에드 혹 네트워크에서 이동성 관리를 위한 적응적 랜덤 데이터베이스 그룹 방안의 설계 및 평가

배 인 한† · 하 숙 정††

요 약

모바일 에드 혹 네트워크는 어떤 고정된 네트워크 인프라구조도 갖고 있지 않는 모바일 노드들로 구성된 망이다. 모바일 노드의 위치는 빈번히 변경되므로 노드의 위치를 효과적으로 관리하는 것은 관심을 끄는 분야이다. 본 논문에서는 모바일 에드 혹 네트워크에서 모바일 노드의 위치를 관리하기 위한 적응적 랜덤 데이터베이스 그룹 스킴을 제안한다. 제안하는 스킴은 노드의 이동성을 관리하기 위해 네트워크 노드들의 위치에 대한 정보를 위치 데이터베이스에 저장한다. 모바일 노드가 이동하거나 다른 노드의 위치를 필요로 한다면, 모바일 노드는 위치 정보를 업데이트하거나 질의하기 위해서 몇 개의 데이터베이스를 랜덤하게 선택한다. 질의를 하는 경우에 선택되는 데이터베이스의 수는 고정되어 있는 반면에, 업데이트를 하는 경우에는 업데이트를 하고자 하는 모바일 노드의 인기도에 따라 데이터베이스의 수가 결정된다. 제안하는 스킴의 성능을 분석적 모델을 사용하여 평가하였으며 기존의 랜덤 데이터베이스 그룹 스킴과 성능을 비교하였다.

Design and Evaluation of ARDG Scheme for Mobility Management in Ad Hoc Networks

Ihn-Han Bae† · Sook-Jeong Ha††

ABSTRACT

Mobile ad hoc networks (MANETs) are networks of mobile nodes that have no fixed network infrastructure. Since the mobile node's location changes frequently, it is an attractive area to maintain the node's location efficiently. In this paper, we present an adaptive randomized database group (ARDG) scheme to manage the mobile nodes' mobility in MANETs. The proposed scheme stores the network nodes' location in location databases to manage the nodes' mobility. When a mobile node changes its location or needs a node's location, the node randomly select some databases to update or query the location information. The number of the selected databases is fixed in the case of querying while the number of the databases is determined according to the node's popularity in the case of updating. We evaluated the performance of the proposed scheme using an analytical model, and compared the performance with that of the conventional randomized database group (RDG) scheme.

키워드 : 에드 혹 네트워크(Ad Hoc Network), 위치 정보(Location Information), 랜덤 데이터베이스 그룹(Random Database Group), 노드의 인기도(Node's Popularity), 그룹의 크기(Size of Group)

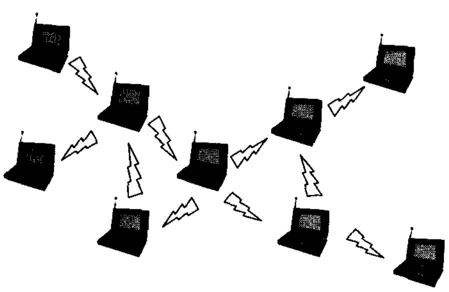
1. Introduction

Advances in wireless communications and small, light-weight and portable computing devices have made mobile computing possible. Recently, one of the research issues that have much attention concerns the design of a mobile ad hoc network (MANET), which consists of a set of mobile hosts that roam at will and communicate with one another. Since a MANET does not support base stations, communication takes place through wireless links among mobile hosts using their own antennas. This means that mobile hosts may not be able to communicate with one another directly because of transmission distance limitation. Hence, several hosts may need to relay a packet until it reaches its final destination. This situation requires each mobile host in a MANET to serve as a router [1].

(Figure 1) shows a typical MANET which works best in situations like battlefields, festival grounds, assembles, outdoor activities, rescue actions or major disaster areas, where users need to deploy networks immediately without the benefit of base stations or fixed network infrastructures. Mobility of the network nodes, limited resources (e.g., bandwidth and battery power), and potentially large number of nodes make the routing and mobility manage-

† 정 회 원 : 대구가톨릭대학교 컴퓨터정보통신공학부 교수
†† 정 회 원 : 경북대학교 전자전기컴퓨터학부 초빙교수
논문접수 : 2004년 9월 20일, 심사완료 : 2004년 12월 7일

ment in ad hoc networks extremely challenging problems [2]. Various services such as location dependent query processing, navigation, geographic messaging and neighbor and service discovery may need node location information [3]. Because a mobile node's location changes frequently, it is an attractive challenge to maintain the location information of the node efficiently.



(Figure 1) Typical MANET

Typically, the frequency of call arrivals i.e. popularity of every mobile node is different from each other. We say that a mobile node that has many request calls per time unit is hot. Since the number of call requests per time unit is greater than that of a cold node, the higher popularity of the hot node is, the more increased total query cost per time unit are. If we can reduce the query cost of the hot node even though we pay for the additional update cost, we can reduce total cost for update and query. In this paper, we propose a scheme using adaptive randomized database groups (ARDG), based on the virtual backbone architecture. The databases to construct a RDG are selected randomly, but the size of a RDG for update is determined according to the mobile node's popularity. Because a hot mobile's location is stored in more databases than that of a cold mobile, our scheme can have spatial locality and fast response time in the case of location query for the hot node.

The rest of the paper is organized as follows. Section 2 gives a brief description of related work. Section 3 contains details of the proposed ARDG scheme. Section 4 presents the performance of the ARDG that is evaluated through an analytical model and finally Section 5 concludes the paper.

2. Related Work

In order to implement the mobility management scheme, various researches have been conducted in [2, 4, 5], etc., where a set of special mobile nodes is chosen to contain the location databases. The set comprises a self-organizing virtual backbone in a MANET. Under ideal conditions, all nodes in the virtual backbone are interconnected and every non-backbone node is connected to at least one backbone node so that any two nodes can communicate with each other. In addition, all nodes in the network can carry out routing of the actual traffic. Moreover, the virtual backbone nodes can communicate with each other through routes that pass through the non-backbone nodes [4]. The virtual backbone nodes maintain interconnection among themselves by using any appropriate routing method. Addition to mobility management, the virtual backbone can also perform other network functions, such as channel assignment, flow control and system security.

In [4], Haas proposed an ad-hoc mobility management scheme that utilized a set of Uniform Quorum Systems (UQS). A set of mobile hosts, which forms a virtual backbone, is chosen to contain the location databases. These databases in the virtual backbone serve only as containers for storing and retrieving the location information of every mobile host, and are organized into quorums so that there is always at least one common database between any two quorums. When a mobile host needs to update its location or initiate a call to a destination node, a mobile's location information is written to or read from all the databases in a quorum that is chosen from the set of quorums in a non-deterministic manner. Since there is always one intersection in any two quorums, the mobile host, which wants to make a call to a destination node, can get the location information of the desired node.

In [5], Haas proposed a scheme using randomized database groups (RDG) based on the virtual backbone architecture. The number of databases chosen to update or query the location information of a node is fixed, but databases to construct a quorum are chosen randomly in contrast to [4]. That is, if a virtual backbone consists of n databases and the size of a quorum is K, any combination of K databases forms a RDG, where the location information of a mobile node is stored and retrieved. In details, if a mobile host wants to update its location, it constructs a RDG with K random databases out of n databases, and updates its location information in K databases in the RDG. If a mobile host wants to initiate a call to a destination node, it simply queries K random databases to find the location information of the destination node. If the K random databases have at least one location database that has the destination node's correct location information, we say that the query is successful.

In [2], several RDG query schemes were studied and their performance was compared. The probability of the first query being successful, the average query delay to find the mobile node's location, and the cost estimation function to implement the RDG scheme were presented.

3. ARDG Scheme

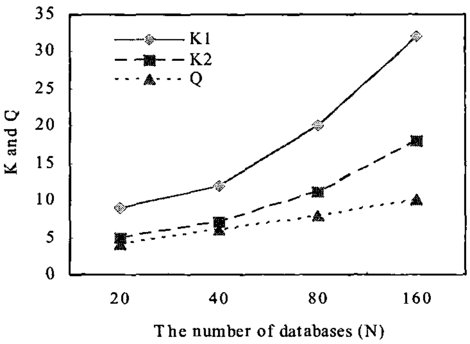
We consider a MANET that utilizes location databases to manage ad-hoc mobility as [4]. When a mobile host wants to update its location information, RDG randomly chooses some databases to construct a RDG regardless of the node's popularity. The popularity of a node represents the frequency of call arrivals and the popularity of every node is typically not uniform. A hot node means that the number of request calls for the node is greater than that of a cold node per time unit. Let us suppose that the probability that one query for a node's location fails is identical. Then, the total query rounds required to find the hot node's location are more than those of a cold node's location since the number of call request is more than that of a cold node per time unit. The higher popularity of the hot node is, the more total query rounds is per time unit. If we can reduce the number of query rounds for a hot node, the total number of query rounds for all the nodes can reduce. Since the number of updated data base is larger than that of RDG scheme, the probability that ARDG can retrieve a hot node's location increases in contrast to RDG scheme. Therefore, we propose a scheme with adaptive randomized database groups (ARDG) that determines the size of RDG for update according to a mobile node's popularity in this paper. ARDG use K_1 as the size of database groups for update if the mobile node is hot, while it uses K_2 as the size of database groups for update if it is not hot, where $K_1 > K_2$. When a mobile host needs a destination node's location information, it randomly chooses Q databases to construct a RDG and queries to the databases in the RDG like RDG scheme. If the queried Q databases have at least one common database with the updated RDG, the query is considered successful. However, if there is no common database, a group of newly randomized Q databases is constructed and queried. This process repeats until there is at least one common database between the newly queried Q databases and the updated K_1 or K_2 databases so that the mobile host can get the desired location information from the common database. This adaptive randomized database group (ARDG) scheme is based on the fact that the reduced cost caused by the rapid retrieval of hot node's location information can offset the additional cost caused by updating more databases when the ratio of call arrival to update request is high.

The operation of the ARDG scheme can be modeled as "Sampling from dichotomous population" [6] as follows. Let us consider a population that consists of N balls. The population has K white balls numbered from 1 through K, and $N-K$ black balls numbered from $K+1$ through N. Samples of Q balls are successfully drawn randomly from this population until one of the samples contains at least one white ball. In our model, the ball population represents the nodes in a MANET ; the white balls represent updated databases while the drawn balls represent queried databases.

Let X denote the number of drawings required to obtain a white ball. The probability that the first query will be successful is equal to one minus the probability that no such databases are found in the first trial [2] :

$$P(X=1) = 1 - \frac{\binom{N-K}{Q}}{\binom{N}{Q}} \tag{1}$$

A mobile node maintains the number of calls arriving from every mobile host. If the number during time interval $[t-T, t]$ exceeds the threshold value that can be determined by system, the node is considered hot. Otherwise, the node is considered cold. We use K_1 instead of K as the update group size of a hot node, where K_1 makes the hot node's $P(X=1)$ approximate to 0.9 in equation (1), so that ARDG can find the hot node's updated location among K_1 randomized databases at the first query round. We use K_2 instead of K as the update group size of a cold node, where K_2 makes the cold node's $P(X=1)$ approximate to 0.7 in equation (1) and $K_1 > K_2$. (Figure 2) shows the different values of K_1 and K_2 for different N and Q. If $N=20$ and $Q=4$, K_1 is 9 and K_2 is 5. If $N=160$ and $Q=10$, K_1 is 32 and K_2 is 18.



(Figure 2) Values of K_1 and K_2 according to $P(X=1)$ for N

To update location information in databases, ARDG scheme uses location-change update method as [4, 5]. That is, when a mobile host changes its location, it updates its locations in an RDG to new location. In many applications, the delay in obtaining the location information is critical, thus small number of queries is desirable [2]. To minimize the number of query rounds, we can use the following search strategies in [2] :

① Strategy A : The first query uses Q as group size. If the query is not successful, the second query uses $N-Q-K_2+1$ as group size, where $K=K_1$ if destination node is hot, and $K=K_2$ if destination node is not hot. If the query is not successful, the rest $(N-Q)$ databases must have K updated databases. We need only one updated database among K updated databases. Therefore, if we use $(N-Q)-K$ as the second group size, we can always obtain an updated database. Hence, the maximum delay becomes two queries.

② Strategy B : The first query uses Q as group size. If the query is not successful, the second query uses Q as group size again. If the second query is not successful, the final query uses $N-2Q-K_2+1$ as group size. Hence, the maximum delay becomes three queries.

4. Performance Evaluation

To evaluate the cost of a dynamic querying/updating process, we use the analysis method in [5]. Let Y be the number of updated databases in a sample of Q databases $(Q>1)$. Then, the probability function of Y is as follows :

$$P(Y=k)=\binom{Q}{k}\frac{K^{(k)}(N-K)^{(Q-K)}}{N^{(Q)}}=\frac{\binom{K}{k}\binom{N-K}{Q-k}}{\binom{N}{Q}} \equiv P(k,Q,K,N),\ k=0,1,\cdots,Q \quad (2)$$

From equation (2), the probability function of X, which represents the number of query rounds to obtain an updated database, can be obtained as follows :

$$\begin{aligned} P(X=1)&=1-P(0,Q,K,N)\\ P(X=2)&=P(0,Q,K,N)(1-P(0,Q,K,N-Q))\\ P(X=3)&=P(0,Q,K,N)(P(0,Q,K,N-Q))\\ &\quad(1-P(0,Q,M,N-2Q) \end{aligned} \quad (3)$$

Each cost of the querying process for the different search strategies described in Section 3 is as follows :

- earch Strategy A :

$$C_A=P(X=1)\times Q+(1-P(X=1))\times(N-Q-K+1) \quad (4)$$

- Search Strategy B :

$$\begin{aligned} C_B=\ &P(X=1)\times Q+P(X=2)\times Q\\ &+(1-(P(X=1)+P(X=2))\times(N-2Q-K+1) \end{aligned} \quad (5)$$

The cost of a dynamic process is given by the sum of the total update cost and the total query cost as follows :

$$\begin{aligned} C=\ &\lambda_u(P_{h1}K_1C_u+(1-P_{h1})K_2C_u)\\ &+\lambda_q(P_{h2}(C_{shm}C_q)+(1-P_{h2})(C_{scm}C_q)) \end{aligned} \quad (6)$$

$$\begin{aligned} C=\ &\lambda_u(P_{h1}K_1C_u+(1-P_{h1})K_2C_u)\\ &+\frac{\lambda_q}{\lambda_u}(P_{h2}(C_{shm}C_q)+(1-P_{h2})(C_{scm}C_q))) \end{aligned} \quad (7)$$

In equation (7), λ_u and λ_q represent update rate and query rate (arrivals per unit time), respectively. C_u and C_q represent the cost per update and the cost per query, respectively. P_{h1} is the probability that the mobile host, which wants to update its location, is hot, and P_{h2} is the probability that the destination node is hot. C_{shm} and C_{scm} are the costs of querying to find a hot node and a cold node through a searching strategy, respectively. C_{shm} and C_{scm} can be obtained from equation (3) and (4) by substituting K_1 and K_2 for K, respectively. Lastly, λ_q/λ_u represents the ratio of call to mobility (CMR).

$$\begin{aligned} C_{total}=\ &(P_{h1}K_1C_u+(1-P_{h1})K_2C_u)\\ &+\frac{\lambda_q}{\lambda_u}(P_{h2}(C_{shm}C_q)+(1-P_{h2})(C_{scm}C_q)) \end{aligned} \quad (8)$$

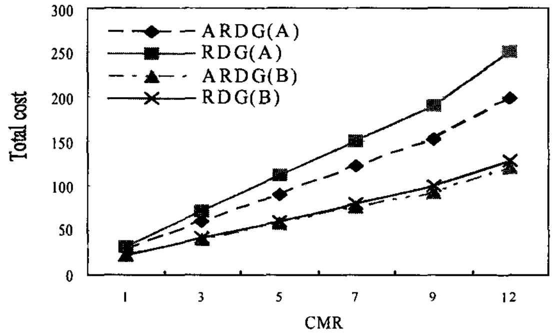
Since equation (8) can indicate the impact degree of mobile node's mobility and popularity on the total cost, we use equation (8) to evaluate the performance of our scheme.

(Figure 3) shows the results of the analytical performance evaluation using parameters in <Table 1>.

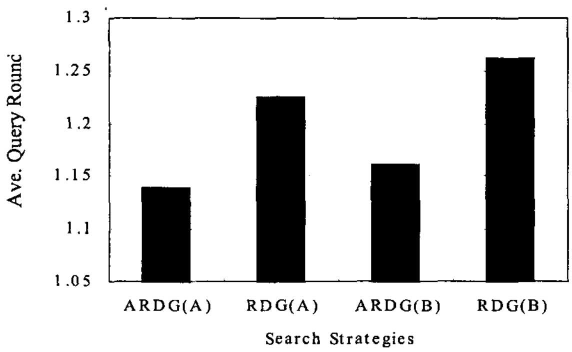
As shown in the (Figure 3), the performance of the proposed ARDG scheme is better than that of RDG scheme in any search strategy and CMR. Especially, the performance of the ARDG scheme is getting better in high CMR. It results from the spatial locality of hot mobile nodes because the location information of hot nodes is stored in more databases. The total cost of the ARDG (B) scheme which uses search strategy B is better than that of the ARDG(A) which uses search strategy A.

〈Table 1〉 Parameters for analytical evaluation

Parameters	Value
N	80
K	13
K_1	20
K_2	11
Q	8
Ph_1	0.25
Ph_2	0.75
C_u, C_q	1



(Figure 3) Result of cost comparison



(Figure 4) Results of the average number of query rounds

(Figure 4) demonstrates the average number of query rounds to manage mobility. From the (Figure 5), we can see that ARDG scheme is superior to RDG scheme in any search strategy. The total cost of the ARDG(B) is better than that of the ARDG(A), but the average number of query rounds of the ARDG(B) is little worse than that of the ARDG(A). Therefore, we can say that the performance of the ARDG(B) is better than that of the ARDG(A), but the ARDG(A) scheme is more applicable to real time applications.

5. Conclusion

RDG scheme is a simple but robust and efficient method to implement a mobility management in mobile ad hoc networks. In this paper, we have proposed an ARDG scheme that considers mobile's popularity on the basis of the RDG scheme. We have evaluated the performance of the proposed ARDG scheme using an analytical model. Results of the performance evaluation have shown that the performance of the proposed ARDG scheme is beztter than that of RDG scheme in any search strategy and the ratio of call to mobility. Therefore, the proposed ARDG scheme can be considered as a mobility management scheme that provides a destination node's location information to mobile nodes fast and efficiently in MANETs. In the near future, we will study a uniform quorum system (UQS) scheme that considers mobile's popularity to manage mobility in MANETs.

References

[1] Yu-Chee, Shih-Lin Wu, Wen-Hwa Liao, Chih-Min Chao, "Location Awareness in Ad Hoc Wireless Mobile Networks," IEEE Computer, Vol.34, No.6, pp.46-52, 2001.

[2] J. Li, Z. J. Haas and B. Liang, "Performance Analysis of Random Database Group Scheme for Mobility Management in Ad hoc Network," IEEE International Conference on Communications, pp.11-15, 2003.

[3] S. Bhattacharya, "Randomized Location Service in Mobile Ad Hoc Networks," Proceedings of the 6th international workshop on MSWIM'03, pp.66 73, 2003.

[4] Z. J. Haas and B. Liang, "Ad-Hoc Mobility Management with Uniform Quorum Systems," IEEE/ACM Transaction on Networking, Vol.7, No.2, pp.228-240, 1999.

[5] Z. J. Haas and B. Liang, "Ad-Hoc Mobility Management with Randomized Database Group," International Conference on Communication, pp.6-10, 1999.

[6] E. Parzen, "Modern Probability Theory and Its Applications," John Wiley and Sons Inc., 1992.

[7] Yuh-Jzer Joung, "Quorum-Based Algorithms for Group Mutual Exclusion," IEEE Transactions on Parallel and Distributed Systems, Vol.14, No.6, pp.463-476, 2003.

배 인 한

e-mail : ihbae@cuth.cataegu.ac.kr

1984년 경남대학교 전자계산학과(학사)

1986년 중앙대학교 대학원 전자계산학과(석사)

1990년 중앙대학교 대학원 전자계산학과(박사)

1996년~1997년 Computer and Information Science, The Ohio State University (Post-doctoral)

2002년~2003년 Department of Computer Science, Old Dominion University (Visiting Professor)

1989년~현재 대구가톨릭대학교 컴퓨터정보통신공학부 교수

관심분야 : 이동 무선망, 무선 인터넷, M-Commerce, 멀티미디어 시스템 등

하 숙 정

e-mail : sjha@m80.knu.ac.kr

1988년 계명대학교 전자계산학과(학사)

1990년 중앙대학교 전자계산학과(석사)

1998년 대구가톨릭대학교 전산통계학과 전산전공(박사)

2001년~현재 경북대학교 전자전기컴퓨터학부 초빙교수

관심분야 : 이동 무선망, 임베디드 소프트웨어, 멀티미디어 시스템 등