

## Use of Hidden Markov Mobility Model for Location Based Services

Bhakti D. Shelar\*, Mr. D. K. Chitre\*\*

\*(Department of Computer Engineering, Terna Engineering College, Neral, Mumbai University, Navi Mumbai)

\*\* (Department of Computer Engineering, Terna Engineering College, Neral, Mumbai University, Navi Mumbai )

### ABSTRACT

These days people prefer to use portable and wireless devices such as laptops, mobile phones, They are connected through satellites. As user moves from one point to other, task of updating stored information becomes difficult. Provision of Location based services to users, faces some challenges like limited bandwidth and limited client power. To optimize data accessibility and to minimize access cost, we can store frequently accessed data item in cache of client. So small size of cache is introduced in mobile devices. Data fetched from server is stored on cache. So requested data from user is provided from cache and not from remote server. Question arises that which data should be kept in the cache? Performance of cache majorly depends on the cache replacement policies which select data suitable for eviction from cache. This paper presents use of Hidden Markov Models(HMMs) for prediction of user's future location. Then data item irrelevant to this predicted location is fetched out from the cache. The proposed approach clusters location histories according to their location characteristics and also it considers each user's previous actions. This results in producing high packet delivery ratio and minimum delay.

**Keywords** – Cache replacement policy, Hidden Markov Model, Hidden state, Location Based Services, random Variables

### I. INTRODUCTION

A mobile ad hoc network is a type of wireless network. Wireless devices come together and form a temporary infrastructure-less network for communication purpose. Each device is free to move in any direction and it must pass on the data traffic unrelated to its own use. The basic aspect of constructing a MANET is each device must be efficient enough to maintain the information required to properly route traffic.

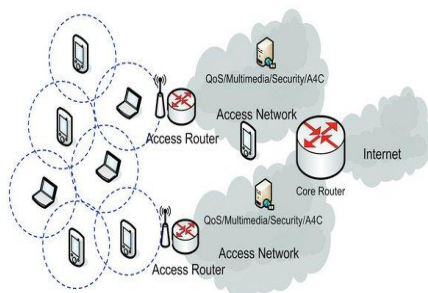


Fig. 1 Mobile Ad hoc Network(MANET)

As shown in Fig. 1[1], such networks are capable to be operated by themselves by using else they are inter connected to the Internet through Access Router. Security of data traffic is tried to achieve by using cryptography. Wireless devices may contain one or multiple and different transceivers between nodes.

Location based service (LBS) is a programmed application for mobile client which requires information related to its current location. This service is query-based and provides useful information to the user such as “nearest Gas Station ” or “nearest Restaurant”. Also LBS can be push-based which periodically provides location relevant and useful information to the user.

As shown in Fig. 2[2], mobile devices form cellular network connected with satellites. One location server is maintained to store data local to that cellular network, which is provided to the client. The frequently accessed data is then placed in client's cache. An LBS requires five basic components: end user's mobile device, a content provider to supply the end user with geo-specific information, the service provider's software application, a mobile network to transmit data and requests for service, , a positioning component like Global Positioning System(GPS).



Fig. 2 Location Based Service System

Caching in MANET has some constraints like limited cache space, limited power supply and high mobility. The availability of the data in local cache minimizes access cost and transmission delay. An efficient cache replacement mechanism is needed to differentiate the items to be kept in cache and that is to be evicted when the cache is full. Rather than selecting random data item from cache, system performance will be better if we select data item that is not frequently asked or not relevant to the user's current location. So a good replacement policy is essential to achieve high hit rates.

The rest of the paper is organized as follows. Section II briefly describes survey of the different cache replacement policies. Section III details the Hidden Markov Mobility model. Section IV explains Least Recently Used and section V explains Furthest Away Replacement policies implemented in NS2. Section VI and section VII deal with Experimental scenario and Result Analysis. Section VIII concludes the paper.

## II. LITERATURE SURVEY

Existing cache replacement policies determine the cost functions of data item. These functions may be access frequency, update rate, size of item etc. Least Recently Used (LRU), Least Frequently Used (LFU) and LRU-k [13] are the temporal based replacement policies. These algorithms replace data items which are accessed recently or frequently. These policies are based on the assumption that Mobile user's access patterns exhibit temporal locality. It means that data item objects that were queried frequently in the past will be queried in future). However, in mobile environments where clients take service from location based service system. Clients access pattern don't exhibit only temporal locality, but also exhibit dependence on location of data, location of the client and direction of the client's movement [4][5]. Due to this, these described policies are unsuitable for supporting location dependent services because they do not consider the movement of mobile clients and location of data items. Hence, depending only on temporal locality when making cache replacement decisions will result in poor cache hit ratio in LBS.

To Solve these difficulties, some location-aware cache replacement strategies[5][7][9][10] have been proposed for location based services. Manhattan Distance-based cache replacement policy [10] supports location dependent queries in urban environments. Cache replacement decisions are made on the basis of distance between a client's current location and the location of each cached data object. Objects with the highest Manhattan distance from the client's current location are evicted at cache replacement. Furthest Away Replacement (FAR) [9] policy uses the current location and movement

direction of mobile clients to make cache replacement decisions. Cached objects are grouped into two sets, viz., in-direction set and the outdirection set. Data objects in the out-direction set are always evicted first before those in the in-direction set.

In Probability Area Inverse Distance (PAID) [5] policy, the cost function of data item  $i$  takes into account the access probabilities ( $P_i$ ) of data objects, area of its valid scopes  $A(vsi)$  and the distance  $D(vsi)$  between the client's current position and the valid scope of the object concerned (known as data distance None of these cache replacement policies are suitable if client changes its direction of movement quite often. Existing cache replacement policies only consider the data distance but not the distance based on the predicted region or area where the mobile user may be in future. Very few of these policies [5][7] account for the location and movement of mobile clients.

In this paper, we study the problem of analysing human location histories to predict the next places to be visited This we have done by using an approach based on Hidden Markov Models. Also we have evaluated its limitations on real-world data. This paper proposes an approach for predicting the future locations of mobile users and then keep location-related data in the cache and evict unrelated data from the cache. These decisions are made on the basis of user's previous visits to other locations, and leveraging on Hidden Markov Models for capturing the patterns embedded in previously collected location histories. This policy considers most probable future location of the user and then takes the replacement decisions.

## III. PROPOSED METHOD

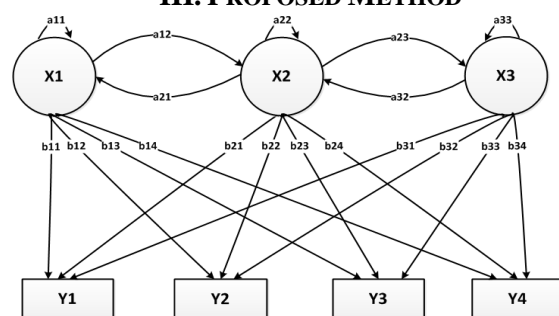


Fig. 3 Hidden Markov Model

The diagram in Figure 3 shows the general architecture of an instantiated HMM. Each shape in the diagram represents a random variable that can adopt any of a number of values. The random variable  $x(t)$  is the hidden state at time  $t$ .  $x(t) \in \{x_1, x_2, x_3\}$  random variable  $y(t)$  is the location visited at time  $t$  (with  $y(t) \in \{y_1, y_2, y_3, y_4\}$ ). The arrows in the diagram denote conditional dependencies. conditional probability distribution of the hidden

variable  $x(t)$  at time  $t$ , depends only on the value of the hidden variable  $x(t-1)$ , and thus the values at time  $t-2$  and before have no influence. This is called the Markov property. The value of the observed location  $y(t)$  only depends on the value of the hidden variable  $x(t)$ , at time  $t$ . The probability of next location is calculated by using given formula :

$$P(Y) = \sum_X P(Y|X) P(X)$$

Where,  $X$  is hidden variable, i.e. previous location of user and  $X = \langle x(1), x(2), x(3), \dots \rangle$ .  $Y$  is current location visited at time  $t$  of user and  $Y = \langle y(1), y(2), y(3), \dots \rangle$ . Overcomes the limitation of Random models to capture temporal dependency. Mobility of a node constrained by the physical laws of acceleration, velocity and rate of change of direction. the velocities of single node at different time slots are 'correlated'. Very much useful to capture real world scenarios.

#### IV. LEAST RECENTLY USED(LRU) CACHE REPLACEMENT POLICY

The least-recently-used cache replacement policy is one that evicts items that are accessed least frequently once its threshold has been exceeded. In simple terms, the LRU cache will remove the element in the cache that has not been accessed in the longest time. LRU replaces data item based on the recency of its access. It considers temporal locality of client's access pattern. Also it neglects location and movement direction of mobile client.

LRU Algorithm:-

1. Declare a reference count.
2. When data is accessed reference count incremented by 1.
3. Consider timestamp of cached data.
4. Determine a threshold value.
5. If (timestamp(data) < threshold) Eject from the cache else Retain in the cache.

#### V. FURTHEST AWAY REPLACEMENT (FAR) CACHE REPLACEMENT POLICY

FAR uses current location and movement direction of the client to make replacement decisions. Also it neglects the temporal locality of cached data item.

FAR Algorithm:-

1. Consider the  $(x,y)$  coordinates of client's location.
2. Calculate distance between data item and client
3. Determine a threshold value.
4. If (distance  $d_i >$  threshold) Put data item in the out\_direction set. Else Put data item in the in\_direction set.
5. Always evict the data items in out\_direction set first.

Later evict the data items in in\_direction set if required.

#### VI. EXPERIMENTAL SCENARIO

We used Network Simulator version 2 (NS2) simulator for implementation of topologies and three (LRU, FAR AND HMM based ) cache replacement policies. NS2 simulator provides more flexible coding support to design topologies. Also, it generates trace file to analyse the performance of algorithms implemented. The simulation runs for 10 milliseconds and uses 50 mobile nodes. Geographical region boundary is defined within 500 x 500 area. When any node reaches the boundary, it is repelled back towards the centre. Flatgrid network topology is used. For transmission purpose, omni-directional antenna is used. Omni directional antenna is best suitable for moving nodes. For comparison of three cache replacement policies we have considered following three performance metrics:

- a) Packet Delivery Ratio:- The ratio of packets that are successfully delivered to a destination compared to the number of packets that have been sent out by the sender.
- b) Link Throughput :- is the rate of successful message delivery over a communication channel.
- c) Transmission Delay:- end to end delay is the duration since a packet enters a queue of a source node until it arrives at the packet node.

#### VII. RESULTS AND ANALYSIS

Following graphical analysis is done based on the packet delivery ratio, throughput and delay calculated. The values for these parameters are calculated from the NS2 trace files.

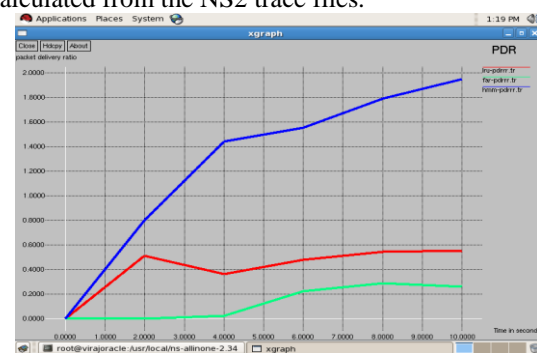


Fig. 4 Packet Delivery Ratio of HMM, LRU, FAR

As shown in Fig. 4, x-axis indicates time parameter in seconds and y-axis indicates packet delivery ratio. Blue line indicates packet delivery ratio generated by HMM based cache replacement, red line indicates indicates packet delivery ratio generated by LRU based cache replacement and green line indicates packet delivery ratio generated by FAR based cache replacement. So graph shows that proposed approach of HMM gives highest cache hit ratio for packet transmission.

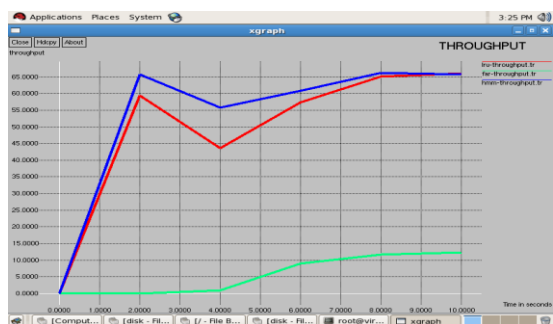


Fig. 5 Throughput of HMM, LRU, FAR

In Fig. 5 graph describes the link throughput given by HMM based, LRU based and FAR based cache replacement. Time in seconds is plotted against x-axis and throughput in number of packets is plotted against y-axis. Graphical analysis indicates that HMM based approach is caliper to generate highest throughput than LRU and FAR approach, which is really expected from real world applications.

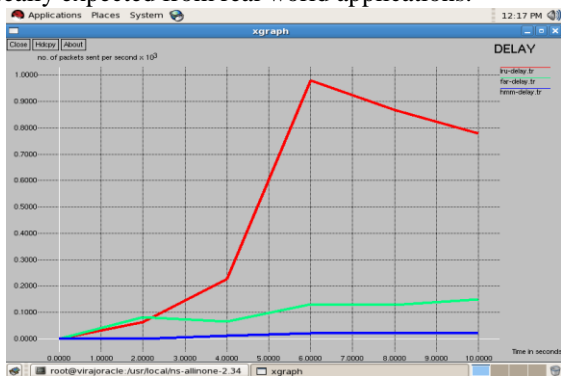


Fig. 6 Transmission Delay of HMM, LRU, FAR

As shown in Fig. 6 HMM based cache replacement overcomes the unwanted delay just because of the more accurate prediction of the future location. It produces the lowest delay while transmission.

Table 1 summarizes the performance evaluation of all three cache replacement policies. Delay parameter is inversely proportional to the number of packets transmitted. In FAR packet delivery ratio increases as compared to that of LRU. FAR policy gives better cache hit ratio than LRU.

TABLE 1. Comparison of LRU, FAR and HMM Based cache replacement policies

	LRU	FAR	HMM	Preferable
Pdr	High	Low	Highest	Hmm
Through Put	High	Low	Highest	Hmm
Delay	Medium	High	Low	Hmm

FAR packet delivery ratio increases as compared to that of LRU. Also when we follow FAR cache replacement policy, then throughput also increases as compared to the LRU. FAR policy gives better cache hit ratio than LRU. HMM based policy gives best results in all the three aspects. Its prediction accuracy gives benefit of better cache performance.

### VIII. CONCLUSION

HMM based CRP considers current location of moving client. Also it considers moving direction of user and traces location histories of client. Then it predicts probable future location. HMM based CRP keeps only that data in cache, which is relevant to that predicted location. So, HMM based CRP proves most accurate replacement policy compared to the LRU and FAR. LRU considers only no. of times data item is accessed. This information is partial and abstractive in order to make replacement decisions. FAR considers current location and movement direction of client to take replacement decisions. FAR neglects temporal properties of data item leading to less accuracy.

### IX. ACKNOWLEDGEMENTS

The authors would like to express their gratitude to Terna Engineering College, Nerul, Navi Mumbai. for technical support offered required for experimental evaluation. Also we express special thanks to the “Network Simulator version 2” Training and Workshop session held at Sinhgad Institute Of Technology Lonavala, Pune.

### REFERENCES

- [1] <http://www.personal.psu.edu/sxr360/blogs/mobileadhocnetworkpic>
- [2] <http://www.watblog.com/2008/04/16/mobile-positioning-and-location-based-services-an-insight-into-the-future-of-mobile/>
- [3] Ajey Kumar, Manoj Misra, Anil K. Sarje, “A Predicted Region based Cache Replacement Policy for Location Dependent Data in Mobile Environment”, I. J. Communications, Network and System Sciences. 2008; 1: 1-103.
- [4] D. Barbara, “Mobile Computing and Databases: A Survey,” IEEE Transactions on Knowledge and Data Engineering, Vol. 11, No. 1, pp. 108-117, January/February 1999.
- [5] D. Barbara and T. Imielinski, “Sleepers and Workaholics: Caching Strategies in Mobile Environments,” In the Proceedings of the ACM SIGMOD Conference on Management of Data, Minneapolis, USA, pp. 1-12, 1994.
- [6] E. O’Neil and P. O’Neil, “The LRU-k page replacement algorithm for database disk

- buffering”, In the Proceedings of the ACM SIGMOD, Vol. 22, No. 2, pp. 296-306, 1993.
- [7] Q. Ren and M. H. Dhunham, “Using Semantic Caching to Manage Location Dependent Data in Mobile Computing,” In the Proceedings of 6<sup>th</sup> ACM/IEEE Mobile Computing and Networking (MobiCom), Boston, USA, pp. 210-221, 2000.
- [8] B. Zheng, J. Xu and D. L. Lee, “Cache Invalidation and Replacement Strategies for Location-Dependent Data in Mobile Environments,” IEEE Transactions on computers, Vol. 51, No. 10, pp. 1141-1153, October 2002.
- [9] K. Lai, Z. Tari and P. Bertok, “Location-Aware Cache Replacement for Mobile Environments,” IEEE Global Telecommunication Conference (GLOBECOM 04), Vol. 6, pp. 3441-3447, 29<sup>th</sup> November- 3<sup>rd</sup> December 2004.
- [10] S. Dar, M. J. Franklin, B. T. Jonsson, D. Srivastava and M. Tan, “Semantic Data Caching and Replacement,” In the Proceedings of the 22<sup>nd</sup> International Conference on Very Large Databases (VLDB), pp. 330-341, 1996.
- [11] Balamash and M. Krunz, “An Overview of Web Caching Replacement Algorithms,” IEEE Communications Surveys & Tutorials, Vol. 6,
- [12] D. L. Lee, W.-C. Lee, J. Xu and B. Zheng, “Data Management in Location-Dependent Information Services,” IEEE Pervasive Computing, Vol. 1, No. 3, pp. 65-72, July 2002.
- [13] J. Jing, A. Helal and A. Elmagarmid, “Client-Server Computing in Mobile Environments,” In the Proceedings of the ACM Computing Surveys, Vol. 31, No. 2, pp. 117-157, June 1999.
- [14] T. Camp, J. Boleng and V. Davies, “A Survey of Mobility Model for Ad Hoc Network Research,” Wireless Communication & Mobile Computing (WCMC): Special Issue on Mobile AdHoc Networking: Research, Trends and Applications, Vol. 2, No. 5, pp. 483-502, 2002.
- [15] A. Balamash and M. Krunz, “An Overview of Web Caching Replacement Algorithms,” IEEE Communications Surveys & Tutorials, Vol. 6, No. 2, pp. 44-56, 2004.
- [16] I. A. Getting, “The Global Positioning System,” IEEE Spectrum, Vol. 30, No. 12, pp. 36-47, December 1993.