International Journal of Recent Engineering Science (IJRES),
ISSN: 2349-7157, *volume 1 Issue 4 April 2014*

"Dynamic Mobility Management Algorithm with Node Parameters to Optimize the Resource Utilization"

R.Abinaya[1] , Dr.S.Nithya[2]
M.tech , Professor
Department of Computer Science and Engineering,
Kodaikanal Christian College,
Kodaikanal.

Abstract

Mobility management plays a significant role in the current and the future wireless mobile networks in effectively delivering services to the mobile users on the move. This paper describes mobility management strategies from the point of view of their need of signalling and processing resources on the backbone network and load on the air interface. A method is proposed to model the serving network and mobile node mobility in order to be able to model the different types of mobility management algorithms. This dynamic mobility management algorithm can be easily implemented and integrated in mobile handsets, providing efficient mobility management for personal communications and mobile computing. Traditionally, mobility management includes functions to passively keep track of the location of the user/terminals and to maintain connections to the terminals belonging to the system. To make intelligent mobile-aware applications, it is important that a mobile terminal/system be more intelligent and can anticipate the change of the location of its user.

1. Introduction

Mobility Management is one of the major functions of a GSM or a UMTS network that allows mobile phones to work. The aim of mobility management is to track where the subscribers are, so that calls, SMS and other mobile phone services can be delivered to them. In wireless mobile networks, in order to effectively deliver a service to a mobile user, the location of a called mobile user must be determined within a certain time limit (before the service is blocked). When the call connection is in progress while the mobile is moving, the network has to follow the mobile users and allocate enough resource to provide seam-less continuing service without user awareness that in fact the network facility (such as base station) is changing. Mobility management is used to track the mobile users that move from place to place in the coverage of a wireless mobile network or in the coverage of multiple communications networks working together to fulfill the grand vision of ubiquitous communications. Thus, mobility management [7] is a key component for the effective operations of wireless networks to deliver wireless Internet services [6].

2. Various Mobility Effects for Communication

Mobility affects mobile communications on all the components, including devices, networks, and services [1]. To a mobile device, there are some requirements suitable to mobility scenario, e.g. weight, size, power, display etc. To a mobile service, the most important requirement is adaptation. A mobile service should be adaptive to different transmission links, different user mobile devices, and different using contexts. The mobility modes can be divided into three main classes.

- **Nomadic or portable communications** In which no network connection is needed during the movement. A new connection will be re-established only after the mobile node has arrived at its new location. Nomadic communications are not necessarily based on wireless networks.
- **Cellular communications** In which the wireless network is organized as a cellular structure. Each cell encompasses a certain area. Continuous connectivity should be provided when an on-served mobile node is moving from one cell into another (maybe either neighboring or overlapping cell).
- **Pervasive communications** In which the communications between mobile nodes are ubiquitous and invisible. The

scenario is based on a dynamic set-up without using any pre-existing network infrastructure, known as mobile ad hoc networking.

3. Network Graph and Node Mobility Parameters

We introduce how we will handle the networks on which the mobility management algorithms work. To derive the main parameters we will have to model the behavior of the mobile nodes as well.

- **Deriving Parameters of a Given Network** Let us have a given network topology with a given mobile node (MN) behavior. The network is modelled with a graph just like the possible movements of the mobile. We assume that the behavior of the MN can be modeled with a continuous Markov chain [3]. The chain is given with a rate matrix BQ = $[b_{i,j}]$.

Let us have the corresponding network graph given with its adjacency matrix: A. this matrix should include all the nodes in the network where the mobility application runs (all the MAs again) so has the same $n \times n$ size as matrix B_Q and B_Π. With the Floyd algorithm the optimal distances between the nodes can be calculated even with weighted or directed edges as well). The distance between nodes will be the sum of weights on the shortest path from one to the other. Let this result matrix be given by A_d. In the i^{th} row of the matrix, the distances from FA_i are listed. Let the distances from the HA, - a special FA, - be given with the vector $\underline{a}$. We will parameter w to denote the average of the weights in the network. It can be calculated by summing up the elements of A and dividing it with n^2.

- **Determining m** Parameter m will denote the average depth level that is the average number of edges on the shortest path from the MN to the home agent, HA. Clearly, the average number of vertices among the path is m+ 1.

We will use matrix A_d and vector $\underline{a}$ to calculate this parameter. Both have to be normalized with the average weight of edges in the network (w). Now mw can be calculated with determining the weighted average of the distances where the weights are the probabilities that the node is under a given MAP.

$$m = (\underline{a} * \underline{b}) / w$$

Where * stand for the scalar product. One can see that the nodes which are not MAPs have a 0 multiplier and does not count in the average distance as expected.

These parameters m, mw we learnt shows the real average number of edges and distance along the shortest path from the HA. If there is a call or delivery request, we suppose that it is routed on this optimal path towards the current MA of the MN. In this sense these are the smallest values of our parameters.

A tree-like hierarchy could be introduced on the same network in such a way that the average distance becomes equally big or bigger than m. Also other protocols might use a different m. It is a question whether the smallest m has the optimal overall cost in all case one or not.

- **Parameter gT** Another parameter assumed like ‘m’ is the average distance between two nodes who handles the MNs handovers. They might be connected, but they can also be quite far from each other logically due to different technologies especially in the case of vertical handovers. So as we see this parameter has to denote the weighted average value of the length between every two neighboring MAs where the mobile can attach. Than it is calculated as follows:

$$gT = \{\underline{b} * tr(A_d . B_\Pi)\} / w$$

We denote it with gT since this parameter will have the most effect on the Tracking-like management solutions.

- **Parameter gH** This parameter denotes how far is the nearest hierarchical junction to register at in average if we consider the optimal covering tree of the network with the HA in the root. The junction node is the nearest common node of the paths from HA to the old and the new foreign agent, FA of the MN. (In most cases, the optimal tree structure is not possible to achieve since the different service providers will not mesh their networks: approximate values can be used instead [5]).

Our hierarchical structure will be built up using two main parameters. First is the number of nodes: 'n'. the other parameter is the average number of neighboring Mas that can be accessed via a wire from a given node: 'δ'. It should be also weighted with the probability density of the MN:

$$\delta = \{(\underline{b} . sign(B_{\Pi})). \underline{1}\}/n$$

- **Parameter gC[4]** This parameter will denote the average distance of MAPs from the main MA of a Location Area in the Cellular-like approaches. It is an NP complete problem to calculate the optimal cell structure.

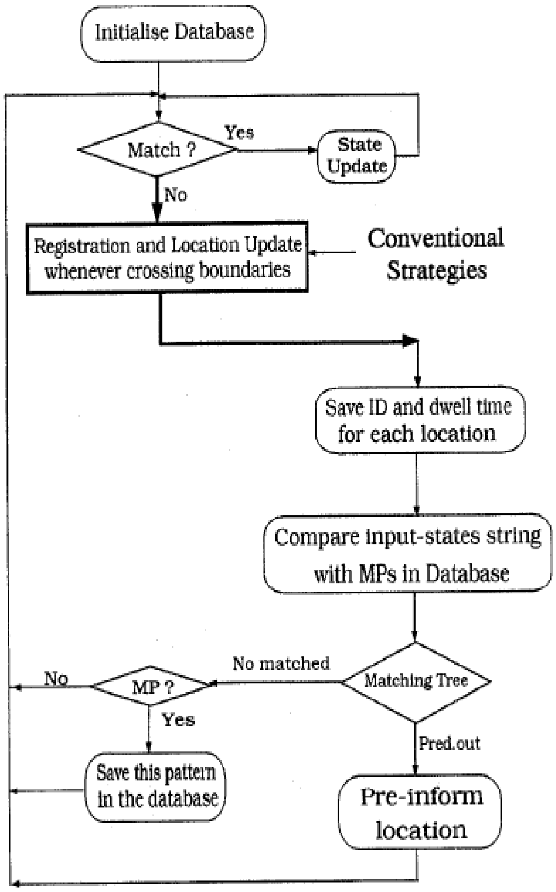
4. Proposed Algorithm

A. Design Strategy:

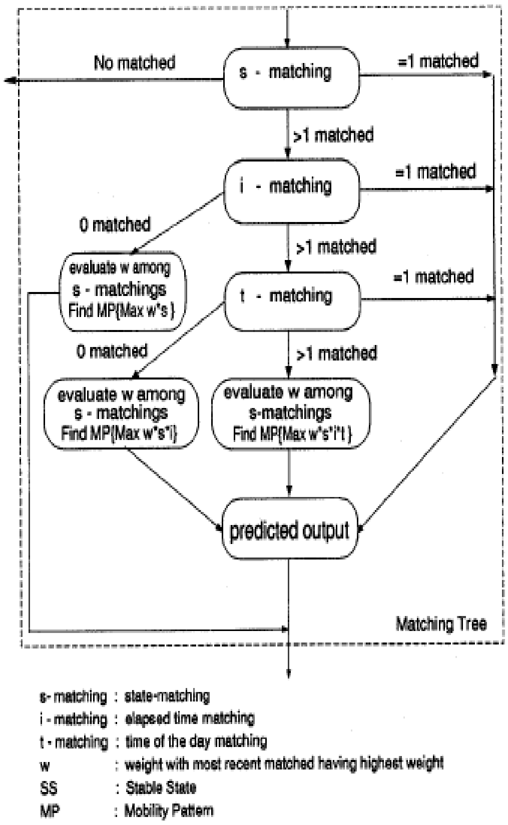
A set of Mobile Motion Prediction algorithms can be used to predict the 'future' location of a mobile user according to the user's movement history patterns or some information directly from the users. Location prediction is carried out only in the mobiles. The resulting predictions are passed to the current location server as a form of advance information. The server is therefore able to inform the 'next' servers (via the wired network) of the imminent arrival of the mobile, allowing a reduction in signalling traffic. If the prediction turns out to wrong, the conventional registration comes into effect. Instead of using a completely prediction-based scheme, a flexible strategy is proposed so that a mobile can dynamically participate in mobility management. It is designed as follows:

Only mobiles keep a record of their own movement history, by saving their location identity numbers (IDS), dwell time and other related information after they have first registered with their location servers. By comparing the input sequence trajectory with movement patterns in the Database, a mobile, if a match is found, sends its prediction to the current location server which will then pre-inform the rest of the predicted location servers through wired network, therefore it can virtually use the services without conventional registration and location updating procedures when they arrive at the new location area. A mobile automatically compares its predicted location ID with its current received location ID. If a false prediction is found, for safe reason, it still has chance to register with its current location server.

B. Algorithm Description:



When a mobile is first used, the user can input its routine or regular movement data to its database. The systems work as usual, but a mobile should keep a queue (or stack) to save its recent movement trajectory (a sequence of its location server IDS and the corresponding dwell time, etc.). When entering a new location area, mobile checks its previous prediction record against current location ID; if a match is found, it updates its own location states and no conventional registration procedure is needed, otherwise it initiates the conventional registration protocol.



Matching Tree

Three matching criteria[2] that is, state matching, elapsed time matching and time of the day matching are used. Among those matched by the mentioned criteria, weighted matching is applied to find the best prediction, and the weight is calculated in such a way that the most recent location matching is assigned the highest weight. Denote current matching pattern as $a_{t0}, a_{t1}, \ldots\ldots, a_{tn}$, then its weight w = $a_{t0}*2^0 + a_{t1}*2^1 + \ldots\ldots\ldots\ldots + a_{tn}*2^n$ For example, if the current matching pattern is 11101011, with the most significant bit representing the most recent location matching result, here, '1' indicates matched; '0' indicates unmatched, the weight of value 235 is assigned. The length of patterns is decided depending on the division of the system service location areas, usually to be 16, 32, or, 64 bits. The number of mobility patterns or size of the database may be decided according to individual mobile's mobility characteristics and its memory capacity. Proposed algorithm use 30 enteries as an experimental parameter.

5. Conclusions:

The paper examines numerous significant parameters of mobility and modelled the mobile node behavior as well as the network and some general management strategies. It can be shown which mobility management gives the optimal solution in different network scenarios and which aspect of resources could be a bottleneck in each case.
The secondary aim, that is part of future work, is to use the measurements to give guidelines for the design of new mobility management algorithms and give proposals for solutions on different requirements. This paper has presented a flexible mobility management algorithm for future personal communications and mobile computing. The performance evaluation indicates that the proposed algorithm will significantly improve existing systems performance by reducing mobility traffic overhead and saving wireless network capacity. The algorithm can be easily and flexibly implemented in mobile handsets with minimal change in the network.

Reference:

[1].A.Abutaleb and V. O. K. Li, "Paging strategy optimization in personal communication systems," ACM Wireless Networks, vol.3, pp.195-204, 1997.

[2]. George Y Liu, Gerald Q Maguire Jr., "A Predictive Mobility Management Algorithm for Wireless Mobile Computing and Communications", Proceedings of ICUPC'95, Tokyo, Japan, Nov.6-10, 1995.

[3] Fang, Y. and Lin, Y.: Portable Movement Modelling for PCS Networks. IEEE Transactions on Vehicular Technology. (2000) 1356–1362.

[4]. Kovacs, B. Szalay M. and Imre, S.:Modelling and Quantitative Analysis of LTRACK - A Novel Mobility Management Algorithm. Mobile Information Systems, Issue: Volume 2, Number 1/2006 (2006) 21–50

[5].A. T. Campbell, J. Gomez, S. Kim, C.-Y. Wan, Z. R. Turanyi and A. G. Valko, "Comparison of IP micromobility protocols," IEEE Wireless Communications, February 2002.

[6].F. Akyildiz, J. McNair, J.S.M. Ho, H. Uzunalioglu and W. Wang, "Mobility management in next-generation wireless systems," Proc.of the IEEE, vol.87, no.8, pp.1347-1384, August 1999.

[7].Wireless Communications Systems and Networks ,Edited by Mohsen Guizani, Western Michigan University Kalamazoo, Michigan