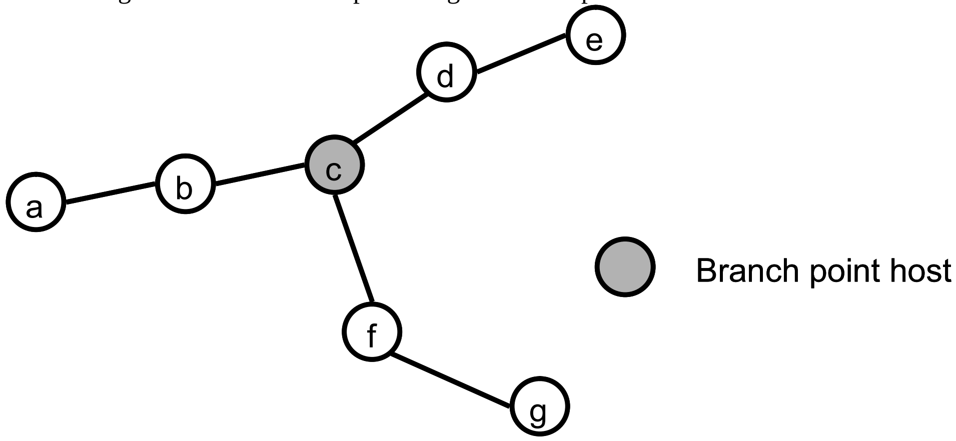


Figure 4 Branch point in ad hoc network

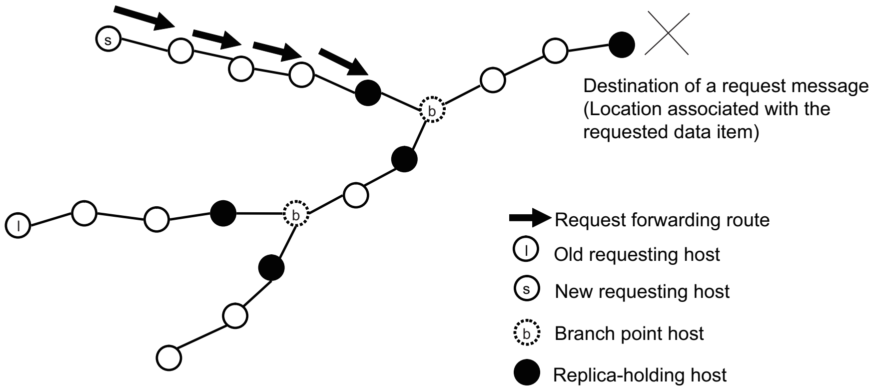


Figure 5 Replica arrangement with LDR method

3.1 Assumptions

In the LDR method, each host periodically sends "hello" messages to its neighbouring hosts so that they know its current position. In SOLA, request messages are forwarded by location based multicasting (LBM) (Ko and Vaidya (2002)). Nodes surrounded by a rectangle region which includes both the sender and the destination position, forwards messages by broadcast. When a node forwards a request message, it adds its ID to the forwarders list in the header of the message. If a node which has the request data item receives the request message, it sends back the data as a reply message. The reply message is forwarded along the reverse route of the request message. Each forwarding nodes specifies the next forwarder according to the list of forwarders included in the request message and the reply message. Messages are forwarded by a series of broadcasts so that the neighbor nodes of the specified next forwarding node can receive and store them if needed. Each reply message includes the data item, the ID of the hosts on the reply route, hop count (h) from the previous replica-storing host, a replica conservation flag (R_H), and the ID of the next reply-forwarding host (i_n). If a node receives a reply message with $R_H =$ "on" and it is specified as the next reply-forwarding host i_n, it stores the replica of the data item delivered by the reply message. The initial state of R_H is "off".

3.2 Information Dissemination Algorithm

The first replying host sends a reply message (with a replica of the data item) with replica conservation flag $R_H = \text{off}$ and $h = 0$.

When host i which is the next reply forwarding host receives the reply message, it increments h and forwards the message to its neighbours if it is on the reply-forwarding route. If the R_H of the message is off and h and $|\mathbf{M_i}|$ satisfy the following conditions, i sets R_H to "on" so that the next forwarding host to store the replica.

$$h \geq H_{\min} \tag{2}$$

$$|\mathbf{M_i}| \geq 3 \tag{3}$$

$H_{\min}$ is a positive constant, and $\mathbf{M_i}$ is the list of the neighbours of host i. Then host i forwards the reply message (replica). The branch point host has at least three neighbour hosts. Thus we need a condition (3) $|\mathbf{M_i}| \geq 3$. However, if many hosts satisfy the condition (3), too many replicas may be arranged. To avoid this problem, we use condition (2) to control the density of replica.

If any branch points on the reply-forwarding route do not exist, no replicas are arranged on hosts on the route. To avoid this condition, we introduce a parameter C_r the maximum replica interval into the LDR method. If $h = C_r$, host i stores the replica and sets $h = 0$ before forwarding the replica. C_r is determined by the number of hops between the requesting and replying hosts, e.g. half the number of hops between them.

If R_H of the received message is "on", host i stores the replica in its local storage, sets $h = 0$ and $R_H = \text{off}$, and forwards the reply message.

In the LDR method, replicas of data items are placed not only on hosts on the reply route but also on neighbour hosts of the hosts on the reply route. Neighbour host k of a reply-forwarded host i may receive the reply message because reply messages are forwarded using broadcast. When host k receives the reply message, it checks the list of its neighbours $\mathbf{M_k}$ to determine whether it stores the replica. If the next reply-forwarding host i_n is included list $\mathbf{M_k}$ ($i_n \in \mathbf{M_k}$), host k does not store the replica. Otherwise, host k stores the replica with probability $P(k)$.

$$P(k) = \begin{cases} 1 & (|\mathbf{M_k}| < D_N) \\ k_M/|\mathbf{M_k}| - 1 & (|\mathbf{M_k}| \geq D_N), \end{cases} \tag{4}$$

where D_N is a positive constant such that $D_N \geq 2$ and k_M is a positive constant.

3.3 Example

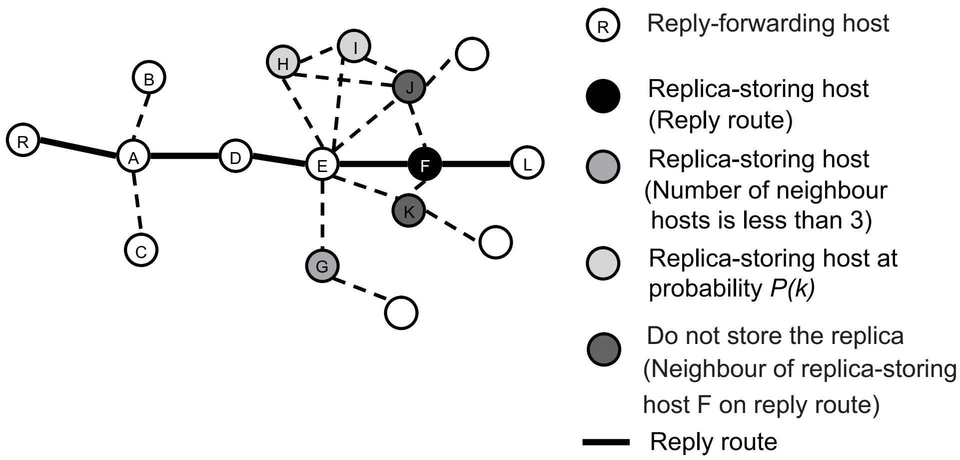


Figure 6 Example of replica arrangement with LDR method ($H_{\min} = 3$, $D_N = 3$)

Figure 6 shows an example of replica arrangement with the LDR method for $D_N = 3$ and $H_{\min} = 3$. Replying host R sends a reply message (with a replica of the data item) with replica conservation flag $R_H = \text{off}$. Host E, which satisfies conditions (2) and (3), forwards the reply message with $R_H = \text{on}$. Host F, which receives the reply message, stores the replica because $R_H = \text{on}$ and F is specified as a reply-forwarding host. Then F forwards the replica with $R_H = \text{off}$ and $h = 0$. Though hosts J and K receive the same message from host E, they do not store the replica because one of their neighbouring hosts, F, is a reply-forwarding host, and they are not specified as a reply-forwarding host. Host G stores a copy of the replica because F is not its neighbour, and the number of its neighbouring hosts, $|M_G|$, is less than D_N. Hosts H and I store a copy of the replica with probabilities $P(H)$ and $P(I)$, respectively, because they are not connected to host F, and $|M_H|$ and $|M_i|$ equal D_N.

4 Simulation

We evaluated the performance of the LDR method by simulation using the JiST/SWANS network simulator (JiST/SWANS (2004)).

4.1 Hosts

First, 100–600 mobile hosts were placed at random in a 2000 × 1000-m field divided into 200 square sub-areas, 100 m per side. Following the random way-point model, each host moved within the simulation field. The speed of each host was set randomly at between 1 and 2 m/s or between 1 and 20 m/s, and the pause time defined in the model was 3 s. Each host could hold ten data items. If a host specified as a replica holder was holding ten data items when it received a new replica to store, it disposed of the replica for the oldest data item to make room for the new one.

4.2 Data Generation

Each host generated a new data item associated with the centre of its current sub-area positioned in accordance with a Poisson model with a mean interval of 200 s. The reply and request messages were 1500 and 128 bytes large, respectively, including the UDP and IP headers. The time to live (TTL) for each data item was 500 s.

4.3 Data Request

We used three models to generate request messages from the hosts: a priority–on–neighbourhood model, a flat–priority model (Tsuchida et al. (2005)), and a priority–on–distant model.

In the priority–on–neighbourhood and flat–priority models, the data access pattern is based on a *Zipf–like* distribution (Zipf (1949)), which is frequently used to model non-uniform distributions (Yin and Cao (2006)). In this model, mobile hosts choose the areas to which they send requests in accordance with the following probability. Each mobile host chooses one of the 200 sub-areas as the area in which it is interested. In the *Zipf–like* distribution, the access probability for the data item d_i associated with the ith sub-area is represented as

(5) $$P_r(i) = \frac{1/{r_i}^{\theta}}{\sum_{j=1}^{200} 1/{r_j}^{\theta}} \quad (0 \leq \theta \leq 1),$$

where r_i is the distance between the current position of the requesting mobile host and the centre of the ith sub-area. When $\theta = 1$, equation (5) follows a strict Zipf distribution. When $\theta = 0$, it follows a uniform distribution. A larger θ results in a more “skewed” access distribution. We set θ to 1 in the priority–on–neighbourhood model and θ to 0 in the flat–priority model. If 200 data items have already been acquired by mobile hosts, ith data item d_i $(i = 1, 2, \cdots, 200)$ is chosen with probability $P_r(i)$

In the priority-on-distant model, each host at the coordinates $(0, y)$ to $(400, y)$ generates request messages for data items generated at the coordinates $(1600, y)$ to $(2000, y)$ $(0 \leq y \leq 1000)$. Similarly, hosts at the coordinates $(1600, y)$ to $(2000, y)$ generate request messages for data generated at the coordinates $(0, y)$ to $(400, y)$.

The generation of request messages follows a Poisson model with a mean interval of 100 s. Request messages are forwarded by location based multicasting (LBM) (Ko and Vaidya (2002)) so that the request messages reach the target sub-area.

Proactive protocols (e.g. Clausen and Jacquet (2003); Perkins and Bhagwat (1994)) can result in higher overhead due to continuous route updating even if there is no demand for data. Reactive protocols (e.g. Johnson et al. (2007); Perkins et al. (2003)) may have higher latency because a route from a source host to a destination host is found only when the source host attempts to send to the destination host. LBM has shorter latency than reactive protocols and results in lower overhead than proactive protocols because it does not send route control packets. Thus we used LBM.

4.4 Evaluation Metrics

We used the following performance metrics for the evaluation of the LDR method.

- Access Success rate A_S

  (6) $$A_S = \frac{A_C}{R_C},$$

  where R_C is the number of requests and A_C is the number of successful responses.

- Request traffic T_{reqeust} and reply traffic T_{reply}

  T_{request} and T_{reply} are the number of packets sent for the request and reply messages, respectively.

- Response Time T_D

  This is the average time from when a host sends request message to the time when it receives a response.

- Redundancy R_e

  This is the average ratio of the number of replicas to the number of original data items.

  (7) $$R_e = \frac{N_{\text{replica}}}{N_{\text{orig}}}$$

  N_{orig} and N_{replica} represent the total number of original data items and the total number of replica data items, respectively.

Table 1 shows the simulation parameters. We ran the simulation for 10,000 s in simulation time. The data collected in the first 1000 s were neglected to avoid the effect of the initial state.

Table 1 Simulation parameters

Parameter	**Default value**	**Range**
Number of square sub-areas	200	–
Data size [KB]	1.5	–
Number of hosts	600	100 – 600
Host speed, v [m/s]	20	2, 20
Pause time [s]	3	–
Bandwidth [Mbps]	11	–
Communication range [m]	105	–
Data generation interval [s]	200	–
Data request interval [s]	100	–
Hello interval [s]	30	29 – 31
Time to live, T_{TTL} [s]	500	–

5 Results and Discussion

We compared five replica arrangement methods.

- LDR: For the LDR method, the replica arrangement parameters were set as follows: $D_N = 3$; $H_{\min} = 2, 3$; $k_M = 1$; Hello interval = 30 s; Hello TTL = 30 s; C_r = half the number of hops between the requesting and replying hosts.
- ASC: For the ASC method, replica-holding hosts α was 2 or 3 and $s_{\min}$ was 2.
- SC: For the SC method, replica placement range R_r was 300 m and skip parameter s_r was 2.
- PATH: In the path replication method, which is used in unstructured pure peer-to-peer (P2P) networks (Cohen and Shenker (2002); Lv et al. (2002)), requested data items are replicated on all hosts along the data transmission path between the requesting and replying hosts.
- Owner: In the owner replication method, which is used in unstructured pure P2P networks (Cohen and Shenker (2002); Lv et al. (2002)), only the sender of the request stores the replica of the requested data item.

We did not compare our scheme with Hara's replica arrangement method (Hara (2001)) because it is based on the assumption that all mobile hosts on the network know the probability of data access to each data item. Here, the assumption is that they do not know the probability.

5.1 Effect of Mobile Host Speed

5.1.1 Access Success Rate

Figure 7 shows the access success rate, A_S, for various numbers of hosts when the flat–priority model ($\theta = 0$) was used. When the number of mobile hosts was less than 200, the rate was virtually the same with all the methods at both speeds. If the number of mobile hosts is small, request messages are more likely to be lost because of unstable host connectivity. Thus, the rate was uniformly low. When the number was more than 300, the rate with the LDR method was the highest, particularly for $v = 20$ m/s. When mobile hosts move quickly, the topology of the ad hoc network changes frequently. If a replica-holding host on a previous reply route moves, its replicas may no longer be accessible. In the other methods, replicas are placed only on hosts on the previous reply route. If the hosts move quickly, there is a greater probability of losing the connectivity between the requesting and replica-holding hosts. As a result, the rate is lower. In the LDR method, replicas of a data item are placed on not only hosts on the reply route but also the hosts neighbouring them. Thus, the connectivity between the replica holders and requesting host has a higher probability of being maintained even if host mobility is high. As a result, A_S with the LDR method is higher than that of the other methods. In contrast, when the hosts move slowly, the network topology changes infrequently. Thus, a requesting host can connect to replica-holding hosts on the reply route with high probability. This accounts for the small difference in A_S among the replica arrangement methods.

5.1.2 Request and Reply Traffic

Figure 8 shows the number of request messages, T_{request}, for various numbers of hosts when the flat–priority model ($\theta = 0$) was used. The total number of packets was virtually the same for all the methods at both speeds. We used LBM for geocasting the request messages. Because this scheme is based on flooding, even if a host replies to a request message and stops forwarding it, other hosts continue to forward it. Thus, the difference in the number of packets among the replica arrangement methods is small. If we used a unicast geocasting scheme, e.g. GPSR (Karp and Kung (2000)), the number of messages would be lower. Figure 9 shows the number of reply messages, T_{reply}, for various numbers of hosts when the flat–priority model ($\theta = 0$) was used. The number was the highest with the LDR method when the number of hosts was more than 400. This is because the number of successful requests with the LDR method is the largest, and the greater the number of successful accesses, the greater the number of reply messages.

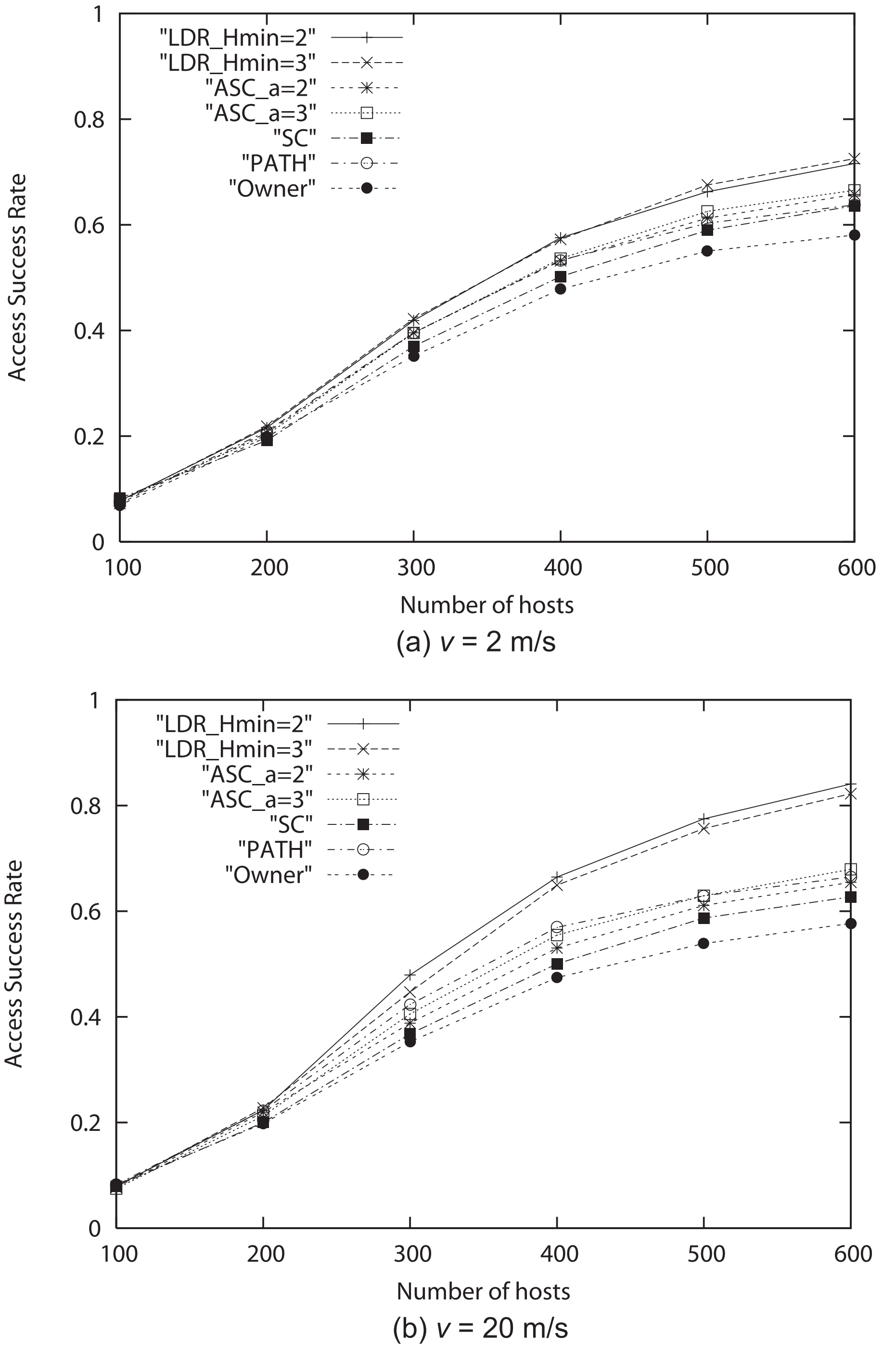
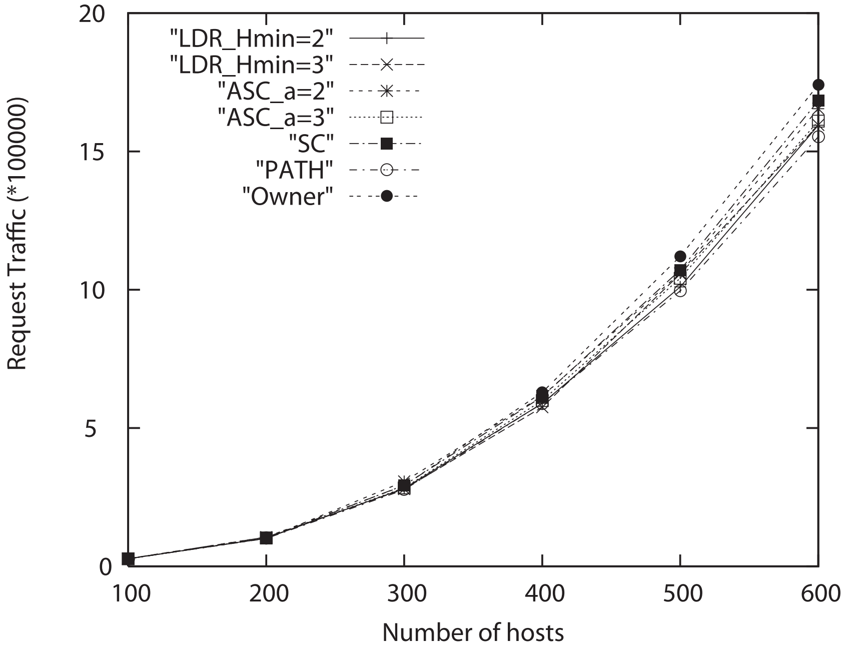
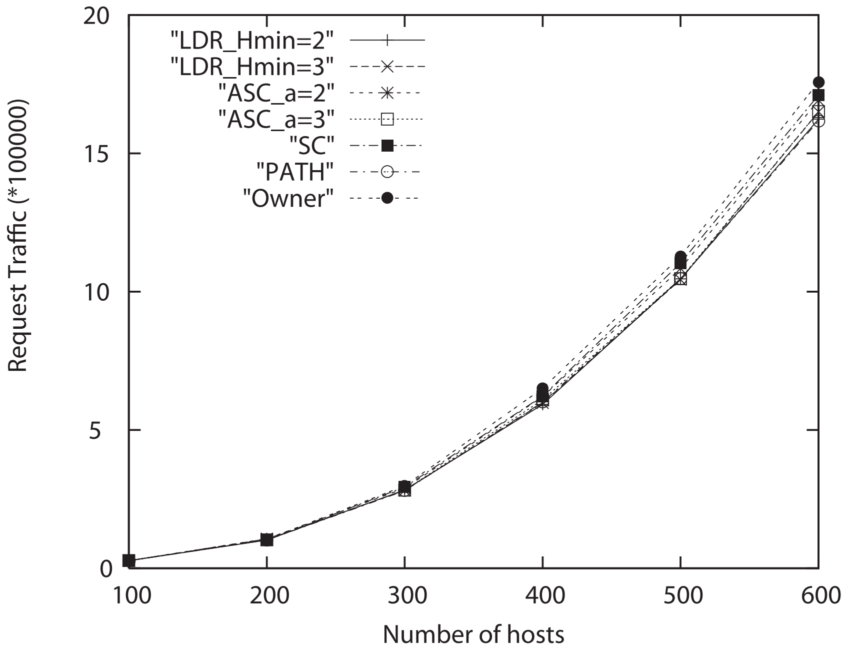


Figure 7 Access success rate vs. number of hosts (flat–priority model)



(a) v = 2 m/s



(b) v = 20 m/s

Figure 8 Number of request messages vs. number of hosts (flat–priority model)

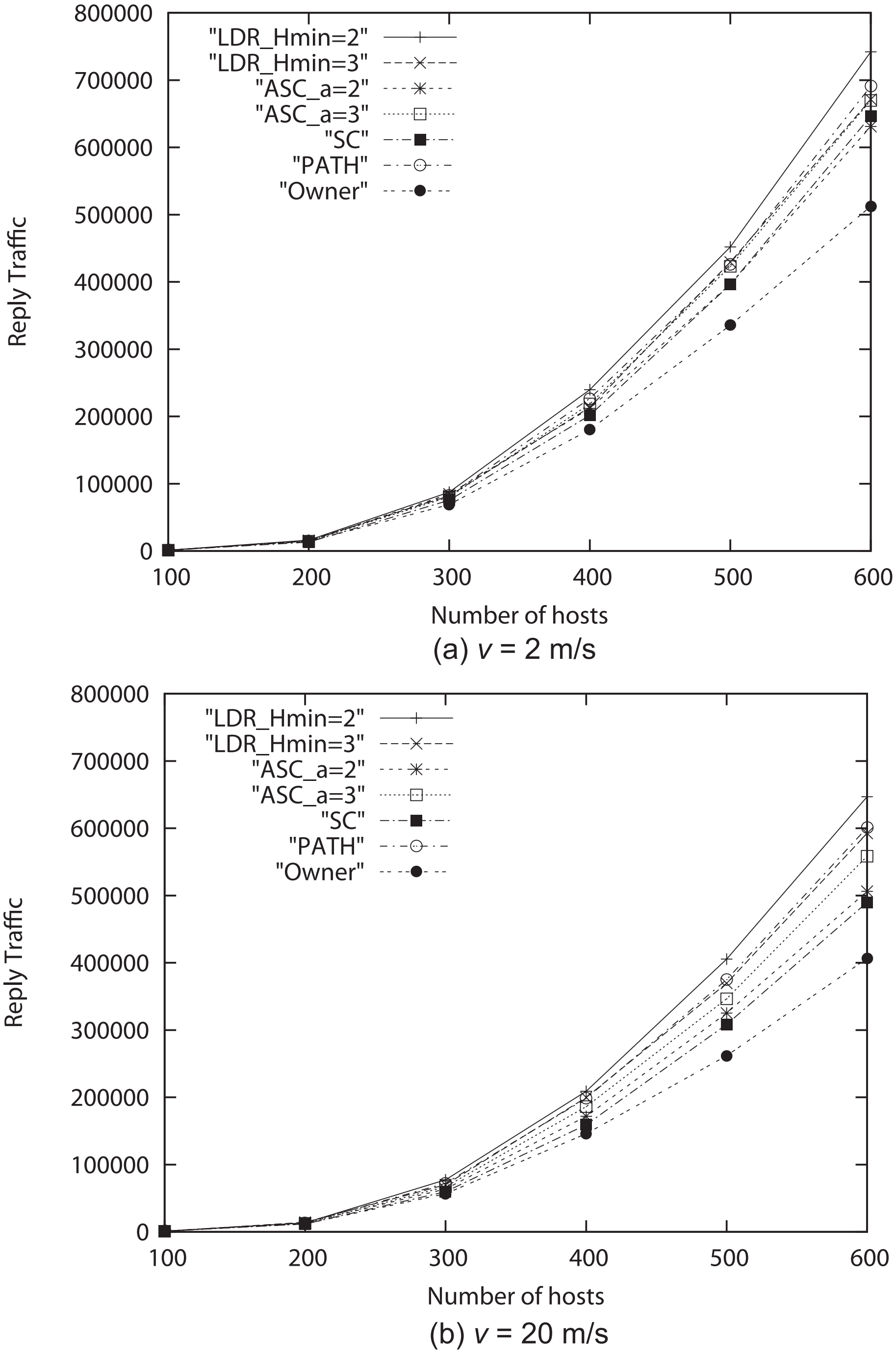
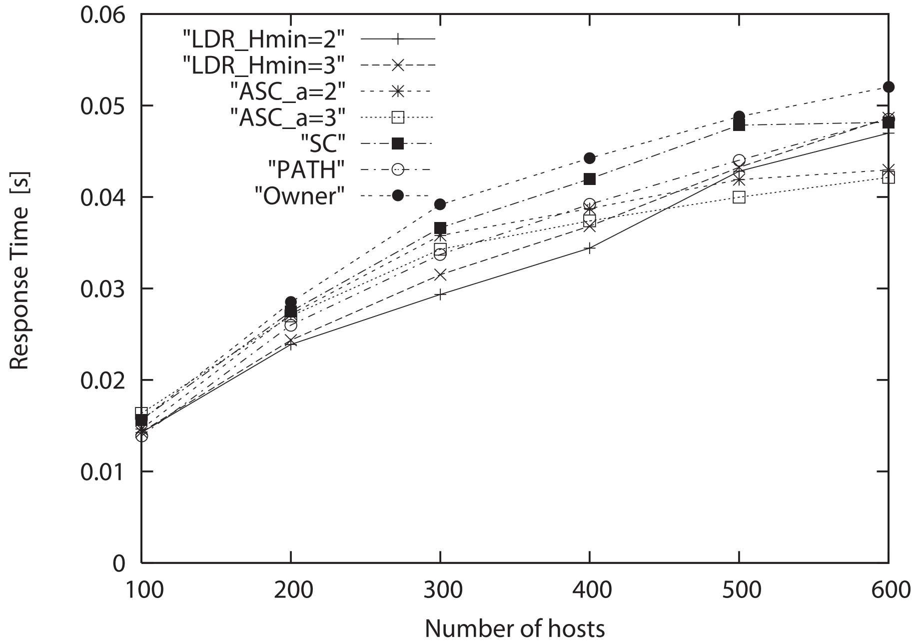
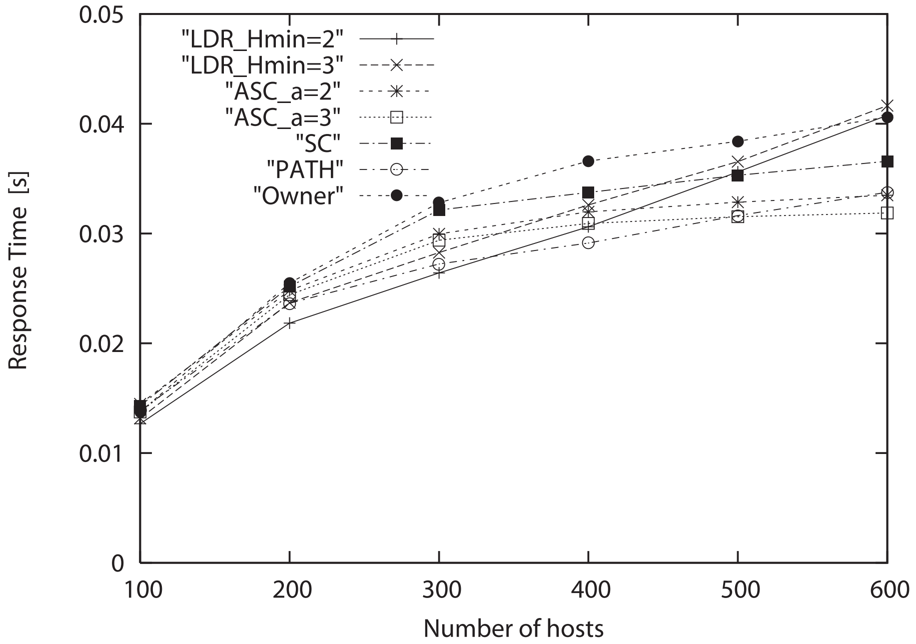


Figure 9 Number of reply packets vs. number of hosts (flat–priority model)



(a) v = 2 m/s



(b) v = 20 m/s

Figure 10 Response time vs. number of hosts (flat–priority model)

5.1.3 Response Time

Figure 10 shows the response time, T_D, for various number of hosts when the flat–priority model ($\theta = 0$) was used. The other parameters were set the same as in the previous section. When the number of hosts was small, the response time with the LDR method was the shortest. When the number was high, it was longer than with the ASC and path replication methods. The main reason for this is the higher access success rate with the LDR method. Because a larger number of hosts reply to a request with the LDR method, contention between their reply messages results in a longer response time. To obtain a high access success rate without increasing the response time in a high density network, we need to tune the parameters for the LDR method, e.g. D_N, which controls the probability of holding replicas at hosts neighbouring branch points. It would also be helpful to use the information for two-hop neighbours to determine the probability.

5.1.4 Redundancy

Figure 11 shows the redundancy, R_e, for various numbers of hosts when the flat–priority model ($\theta = 0$) was used. The redundancy with the LDR method was slightly lower than with the path replication method. In the LDR method, replicas of a data item are placed on not only hosts on the reply route but also the hosts neighbouring them, so there are many hosts holding the same replica. However, the density of replicas is controlled by the host density, the number of neighbour hosts, and the number of hops from a replica-holding host on the reply route. In the path replication method, replicas are placed on all reply-relying hosts. This results in high redundancy.

When the host density was small, the redundancy was lower than 1. This is because mobile hosts holding requested data items are not around the requesting hosts and request message is forwarded far away. Long distance forwarding of request messages result in failure of reaching replica holders and no reply message in high probability. As a result of this, replicas are not arranged.

5.2 Effect of data request model

Figure 12 shows the access success rate for various numbers of hosts when the priority–on–neighbourhood model ($\theta = 1$) was used. The other parameters were set the same as in the previous section. When hosts generate requests in accordance with the priority–on–neighbourhood model, the average distance between requesting hosts and the request destinations is shorter than with the flat–priority model. The connectivity between replica holders and requesting hosts is thus higher than with the flat priority model. The LDR method is designed to minimize loss of connectivity between the replica holders and requesting hosts. The superiority of the LDR method should thus be less with the priority–on–neighbourhood model. However, even with this model, the LDR method outperformed the other methods, while with the flat–priority model it was much more effective when the number of hosts was high. When the number of mobile hosts was more than 300, the access success rate with the LDR method was larger than with the other methods.

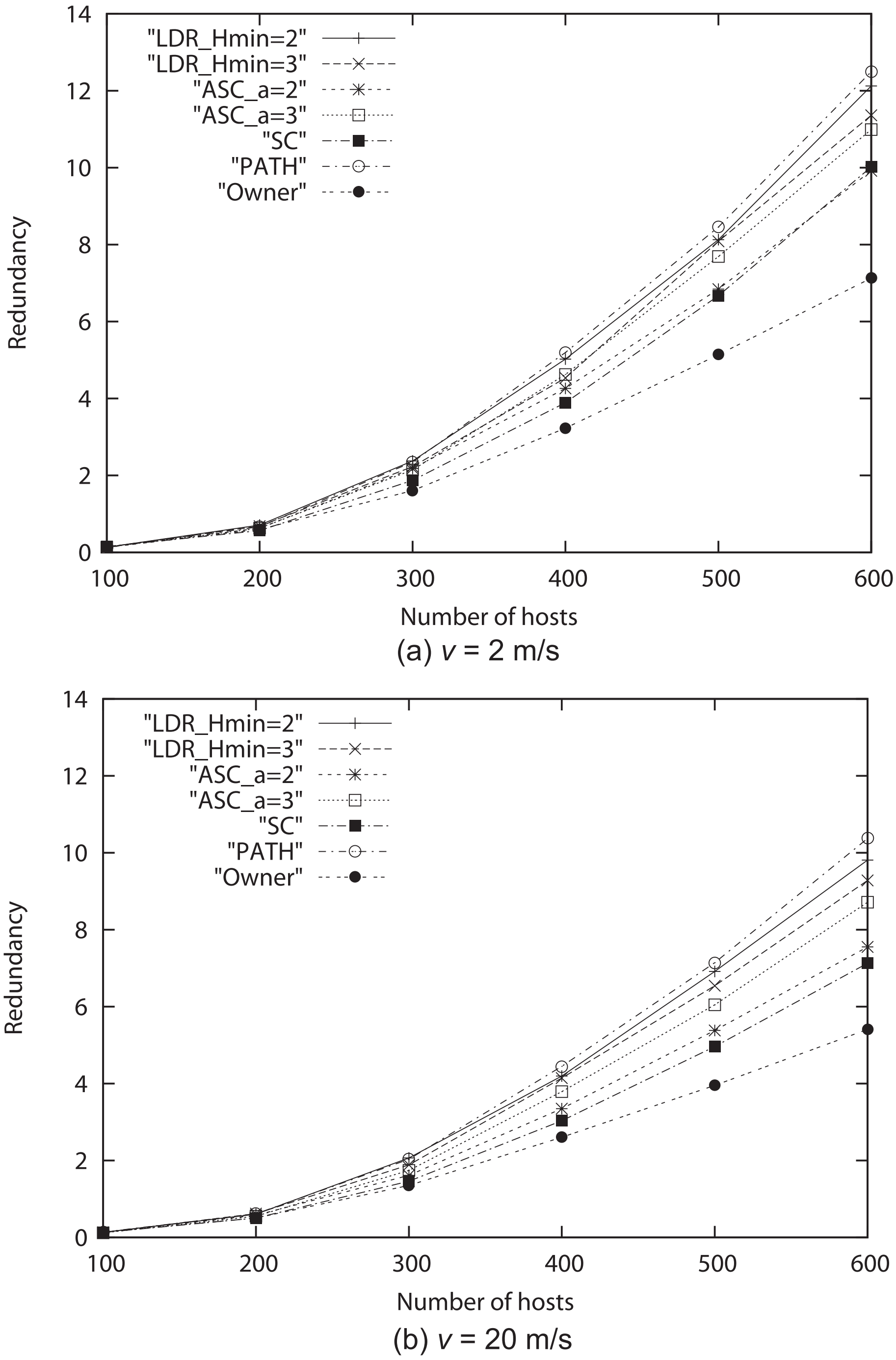


Figure 11 Redundancy vs. number of hosts (flat–priority model)

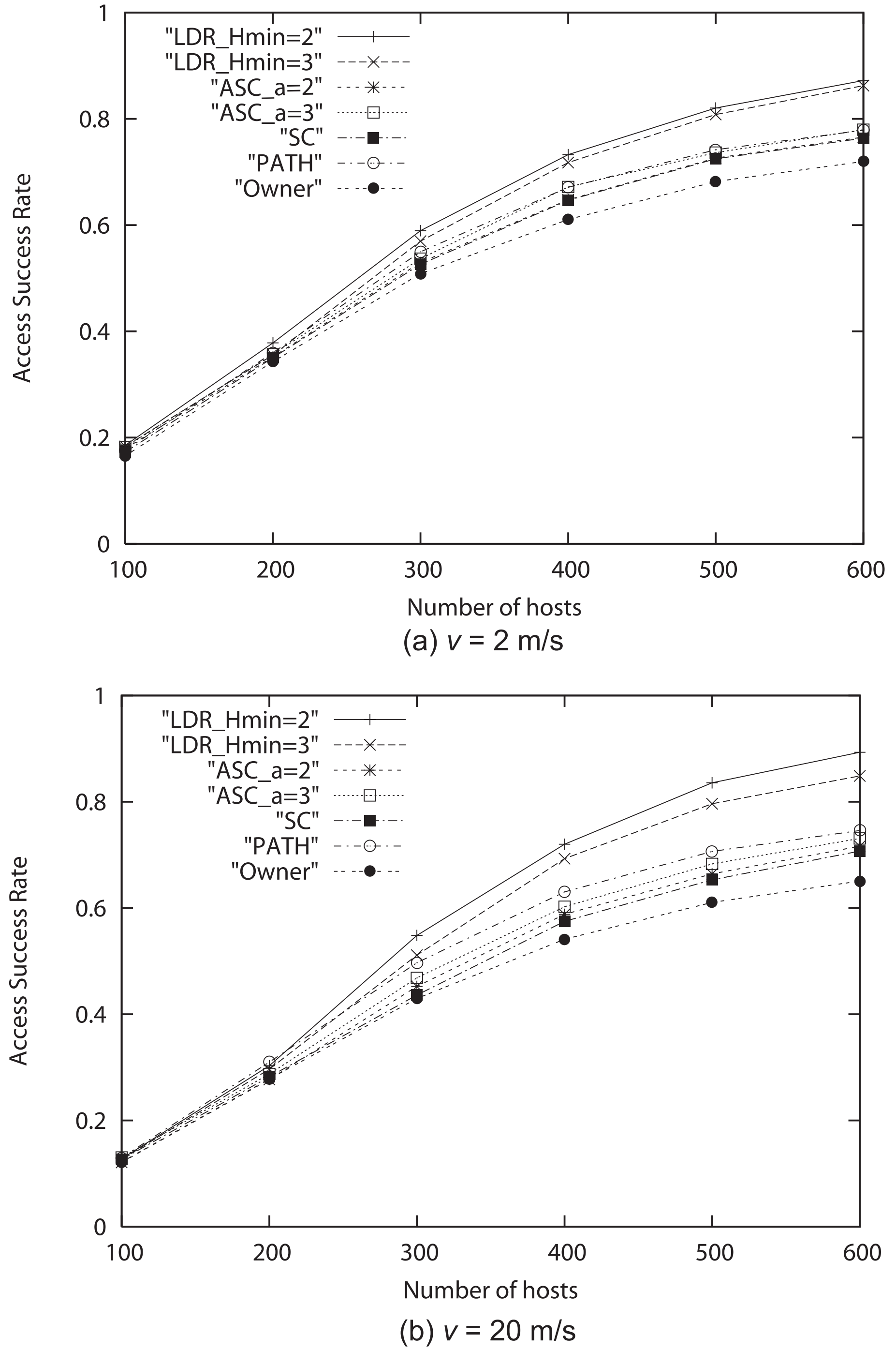


Figure 12 Access success rate vs. number of hosts (priority–on–neighbourhood model)

As shown in figure 12, with the priority–on–neighbourhood model and more than 400 mobile hosts, the access success rate was lower for $v = 20$ m/s than for $v = 2$ m/s. In this model, replicas are placed on hosts near the requesting host. However, if the hosts are moving quickly, the replica-holding hosts move away from the requesting host. As a result, A_S was smaller for $v = 20$ m/s than for $v = 2$ m/s with the priority–on–neighbourhood model.

With the flat–priority model (Fig. 7), replicas are located farther from the requesting host than with the priority–on–neighbourhood model. If the hosts are moving quickly, the replica-holding hosts are closer to the requesting host. However, connectivity between them is not stable if they are moving quickly. As a result, except for the LDR method, the average success rate for $v = 20$ m/s was the same as that for $v = 2$ m/s. With the LDR method, if the hosts are moving quickly, many replica holding-mobile hosts are widely spread because not only hosts on the reply route but also hosts neighbouring hosts on the reply route hold the replica. As a result, if the hosts are moving quickly, the success rate with the LDR method is high.

Figure 13 shows the access success rate for various numbers of hosts when the priority-on model was used. The other parameters were set the same as in the previous section. When hosts generate requests in accordance with the priority–on–distant model, the average distance between requesting hosts and request destinations is longer than when the flat–priority model is used. The connectivity between replica holders and requesting hosts is thus smaller than with the flat–priority model. When the host speeds are low, the LDR method has the lowest A_S. When the speeds are high, is has the highest A_S.

When the speeds are low, the network topology changes infrequently. Replicas on the reply route can thus be reached from a requesting host with high probability. However, with the LDR method, many replicas are arranged on hosts neighbouring branch points on the reply route, and few replicas are arranged on the reply route. As a result, the access success rate with the LDR method is lower than with the other methods. When the speeds are high, the network topology changes frequently. If a replica-holding host on a previous reply route moves, its replicas may no longer be accessible. In the other methods, replicas are placed only on hosts on the reply route. With the LDR method, replicas of a data item are placed on not only on hosts the reply route but also their neighbouring hosts. Thus, the connectivity between the replica holders and requesting host has a greater probability of being maintained even if host mobility is high. Adding to this, when the speed is high, replica holding hosts spread widely in the field in a short period of time. As a result, the access success rate with the LDR method is the highest.

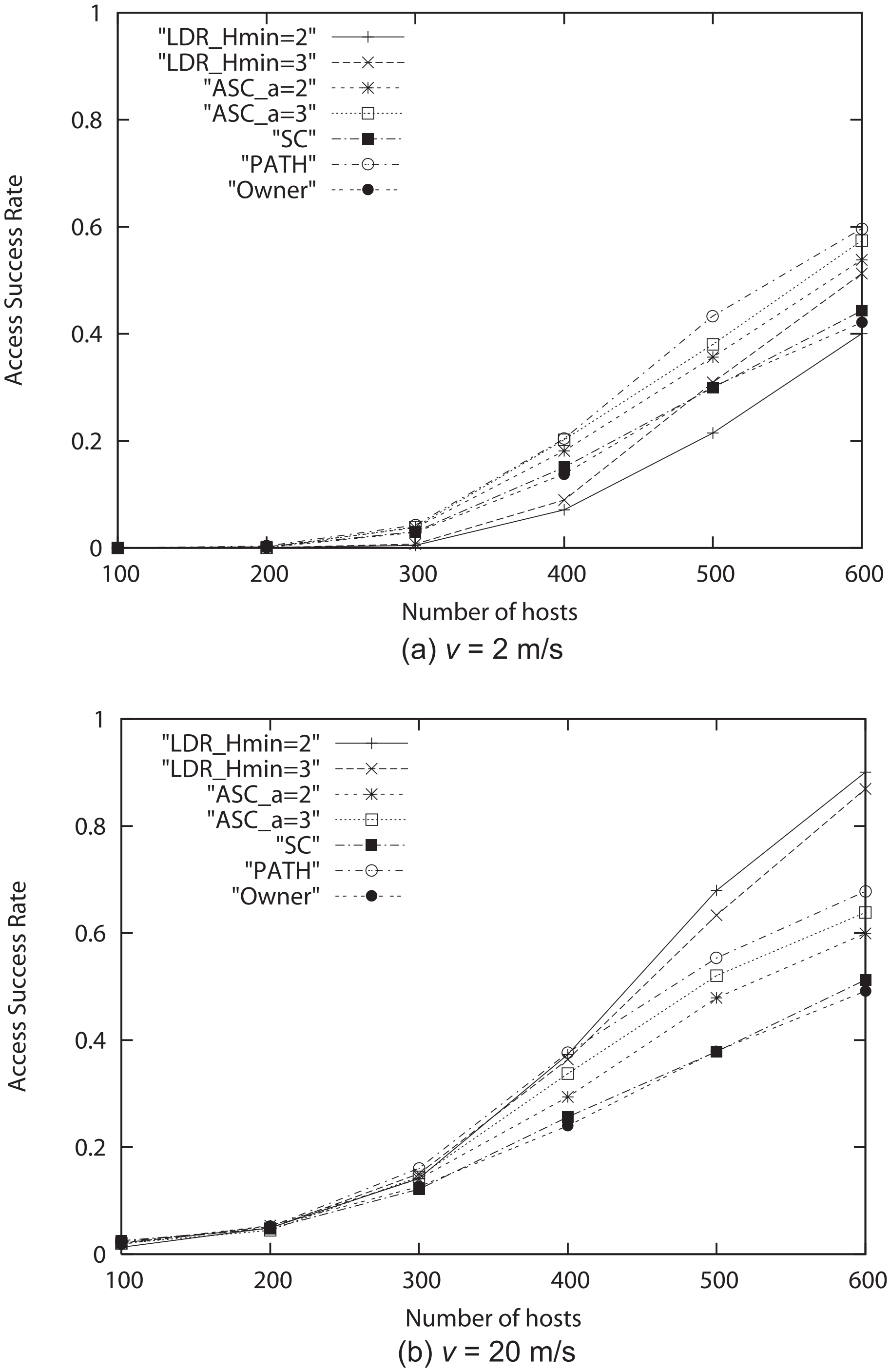


Figure 13 Access success rate vs. number of hosts (priority–on–distant model)

6 Conclusion

We have developed and evaluated a method for arranging replicas of location-dependent data to maintain high availability in mobile ad hoc networks. In our “link-aware and density-based replica arrangement (LDR)” method, replicas are placed on mobile hosts neighbouring the branch points on the reply route, and the density of replicas is controlled in accordance with the degree of the network topology. Simulation results showed that the LDR method has a higher access success rate than conventional methods when mobile hosts move quickly and the requested information is related to a distant location.

We evaluated only the effect of replica arrangement on data access. Our LDR method can also be used for replica distribution when a data item is generated. We plan to evaluate the effects of using it for both replica distribution when a data item is generated and replica arrangement on data access.

Acknowledgements

This work was supported by MEXT Grant-in-Aid for Young Scientists (A) 18680008 and Grant-in-Aid for JSPS Fellows 191313.

References

Broch, J., Maltz, D. A., Johonson, D. B., Hu, Y. C., and Jetcheva, J. (1998). A Performance Comparison of Multi-Hop Wireless Ad Hoc Network Routing Protocols. In *ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom ’98)*, pages 85–97.

Clausen, T. and Jacquet, P. (2003). Optimized link state routing protocol (olsr). RFC 3626, http://www.ietf.org/rfc/rfc3626.txt.

Cohen, E. and Shenker, S. (2002). Replication strategies in unstructured peer-to-peer networks. In *ACM SIGCOMM’02*, pages 177–190.

Hara, T. (2001). Effective Replica Allocation in Ad Hoc Networks for Improving Data Accessibility. In *IEEE INFOCOM 2001*, volume 3, pages 1568–1576.

Hara, T. (2002). Replica Allocation in Ad Hoc Networks with Periodic Data Update. In *3rd International Conference on Mobile Data Management*, pages 79–86.

Hara, T., Loh, Y. H., and Nishio, S. (2003). Data Replication Methods Based on the Stability of Radio Links in Ad Hoc Networks. In *International Workshop on Mobility in Databases and Distributed Systems*, pages 969–973.

JiST/SWANS (2004). JiST – Java in Simulation Time / SWANS – Scalable Wireless Ad Hoc Network Simulator. http://jist.ece.cornell.edu/index.html.

Johnson, D., Hu, Y., and Maltz, D. (2007). The Dynamic Source Routing Protocol (DSR) for Mobile Ad Hoc Networks for IPv4. RFC 4728, http://www.ietf.org/rfc/rfc4728.txt.

Karp, B. and Kung, H. T. (2000). GPSR: greedy perimeter stateless routing for wireless networks. In *ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom 2000)*, pages 243–254.

Ko, Y. B. and Vaidya, N. H. (2002). Flooding-Based Geocastiong Protocols for Mobile Ad Hoc Networks. *Mobile Networks and Applications*, 7(6):471–480.

Lv, Q., Cao, P., Cohen, E., Li, K., and Shenker, S. (2002). Search and replication in unstructured peer-to-peer networks. In *International Conference on Supercomputing*, pages 84–95.

Maihofer, C., Leinmuller, T., and Schoch, E. (2005). Abiding geocast: time–stable geocast for ad hoc networks. In *Second ACM International Workshop on Vehicular Ad Hoc Networks*, pages 20–29.

Perkins, C. E., Belding-Royer, E. M., and Das, S. (2003). Ad hoc on-demand distance vector (aodv) routing. RFC 3561, http://www.ietf.org/rfc/rfc3561.txt.

Perkins, C. E. and Bhagwat, P. (1994). Highly dynamic destination-sequenced distance-vector routing (dsdv) for mobile computers. *Computer Communications Review*, pages 234–244.

Shinohara, M., Hara, T., and Nishio, S. (2007). Data replication considering power consumption in ad hoc networks. In *8th nternational Conference on Mobile Data Management*.

Tsuchida, G., Okino, T., Mizuno, T., and Ishihara, S. (2005). Evaluation of a replication method for data associated with location in mobile ad hoc networks. In *Second International Conference on Mobile Computing and Ubiquitous Networking (ICMU 2005)*, pages 116–121.

Tsuchida, G., Suzuki, N., Yamanaka, M., and Ishihara, S. (2006). Adaptive replication of location-dependent data in ad hoc networks. In *Third International Conference on Networked Sensing Systems (INSS2006)*, page 126.

VICS (1996). VICS – Vehicle Information and Communication System. http://www.vics.or.jp/.

Yashiro, T. (2006). A new paradigm of v2v communication services using nomadic agent. In *Second International Workshop on Vehicle-to-Vehicle Communications 2006 (V2VCOM 2006)*, pages 1–6.

Yin, L. and Cao, G. (2006). Supporting Cooperative Caching in Ad Hoc Networks. *IEEE Transaction on Mobile Computing*, 5(1):77–89.

Zipf, G. (1949). Human Behavior and the Principle of Least Effort. *Addison-Wesley*.