Location Management by Movement Prediction Using Mobility Patterns and Regional Route Maps

R. K. Ghosh[1], Shravan K. Rayanchu[2], and Hrushikesha Mohanty[3]

[1] Department of CSE, IIT-Kanpur, Kanpur 208016, India
rkg@cse.iitk.ac.in
[2] Department of CSE, IIT Guwahati, Guwahati 781039, India
rayanchu@iitg.ernet.in
[3] Department of CIS, University of Hyderabad, Hyderabad 500046, India
hmcs@uohyd.ernet.in

Abstract. In this paper we argue that in most of the cases the movement pattern of a mobile host repeats itself on a day-to-day basis, but for the occasional transient deviations. Taking the spatio-temporal properties of a mobile host into account, we propose a new location management scheme. The scheme achieves the near optimal routing as it bypasses the default reliance on the routes through the home agent for most of the calls made to a mobile host. It uses the mobility pattern of the mobile host to predict the cell location of that host. Transient deviations ranging from 5-30% are tackled by tracking down a host efficiently with the help of a regional route map which is the physical route map of a small neighbourhood of the last known location of that host. The performance of the proposed scheme is evaluated with respect to varying values of call-to-mobility ratio (CMR), and found to be quite good even for transient deviations ranging upto 30%.

1 Introduction

In this paper we propose a new location management scheme by taking the spatio-temporal localities of mobile hosts into account. The strategy works on the idea that the movement patterns of most mobile hosts get repeated on day-to-day basis, but for occasional transient deviations. Thus, by modeling the mobility pattern of a mobile host at home agent and replicating the same at some chosen network sites it would be possible to predict the likely location of that mobile host. The transient deviations, if any, can be tackled by localizing the search to the neighbourhood of the last known location of a mobile host. We use the physical route map of the neighbourhood region of the last known location of a mobile host to capture its spatial locality assuming that the mobility of the host almost surely be guided by the region route map. An update is not required as long as a host moves within the area under a Mobile Switching Center (MSC) and conforms to its mobility pattern. When a mobile host deviates, it is tracked down at its new location using the regional route map. An update for

S. R. Das, S. K. Das (Eds.): IWDC 2003, LNCS 2918, pp. 153–162, 2003.

new (deviated) position is made at the MSC where it is found. The proposed scheme have following advantages: (i) it takes into account the call stability as the patron hosts [3] are identified and the mobility pattern is stored at the patron stations to achieve optimal routing; (ii) it takes into account the call locality, i.e., the calls arising from the stations that are under the same MSC as the cell in which the mobile host currently resides find the pattern stored at that MSC and are routed directly bypassing the lengthy route through the home agent; (iii) it improves over the one in [4], where the patron service is invoked for every global move, by invoking the patron service only when the host undergoes global deviations. (iv) the paging costs are optimal and update cost is zero, when the host conforms to the mobility pattern; (v) it is better than [2] which switches to conventional schemes even for a transient deviation.

The rest of the paper is organized into 7 sections. The focus of section 2 is on the attempts to capture spatio-temporal localities of mobile hosts for efficient location management. Section 3 deals with the models of mobility patterns. An overview of the proposed location management scheme is given in section 4. Section 5 provides formal algorithmic specification of the scheme along with justification and analysis. The results of simulation appear in section 6; section 7 concludes the paper.

2 Location Tracking Using Spatio-Temporal Locality

The performance of the location management schemes depends heavily on the subscriber mobility patterns. To analyze the performance of certain selected schemes, two mobility models, namely, activity-based mobility model and random mobility model were used in [6]. Very few schemes take into account the spatio-temporal properties of the host movement pattern [6]. All such schemes use static host profiles to approximately trace the location of the host at any given time. When a call is made, initially the callee's profile is used to find the expected locality. But if the callee is not found at that locality, they use the triangular routing scheme to track the callee. This approach works only for hosts with strictly unchanging schedules. Moreover, many schemes [1, 2] assume particular topologies of the cells — e.g., circular, square, or hexagonal — in order to simplify analysis. In [5], a multilayer neural network was used to model the host mobility pattern. It was observed that the average accuracy of prediction for uniform patterns was 93% and 40-70% for regular patterns. The scheme proposed in [2] also uses a neural network model to capture the mobility pattern of the host. It attempts to reduce the paging and the update costs by using this pattern to predict the location of the mobile host. So, though move stability exhibited by the host is accounted for, the schemes fail to take into consideration the factors like the frequency and the source of calls, and the call stability in designing and evaluating location management strategies. Thus the inefficiency due to triangle routing follows. In summary, the techniques used so far tend to decrease the cost of either the move or the call in the expense of other. Moreover, the scheme switches to conventional schemes (zone based or movement based) too often, even when the deviations are temporary.

3 The Mobility Model

The mobility patterns of majority of hosts can be modeled using the three models, namely, (i) the Traveling Salesman (TS) model, (ii) the Boring Professor (BP) model, (iii) the Pop-Up (PU) model.

The stations where there is a high probability of finding the host at certain predetermined times as Boring Professor (BP) stations. The host moves at approximately same times everyday between two BP stations. All the stations on the path between two BP stations are referred to as trajectory (TY) stations. Typically, a host makes a transient deviation to the locations that are close to its one of its BP stations; and such deviations are occasional. For example, consider the itinerary of a mobile user shown in figure 1. The home of the user is at station A, he works at a location which is denoted by station B; and the club he visits daily is at a station C. Everyday, the user starts from home at 9:00 AM, reaches his office around 9:30 AM, stays in the office till 5:00 PM and then he visits a club at 5:30 PM, He returns home by 8:00 PM starting from the club at 7:30 PM. The mobile host's user spends a significant amount of time at home, office and the club. Thus, these stations are considered as the BP stations. As the user travels from home to office, office to club, club to home, he mostly moves along the same route. All the stations in this route are the TY stations. Sometimes the user visits a shopping center at station X and barber at station Y. These stations are the PU stations for the mobile host. The PU stations account for the occasional transient deviations of the host from its mobility model. In figure 1, a transient deviation to a hospital on the way from home to the office is depicted.

For a majority of the mobile hosts for most of the time, the movement profile repeats itself from day-to-day basis. The mobility pattern is more or less stable but for the occasional transient changes and rarely permanent changes in the pattern. Thus, if we can tackle these transient deviations in an efficient way, then we could use the mobility pattern of the hosts to achieve minimal costs for the paging as well as for the updates.

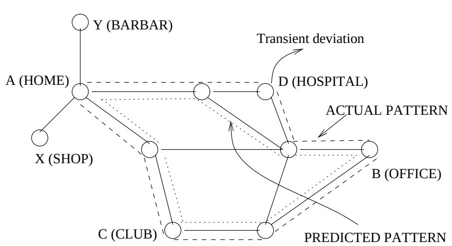

Fig. 1. Graph mobility model

4 The Proposed Scheme

In our scheme, a cellular network is divided into a few big location areas (BLAs). Each BLA, consisting of about 100 cells, is under a higher level MSC. The MSC has the access to the location information databases in the network, which are used to store the location and service information for each registered mobile host of PCS network. We enhance the functionality of the MSC by introducing special mobility functions; and MSC has a unique regional route map (RRM) associated with it.

An RRM is a bounded degree graph connected graph $G = (V, E)$ that represents the physical road connectivity between the cells under an MSC. Each cell is represented by a node. An edge between nodes A and B indicate a direct route between the cells represented by these nodes. Note that A and B must be adjacent cell if an edge exists between them. The weight $W(A, B) = t_{min}(A, B)$ denotes the minimum time required to move from cell A to cell B. The MSC uses this regional route map in order to track down the hosts which have undergone transient deviations.

The mobility pattern of a host can be obtained by monitoring the host's movements for a period of time or the information can be gathered from the host's user in advance. This pattern is replicated at the home station and other MSCs (if any) which come under the span of the mobility pattern. The concept of patron host, introduced in [3] is used here. For each mobile host, the set of source hosts from where the majority of traffic for the host has originated are called patron hosts. We call the MSCs under which one or more patron hosts reside as patron stations. It is not necessary to store the entire mobility pattern at the patron stations, it is only sufficient to know under which MSC the host resides at a particular point of time. Since the number of patron stations is very less in number compared to the number of patron hosts; it would not result in much storage overhead.

When a call request arrives at the patron station, the information from the callee's mobility pattern stored at the patron station is used and the request reaches the MSC under which the callee is present. From the complete mobility pattern, which is stored at this MSC, it finds the cells in which the callee is expected to be present. In the majority of the cases, the callee is successfully tracked down. But if the callee is not found, then the MSC use locally stored route map to track down the callee. When a call request arrives at the home station or other MSC at which the pattern is stored, the same approach is used to track the callee. This optimization is also possible for non-patron hosts residing at the patron stations or at the home station. But if a call originates from a host, outside these stations, then the request is first directed to the home station and then the same process follows. Thus, triangle routing is avoided for most of the calls.

An update is not required when the host moves locally (within an MSC) if it conforms to the mobility pattern. When a call is made to a host, if the MSC finds that the host does not conform to the pattern (local transient deviation), it uses the route map to track down the host. It also updates the new deviated position only at the MSC under which the host is present. For the subsequent calls, when the request arrives at the MSC, it finds out that the callee has deviated and thus a pages the new location. If the callee is still not found then, the hunt may begin from the deviated location or the location at which the callee was supposed to be found at that time. If the host moves into a cell that comes under a different MSC, it is checked whether the host conforms to the mobility pattern. If so, then no update takes place. Now, if there is a call for this host, then the request would indeed arrive at the new MSC as the pattern indicates it and the callee can be tracked down. But if it is found that the host moves globally and does not conform to its pattern i.e., it moves to a BLA in which the pattern is not stored, then the patron service is invoked. The home station and the patron stations are informed about this deviation. Hence, for the subsequent calls which arrive at the patron station (or the home station), it is understood that there is a global deviation. If the new BLA is a neighbour of the home station then the route map of the home station can be copied there so as to carry out the hunt efficiently as in the earlier case. But, if the new MSC is not a neighbour of the home station, then the route map is not copied. This is because, the deviation is no longer transient and the host has deviated a lot from his mobility pattern. Effectively, each MSC also has the route maps of its neighbouring BLAs to assist them in case of global deviations of hosts from their mobility patterns. If the host deviates to BLAs which are not the neighbours of home station, the hunt can still be carried out; but the host mobility pattern will not be used in assisting the MSC. Whenever there is a call, the callee can be tracked down with the help of RRM of the new BLA. The update would made periodically so as to bound the region of paging. However, it is expected that the host would resume its pattern at the earliest, so that the previous scheme can be used again.

The important idea behind the scheme is to diminish the effect of deviations in the host mobility in order to hide the local deviations completely from the rest of the network and to confine the effects of global deviations to those hosts which are more likely to call again, thereby reducing the updates and at the same time achieving optimal paging costs for the majority of cases.

5 The Hunt Algorithm

Let $S(t) = \{C_1, C_2, \ldots, C_k\}$ be the set of cells, in which the callee is most likely to be found at that time t. This set is obtained from the mobility pattern of callee stored in the MSC in which he currently resides. On an average, this set would consist of 1-3 cells. Suppose that the callee gets a call at time t_i. There are two cases depending on whether the callee has deviated or not by a variable $flag\_dev$ stored at the MSC. If a deviation has been recorded for the callee at a time $t_{i-1} \leq t_i$, then the cell in which the callee was found is also stored.

Case 1. If the callee has not deviated at some time t_{i-1} before (i.e., $flag\_dev = 0$), then the set of cells $D = S(t_i)$ in which the callee is most likely to be found are paged. This set is obtained from the mobility pattern of the callee stored in the MSC in which the callee currently resides.

Case 2. If the callee has already deviated (i.e., $flag\_dev = 1$), then the cell *prev_cell* in which the callee was previously found is paged. In this case, $D = prev\_cell$.

In both the cases if the callee is found, the location of the user is relayed to the caller. If the callee is not found then the hunt begins.

We define a reachable cell R of B, as a cell which can be reached from cell B within a time T. For this cell R to be reachable from B within time T it is necessary that $t_{min}(B, R) \leq T$. We also define remaining time t_r, as the maximum time for which the user can still move after reaching R. Hence, $t_r = T - t_{min}(B, R)$. Note that a cell is not reachable if $t_r < 0$. Extending the definition further, a reachable cell R from a set of cells X, is a cell that is reachable from any of the cells of X within time T and the remaining time $t_r = \max(T - t_{min_{R \in X}}(C, R)) = T - \max_{C \in X}(t_{min}(C, R))$, where C is any cell of X from which R is reachable within T.

Now the callee can only be in the cells reachable from D within the time $T = t_i - t_{i-1}$. Let Q be the set of ordered pairs of ⟨`reachable cells`, `remaining time`⟩. Initially $Q = \{(x, T) \mid x \in D\}$. Since this is a transient deviation, there is a strong possibility that the callee would resume its movement pattern at the earliest. Taking this into consideration a cost for all cells is calculated and then the *best* cell is obtained by finding the cell whose cost is minimum among all the cells of Q. The best cell is then paged. If the callee is found then, the new position of the callee is updated and the time t_i is also noted. A variable $flag\_dev$, which represents whether the callee has deviated, is updated or reset. In the case 1, if it is noted that the callee has deviated from his mobility pattern then $flag\_dev$ set to 1; and in case 2, if the callee is found to resume his movement pattern, then the information is updated by reseting $flag\_dev$ to 0. On the other hand if the callee is not found then, the *best* cell is removed from Q and its neighbouring cells which are 'reachable' from Q but are not yet paged are added to the set Q. Again the *best* cell among the cells of Q is found and the algorithm then continues till the callee is found. Note that the callee will definitely be found before all the elements of Q are exhausted. The specification of the hunt algorithm, as discussed above, appears below.

Algorithm 1: Hunt Algorithm

// Executed by MSC under which callee resides or by its home station
if ($flag\_dev = 0$) **then**
    $T = bound$;
    Page all cells in $D = S(t_i)$;
    // Let C_i be the reply of paging from BSs of the paged cells.
    // $C_i =$ `null` implies callee not available in any of the paged cells.
    if ($C_i \in S(t_i)$) **then** return C_i **endif**
else
    $T = t_i - t_{prev\_call}$;
    if (T is very large) **then**
        // The callee may have returned to pattern

08 $T = bound$;
09 Page all cells in $D = S(t_i)$;
10 **else**
11 Page $D = prev\_cell$;
12 **if** $(C_i \in D)$ **then** return C_i **endif**
13 **endif**
14 **endif**
15 $Q = \{(x, T) \mid x \in D\}$;
16 **repeat**
// For estimating the cost of the cells to be paged a separate
// function (see Algorithm 2) was used. Efficiency of search will
// improve with an improved cost estimation function.
17 $best\_cell = \min_{(x,T_x)\in Q}\{Cost(x)\}$;
18 Page *best_cell*;
19 **if** $(C_i = best\_cell)$ **then**
20 **if** $(C_i \in S(t_i))$ **then** $flag\_dev = 0$ **else** $flag\_dev = 1$ **endif**
21 $prev\_cell = C_i$; // Update
22 $t_{prev\_call} = t_i$;
23 return C_i;
24 **else**
25 $R = \{\ (x, T_x) \mid x$ is a *reachable* neighbour cell of *best_cell* not *paged* yet and T_x is the *remaining time*$\}$;
26 $Q = Q \bigcup R$;
27 Remove *best_cell* from Q; // Remove already paged cell.
28 **endif**
29 **until** $(Q = \phi)$

The estimation of cost for paging cell during search at line 17 of hunt algorithm has not been specified so far. We used the following algorithm for estimation of cost for paging the cell.

Algorithm 2: Estimation of Cost

$P = \{j \mid j \in S(t)$, where $t_i - \delta < t < t_i + \delta\}$;
$\eta_1 = |S(t_i)|/(|P| + |S(t_i)|)$;
$\eta_2 = |P|/(|P| + |S(t_i)|)$;
// n is total number of cells under the MSC.
for $(j = 1$ to $n)$ **do**
$W_p(j) = \min_{\forall x\in P}\{t_{min}(j, x)\}$;
$W_s(j) = \min_{\forall x\in S(t_i)}\{t_{min}(j, x)\}$
$Cost(j) = \eta_1 W_p(j) + \eta_2 W_s(j)$;
endfor

5.1 Justification and Analysis

The set of cells P in which the callee is expected to be found between the time $t_i - \delta$ and $t_i + \delta$ is determined from the mobility pattern. The parameter

δ specified either by the callee or a default value is used. As the deviations are transient in nature, the callee must have deviated some time ago; and would resume his movement pattern some time later. The parameter δ is an estimate of this interval. Since the P is the set of cells which the callee may visit in the interval $[t_i - \delta,\ t_i + \delta]$, it is likely that the callee would be: (i) either in one of these cells but his arrival at the predicted cell is delayed due to his busy schedule, traffic, etc.; (ii) or the callee has actually deviated from one of these cells to a nearby PU station and would be returning back to one of these cells again. Hence, more the closer a cell is to P, more is the possibility that the callee be found in that cell. Thus, the cost of each cell in the MSC is found on the basis of the sets P and $S(t_i)$. The cells are then paged in the increasing order of their cost. For a cell k in the MSC, $W_p(k) = \min_{\forall x \in P}\{t_{min}(k,x)\}$ and $W_s(k) = \min_{\forall x \in S}\{t_{min}(k,x)\}$. And the cost of cell k, $Cost(k) = \eta_1 W_p(k) + \eta_2 W_s(k)$. $W_p(k)$ represents the proximity of the cell k from the set of cells P. A small value of W_p indicates that the cell is close to the set of cells P. For all cells in P, $W_p = 0$. For a small value of W_p, the cost is small and the cell is paged early. Similarly, W_s represents the proximity of a cell from the set $S(t_i)$, where the callee should have been found at t_i. The number of cells in P depends on the mobility pattern of the callee and the value of δ chosen. If P is large, then we should avoid paging all cells in P before paging the other cells in proximity of $S(t_i)$. That is, W_s must be given more weightage for large P. But for a cell k in P, the cost is $\eta_2 W_s(k)$, as $W_p = 0$. Hence, η_2 must be proportional to $|P|$. Since the other cells nearer to $S(t_i)$ must be paged before paging cells of P, η_1 must be 0 for these cells. That is, the determining factor must be W_s. It may so happen that the callee is supposed to be at his work place at time t_i, thus he would be staying at the same place during $[t_i - \delta,\ t_i + \delta]$. Here $P = S(t_i)$, that is the callee resides in a BP station during this time. Hence, $\eta_1 = |S(t_i)|/(|S(t_i)| + |P|),\ \eta_2 = |P|/(|S(t_i)| + |P|)$.

When $|P| \gg |S(t_i)|$, $\eta_1 \approx 0$ and $\eta_2 \approx 1$ and $Cost \approx W_s$. Thus, in this case the cells in the proximity of $S(t_i)$ are paged first. No additional preference is given to the cells in P. But if P is relatively small, then it would not cost us much to page the cells in P first and then the cells in the vicinity of $S(t_i)$. So additional preference is given to cells in P. Whereas if, $P = S(t_i)$ then $\eta_1 = \eta_2 = 0.5$. Thus, cost of each cell would be equal to $W_s = W_p$. Thus the cells in the vicinity of the BP station ($S(t_i)$) are paged first. After the calculation of the cost is done, the hunt algorithm can proceed by finding the best cell starting with a cell from D. Any improvement on design of cost function will improve the search and paging.

In case 1, $D = S(t_i)$. The paging starts from the nearest reachable neighbouring cells of D and then gradually proceeds farther away from the pattern to page all the possible reachable cells within the bound T till the callee is found. Depending upon the value of $|P|$, the cells in P or the cells nearer to $S(t_i)$ are paged first. If $|P|$ is large, all the cells nearer to $S(t_i)$ are paged first and then we gradually move away from $S(t_i)$. Whereas if $|P|$ is small, then the cells of P are paged first and then we gradually move away from P to page all the reachable cells. In case 2, the paging starts from $D = prev\_cell$ and then its reachable neighbour cells nearest to $S(t_i)$ or P, (depending upon the value of $|P|$) are paged first. In any case, the search starts from D proceeds towards the pattern till the bound is reached. Then, the other cells farther away from the pattern are searched till the callee found.

6 Simulation Results

The performance of the proposed scheme was evaluated for a MSC with 50 cells. The road connectivity i.e., the edges between the nodes and their weights were randomly generated. The total simulation time was 200 units. The calls for a host were also randomly generated within this interval. The scheme was simulated for a number of mobility patterns with varying degree of deviations. The deviations were randomly generated and the average search and the update cost were calculated for varying call-to-mobility ratios (CMRs). For the purpose of simulation, $S(t)$ is assumed to contain only one cell; and δ was taken to be 15 units. The total number of moves made by the host is taken to be 10. The plots in left half of figure 2 show the average number of cells paged for different CMRs and varying percentages of deviations. Whereas, the right half of the figure shows the average number of updates per call for different CMRs and varying percentages of deviations.

It is found that on the average only 1-2 cells are required to be paged. The paging cost is considerably less even for 30% deviation in the mobility pattern for the hosts with very high CMR. This is because, the host gets calls very frequently and thus would not be very far away from the position where it received the last call. Thus, the bound T would be much less for many cases. Consequently the paging cost is very small. But with a very low CMR, the cost would be high at large percentage of deviations. For very high CMRs, the update cost is found to

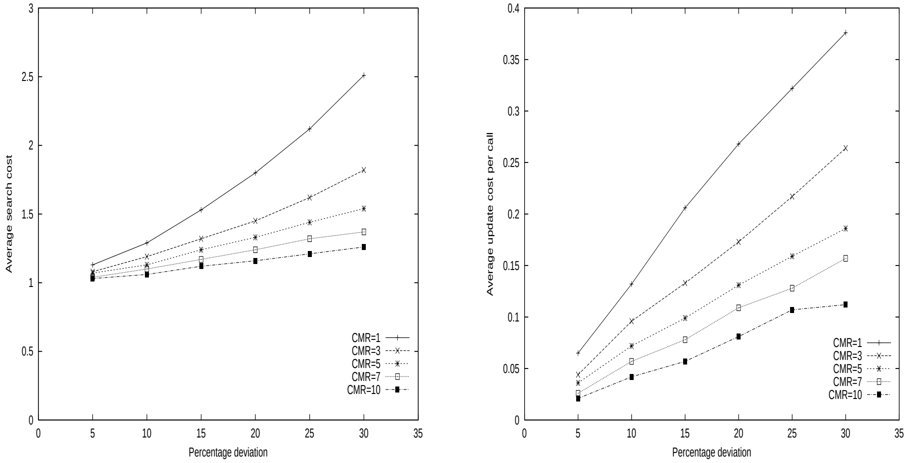

Fig. 2. The search and update costs

be very less. When the CMR is 10, the average update cost is found to be 0.021 per call. That is, the update takes place every 50 calls. Even for a deviation of 30%, the update takes place once for every 10 calls. But for 30% deviation with CMR=1, one update is required approximately after every 3 calls. However, for a deviation of 10% even for a very low CMR value, on the average one update takes place for every 10 calls.

7 Conclusion

In this paper we have proposed a scheme for locating a mobile host taking its spatio-temporal properties into account. The scheme uses the host mobility pattern and the regional route map of the of the last known location of the mobile host to determine the candidate cells where the target host may be found. The performance of the proposed scheme is found to be acceptable even when there are transient deviations upto 30% over the known host mobility patterns. The deviations are tackled using a best-first branch and bound search to locate the candidate cells for likely locations of the destination. The bound function is modeled to restrict the locality search space by using appropriate weighted functions of mobility pattern and regional route map.

References

[1] A. Bar-Noy, I. Kessler, and M. Sidi. Mobile users: To update or not to update ? *Wireless Networks*, 1(2):175–195, July 1995.

[2] Goutam Chakraborty. Efficient location mangement by movement prediction of the mobile host. In *IWDC, LNCS*, volume 2571, pages 142–153, 2002.

[3] G. Cho and L. F. Marshall. An efficient location and routing scheme for mobile computing environments. *IEEE Journal on Selected Areas in Communications*, 13(5):868–879, June 1995.

[4] A. Hac and Y. Huang. Location update and routing scheme for a mobile computing environment. *International Journal of Network Management*, 10:191–214, July-August 2000.

[5] B. P. Vijay Kumar and P. Venkataram. Prediction based location management using multilayer neural networks. *Journal of IISc*, 82:7–21, 2002.

[6] A. A. Siddiqui and T. Kunz. The peril of evaluating location management proposals through simulations. In *Proceedings of 3rd International workshop on Discrete algorithms and methods for mobile computing and communications*, pages 78–85, August 1999.