Trajectory Predictions in Mobile Networks

Shiang-Chun Liou [1], Yueh-Min Huang [2]

Dept. of Engineering Science, National Cheng Kung University, 701 Taiwan[1,2]
Dept. of Computer Science and Information Engineering, Leader University, 709 Taiwan[1]

1 dahlia@mail.leader.edu.tw
2 huang@mail.ncku.edu.tw

Abstract

This paper proposes a mobility model and a neural location predictor (NLP) to predict the location of a mobile host (MH). The seemingly random movement is actually a logical function of the user's position, speed, acceleration and direction. (x,y) of a coordinate should be treated independently such that NLP can be designed with fewer mathematic operations and achieves high computing efficiency. Accurate and speedy location prediction can also minimize signaling traffic required by QoS maintenance during seamless handoff.

Keyword: Trajectory Prediction, Neural Location Predictor, and Mobility Model

I. Introduction

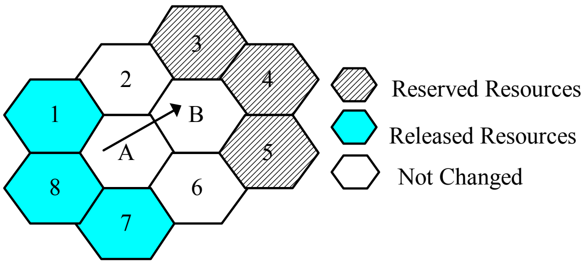


Fig. 1. Bandwidth reservation procedure

A cellular network [1] consists of cells. As mobile users travel, their conversations are handed off between cells to maintain a seamless quality of service (QoS). This handover procedure is critical to ensure the continuity of a call.

Figure 1 shows an example of bandwidth reservation procedure during handoff. In papers [2][3][4], bandwidth is allocated to the connection in the current cell A, and bandwidth is reserved in all neighboring cells B, 1, 2, 6, 7 and 8. When the user moves from cell A to cell B, the reserved bandwidth (and additional unused bandwidth, if there is any) in cell B is used to accommodate the hand-off connection, and the reserved bandwidth in cells 1, 7 and 8 (*i.e.,* cells which are not adjacent to cell B) is released. At the same time, bandwidth in cells A, 3, 4 and 5 (*i.e.,* new neighboring cells) is reserved.

An accurate and speedy location prediction can greatly reduce the amount of reserved bandwidth when a MH is going to move across a cell boundary. It is not necessary to reserve bandwidth in all neighboring cells, if a location predictor can take moving direction into consideration and can make an accurate cell-crossing prediction in time. Therefore, it is possible to reserve bandwidth in cells B, 2 and 6 (even one or two cells only) but not in all neighboring cells. Therefore, location prediction is a topic worth studying to ensure the continuity and integrity of a wireless connection and to improve the QoS in 3G mobile technologies.

II. Location Management

Location management methods of 3G mobile network have been calssified into two major groups[5]: memory-less and memory-based. According to the classification rule, the NLP belongs to the memory-based category that predicts short-term movements of MHs. When a movement of a MH does not fit with location prediction, a location update is triggered by the MH to inform the network about its actual location. Otherwise, no signalling exchange is required. Therefore the NLP approach allows savings in location update process and signalling traffic exchange.

III. The Proposed Models

A. *Mobility Model*

Intercellular movement is actually a logical function of position (p_t), speed (s_t), acceleration (a_t) and direction (d_t). Furthermore, a series of MH's positions implicitly contain speed, acceleration and direction. Each position p is composed of a pair coordinate (x,y). Variable t stands for time. $\triangle p$ is a vector ($\triangle p = p_t - p_{t-1}$), therefore $\triangle p$ can be used instead of direction d_t. $\triangle p / \triangle t = v_t$ can be used instead of speed s_t. $\triangle v / \triangle t$ can be used instead of acceleration a_t.

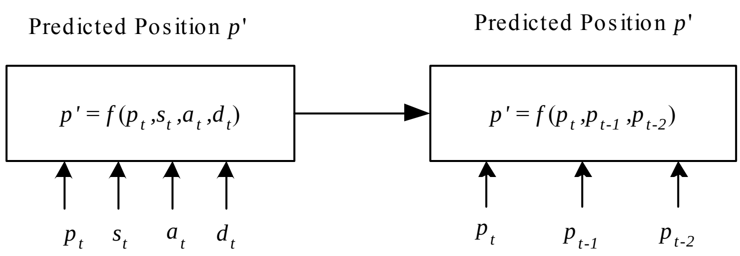


Fig. 2. Proposed mobility model

$$v = v_0 + at \tag{1}$$

$$s = v_0 t + 0.5at^2, \quad s : displacement \tag{2}$$

$$x = x_0 + s\cos\theta_0 \tag{3}$$

$$y = y_0 + s\sin\theta_0 \tag{4}$$

After repetitive substitutions, the mobility model in Figure 2 can be transformed from $p' = f(p_t, s_t, a_t, d_t)$ to $p' = f(p_t, p_{t-1}, p_{t-2})$, when the timing interval of position samples is constant. It is more practical and convenient to use the transformed mobility model to predict locations, because the transformed model needs only three parameters and (p_t, p_{t-1}, p_{t-2}) require less pre-computation than the other model. Equations (1)-(4) show the well-known substitution relation among position, speed, acceleration, and direction. Two sequential position samplings can approximate speed $v_t=\triangle p/\triangle t$. Three sequential position samplings can approximate acceleration $a_t=\triangle v/\triangle t$. This is the reason of using three positions but not four or five positions in the mobility model.

B. *Neural Network Model*

Not only the transformed mobility model but also the coordinate decomposition concept is very important in designing the neural location predictor (NLP). Decomposition let us split three nearest past positions (x_1,x_2,x_3,y_1,y_2,y_3) into two small groups ($x_1,x_2,x_3,$) and (y_1,y_2,y_3). Therefore the NLP architecture should be designed in two-group-complete-bipartite-meshed feed-forward neural network rather than one-group-complete-bipartite-meshed feed-forward neural network. Figure 3 and Figure 4 tell the difference between both architectures. As can be seen from the figures, NLP use fewer edges. Each edge represents one multiplication and one summation. The inputs of both neural networks are the same. Three nearest past positions (x_1,x_2,x_3,y_1,y_2,y_3) are fed into networks repetitively. Location predictions (x_4,x_5,x_6,y_4,y_5,y_6) and cell predictions (c_4,c_5,c_6) are the outputs. In NLP, intermediate coordinate pairs (x_4,y_4), (x_5,y_5) and (x_6,y_6) are also treated independently. Therefore the number of edges is also reduced when it is mapping from (x_4,x_5,x_6,y_4,y_5,y_6) to (c_4,c_5,c_6).

NLP uses fewer operations of addition and multiplication as compared to the one-group-complete-bipartite-meshed feed-forward neural network. Therefore NLP achieve better computational efficiency than the full-mesh one. It is assumed that each MH can periodically obtain its position (x,y) by using methods in section 4. NLP starts in initialization phase until it gathers the first three planar-coordinates (p_1, p_2, p_3) =(x_1, y_1, x_2, y_2, x_3, y_3). The information about speed, acceleration, and direction were concealed in the coordinates.

NLP has computational simplicity, because what it needs is only three nearest past positions. Other predictors may require more complicate inputs like speed, acceleration and direction that need many pre-computations. The proposed mobility model and NLP predictor provides a unique architecture. Also, J. Biesterfeld et al. [6] has done examination and found that feed-forward networks delivered better results than the feedback networks. The proposed NLP (Figure 4) looks similar to a traditional one (Figure 3), and their time complexity are the same $\boldsymbol{O(l*m + l*n)= O(l*(m + n))}$. The details are given in Appendix. However the proposed NLP indeed achieves computation time reduction in practical programming. The following gives a proof of NLP's computational reduction.

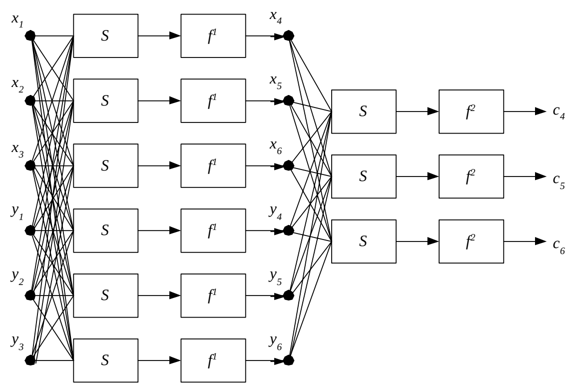


Fig. 3. One-group-complete-bipartite meshed feed-forward neural network (n=3)

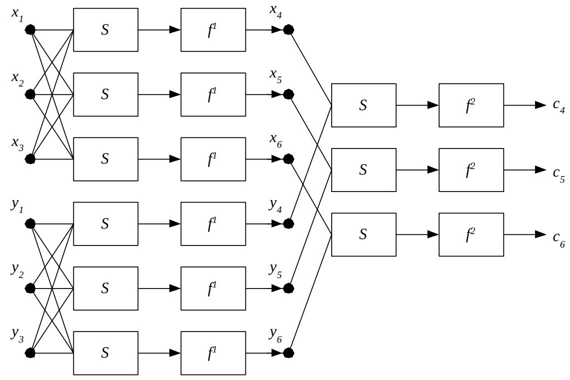


Fig. 4. Proposed NLP architecture (n=3)

Theorem 1: If n positions are used to predict the next crossing cell, then the time complexity reduction ratio of NLP is

$$R = (2n^2 + 6n)/(6n^2 + 4n) \quad (5)$$

Proof: A complete bipartite graph $K_{m,n} = \{V, E\}$ where $V = \{V_1 \cup V_2\}$ and each vertex in V_1 is adjacent to all the vertices in V_2. There are $(m*n)$ edges in graph $K_{m,n}$. In Figure 3 and 4, a solid edge without arrow represents a multiplication and an addition operation, and a solid edge with arrow denotes a data transfer.

The one-group-complete-bipartite meshed feed-forward neural network does not treat (x,y) independently, such that the total number of edges is $2n * 2n + n + n + 2n * n + n + n = (6n^2 + 4n)$. After splitting inputs into two groups, the total number of edges in NLP network will be $2(n * n) + n + n + 2n + n + n = (2n^2 + 6n)$. The complexity reduction ratio of mathematical operation is $R = (2n^2 + 6n)/(6n^2 + 4n)$. As $n \to \infty$, $\lim R = 2/6 = 1/3$.............. ■

C. *Two-tier Cell Structure*

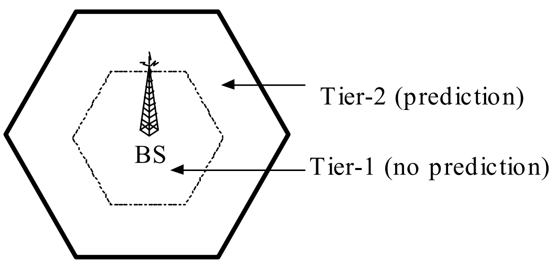


Fig. 5. Two-tier cell structure

Let us consider a real scenario of bandwidth reservation problems. If a MH resides in a particular cell for a long time before handoff, bandwidth is wasted due to long-time reservation. Furthermore, Next crossing cell prediction will be less accurate if a MH appears in tier-1 area and it make a big turn. Less prediction accuracy results in more signaling exchanges and more waste of unused bandwidth reservation. To prevent such problems, the 2-tier cell structure [7] should be adopted as shown in Figure 5.

Increasing propagation distance between a BS and a MH degrades RSSI [8]. A MH's handoff probability increases when its distance to a BS increases. A 2-tier cell structure utilizes RSSI to predict the right moment of triggering bandwidth reservation process before handoff. A MH with a RSSI greater than the average signal strength is located in tier-1; otherwise, it is in tier-2. BS should reserve bandwidth for a MH located in tier-2, because the MH has higher handoff probability than that in tier-1. Due to 2-tier architecture, RSSI is helpful to reduce waste of bandwidth, reduce amount of signaling traffic and increase bandwidth utilization.

IV. Position Measurement

There are three methods to measure a location of a MH. Tong Liu et al. proposed the first method [8]. The RSSI (p_i) from a particular $cell_i$ can be formulated as

$$p_i = p_{oi} - 10r \log d_i + \xi_i \tag{6}$$

Where p_{oi} is a constant determined by transmitted power, wavelength, and antenna gain of $cell_i$. r is a slope index (typically, r =2 for highways and r =4 for micro cells in a city), and ξ_i is the logarithm of the shadowing component, which is found to be a zero-mean Gaussian random variable with standard deviation 4–8 dB. d_i represents the distance between the mobile and base station of $cell_i$, which can be further expressed in terms of the mobile's position ($x(n),y(n)$) at time and the location of base station (a_i, b_i)

$$d_i = \left[(x(n) - a_i)^2 + (y(n) - b_i)^2 \right]^{1/2} \tag{7}$$

To locate a moving user in the two-dimensional domain, at least three independent distance measurements are needed [11].

The second method of gathering MH's x-y coordinates is GPS. It uses a series of satellites that are constantly circling the earth to triangulate your position. Once your receiver has a lock on at least three satellites, your position can be triangulated to within 3 meters (~ 10 feet) of your exact location. The third method is based on GPS proposed by Bell Labs called Wireless Assisted GPS (WAG). It is accurate to within 15 feet when users are outdoors and within 100 feet when indoors. Up to now, GPS was considered unsuitable for wireless 911 because GPS signals were too slow for emergency use, consumed lots of power, and were too costly or too bulky. Also, they did not work inside buildings or shadowed environments.

V. Performance Analysis

To reduce signaling traffics, it is necessary to achieve high accuracy of location prediction. The user mobility models found in literatures [9] [10] [11] [12] assume straight-line movement or constant speed, which cannot reflect reality. The algorithm (Figure 6) guarantees the QoS during handoff. It can deal with the wrong prediction that cause invalid reservation.

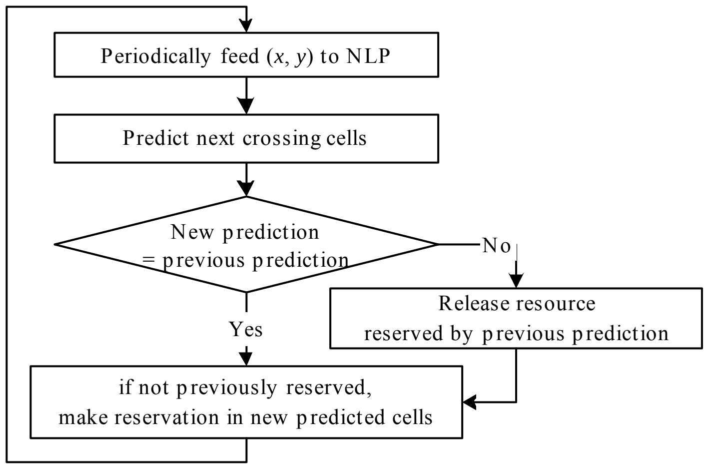


Fig. 6. NLP and reservation mechanism

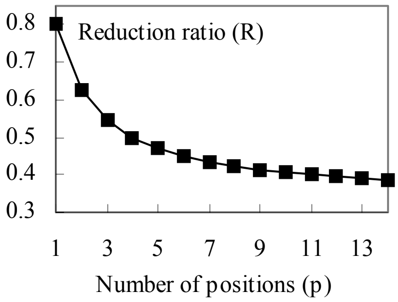


Fig. 7. Reduction ratio (R)

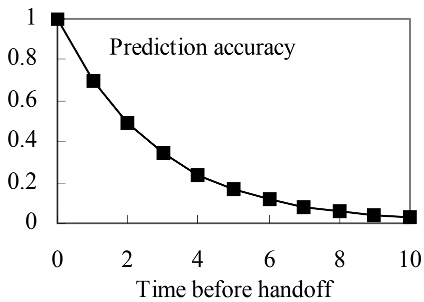


Fig. 8. Time before handoff and prediction accuracy

Figure 7 draws the curve of $R = (2n^2 + 6n)/(6n^2 + 4n)$ in Theorem 1. It seems that more position gathering achieves higher reduction ratio. However, if too many historical data (eg. 10 nearest past positions) are used, NLP becomes too conservative and cannot immediately respond to situations such as making a big turn and sudden brake. On the other hand, prediction accuracy drops very quickly if NLP makes a prediction as a MH still far from handoff time. Figure 8 shows an example of the relationship between time before handoff and prediction accuracy. When a MH appears in a tier-2 area, the next-cell prediction can be made more accurately.

NLP predicts a future location based on three location updates of nearest past. It assumes that each node can periodically obtain its location. The timing interval of position sampling should be a constant. How can a designer decide the frequency of position update? Figure 9 analyzes the relation between update frequency (sampling frequency) and MH velocity. Update frequency can be vary linearly between a maximum (F_{max}) and a minimum (F_{min}) threshold. Frequency should not greater than $F_{max.}$ The faster a MH (eg. a car) runs, the more linear its trajectory is. At such a situation, sampling frequency should keep constant, otherwise increasing frequency causes location predictors unnecessary computation overhead.

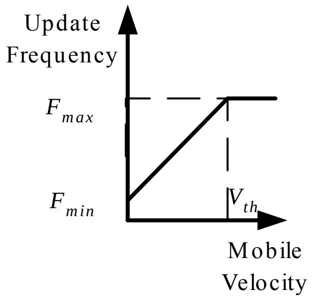


Fig. 9. Update frequency and mobile velocity

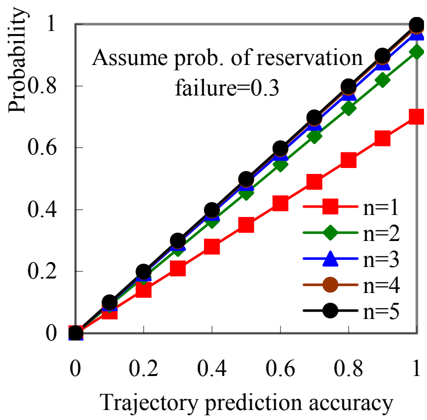


Fig. 10. Probability of successful reservation and prediction accuracy

Algorithms having higher next-cell prediction accuracy will improve QoS and reduce signaling overhead during handoffs. However, it is useless if a prediction algorithm is too complicated to give a good enough prediction to the resource administrator before a MH entering next cell. In addition, before the moment of handoff, a MH may fail to reserve resource because of running short of resource or too late to make a reservation, and then such a situation is also useless.

To overcome these problems, NLP and reservation process work as follows. If an accurate prediction can be made a few sampling periods (n = 2, 3…) before handoff, then the reservation process can be reserved again (and again) if the previous reservation fails. Sampling period is an inverse of update frequency of position sampling, *i.e.* period is the timing interval between successive position updates. If an algorithm can only make a prediction one period before handoff (n = 1), there will be no chance to reserve again when it fails to reserve resource in the first trial. As a consequence the mobile terminal may encounter a massive loss during handoff and QoS drops. Figure 10 shows the relation of prediction accuracy to the successful probability after n times reservation trial with different periods (n = 1, 2, 3, 4, 5). It is assumed that the probability of reservation failure is 0.3. It is obvious that n = 1 is very poor in performance. When $n \geq 3$, the curves almost overlap. Figure 10 also tells us that increasing prediction accuracy and making prediction three or more periods before handoff will make the probability of successful reservation dramatically increase.

Figure 11 demonstrates the prediction-based reservation can effectively reduce the amount of used resources (reserved bandwidth). A successful prediction can save resources including radio capacity and signaling traffic, because all neighbor reservation needs to notify all neighbors to make a reservation and it wastes more resources than prediction-based reservation does. Prediction-based reservation is more efficient because it may notify only one neighbor to make a reservation. When prediction failure happens, there must exist a mistaken reservation. Therefore it should not only predict and reserves again, but also release the incorrectly previous reservation.

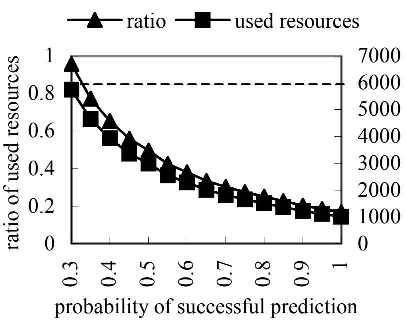


Fig. 11. Reduction of used resources

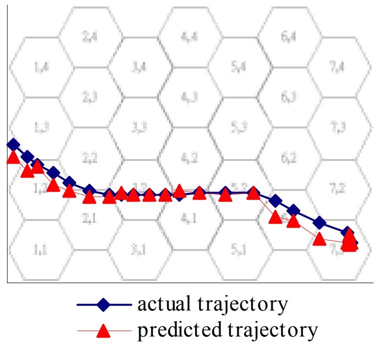


Fig. 12. Actual and predicted user trajectories

To better understand how many resources are required, a simulation environment is designed. In this environment, there are 1000 times of cell crossing events. A cell has six neighbors. For all neighbor reservation, each crossing event requires six units of resources. For prediction-based reservation, it depends on probability, so that each crossing event requires uncertain units of resources. In the 1000 times of cell crossing events, the total amount of used resources is accumulated for both kinds of reservation methods (all neighbor and prediction-based).

The horizontal dash line in figure 11 reveals the constant amount of resources used by all neighbor method. The solid line with box spot represents the variable amount of resources used by prediction-based method. That line varies with different probabilities. The higher successful prediction probability, the lower amount of used resources. The solid line with triangle spot shows the ratio of used resources of two methods. It was found that when probability of successful prediction is lower than a level (e.g. 0.3), prediction-based method is even worse than all neighbor method. The reason of the phenomena is that prediction failure may cause several times of invalid signaling traffics and that will waste lots of resources. In the contrary, prediction-based reservation method with higher prediction success rate can effectively reduce the amount of used resources to almost 1/6.

Figure 12 shows the NLP's prediction results of user trajectory tracking. Cells are numbered according to the x-y coordinates. As a result, cell number is a function of a cell location. The curve with triangle points shows the actual trajectory and the curve with square points depicts the predicted trajectory.

$$\text{Prediction accuracy} = 1 - \left(\frac{\text{mis-predicted times}}{\text{total times}} \right) \qquad \textbf{(8)}$$

On analyzing the prediction results of the next crossing cell and user trajectory, It was found that NLP achieves a high degree of prediction accuracy, and accuracy is about 78% on average for Mountain road to Alishan national scenic area (see figure 17).

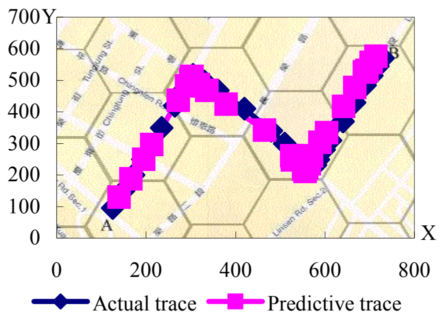


Fig. 13. Right angular turns in Tainan City

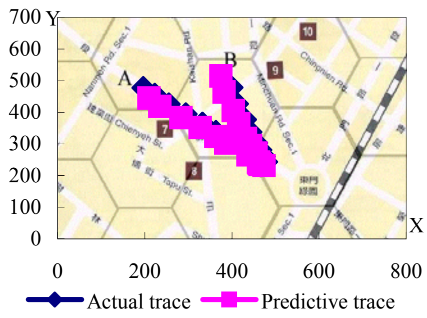


Fig. 14. A larger degree turn (>90°)

In the next step, the designer will simulate the moving car and predict its trace. Most of streets in cities are almost linear, and car driver only turn left or turn right. Therefore, a city street simulation is set up according to the streets of the Tainan City, which is a southern city in Taiwan. Figure 13, a car is moving from point A to point B and its moving path (trajectory) is almost straight with two right angular turns. Linear trajectory with some 90-degree turns is very common in city travel. Measured car speed is between 0 ~ 50 km per hour. Observation from the figure shows that actual trace and predicted trace are very close. It encourages us to do more experiments. Later the turning degree is changed. Figure 14, a car is moving from point A to point B, but turning degree is larger than 90 degree. Actual trace and predicted trace are still very close. It means that the prediction in linear moving path with some turns is highly accurate.

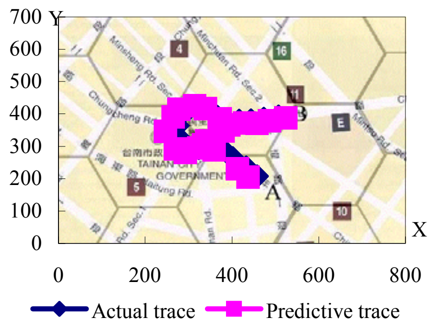


Fig. 15. Traffic circle on Minsheng Road

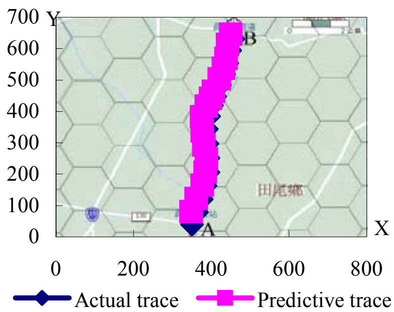


Fig. 16. National freeway at Yuanlin Township

Next, the designer will do a trace prediction on a traffic circle. Figure 15 shows that actual trace and predicted trace are still very close, but there is a difference near the traffic circle. That is because the moving direction of a car is continuously changing for a period around the circle, and the NLP predictor have to adapt to the changing direction. In spite of a little difference, the predictor still achieves high prediction accuracy. Similar experiments have been done for several times, and the accuracy of the predictor is 86% in city street prediction.

After that, a freeway simulation is set up according to the number one national freeway at Yuanlin Township, which is in central Taiwan. Figure 16 shows that a car is moving from point A to point B. The characteristic of a freeway is linearity with some smooth curves. But car speed in freeway is much faster than car speed in city. Measured car speed is between 80 ~ 100 km/h. Due to the high speed in freeway, car drivers usually do not make big turns as they do in city corners. In freeway, the NLP predictor achieves its highest accuracy, which can be over 90%. Due to the high speed, high frequency of lane changing and traffic jam, the predictor is still affected by some unpredictable factors.

Finally, a mountain road simulation to Alishan national scenic area is set up. Figure 17, a car is moving from point A o point B along a typical snake like mountain road. Measured car speed is about 0-40 km/h. There are many big turns in series in a short distance. It is a real challenge to the NLP predictor. Due to too many big turns in series in a short distance, the prediction accuracy in Alishan mountain road is less than that in city streets. Mountain road prediction should be worse. Even that the NLP predictor still exhibit its high accuracy and well adaptation to complicated mountain road. In general speaking, the predictor still performs well. In such road, its accuracy is about 78%.

There are some prediction schemes introduced by J. Chan et al. [13] [14]. A brief description of these prediction algorithms is given in Table 1. Most of these schemes use simulated user movement patterns to evaluate their performance. Figure 18 presents the accuracy comparison of some studied prediction algorithms. The result supports the earlier claim that NLP achieves a high degree of prediction accuracy and is an efficient and effective method based on neural networks for location prediction with less addition and multiplication operations.

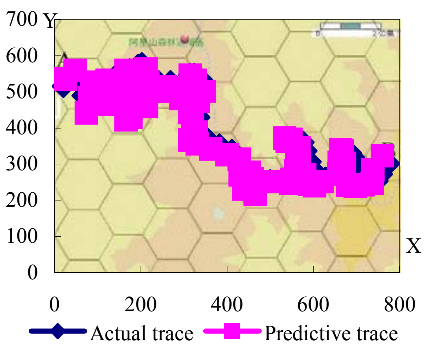


Fig. 17. Mountain road to Alishan scenic area

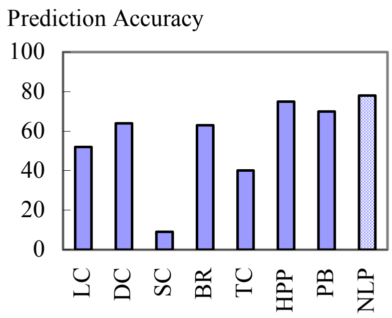


Fig. 18. Accuracy Comparison

Table 1. Some prediction algorithms

Scheme	Description
Location Criterion	The LC identifies present location and uses the departure history of this location to predict.
Direction Criterion	The DC incorporates direction information with the departure history. It identifies present direction and uses departure history of this direction to predict.
Segment	The SC extends the DC algorithm. All previous

Criterion	movement paths are partitioned into segments and stored. A segment starts with a stationary cell in which a MH stays for a sufficiently long time. SC keeps on matching the current segment with those stored segments. A match is found if the present segment is identical to the initial portion of a stored segment. Then the cell right after matched segment is predicted next move.
Bayes' Rule	Bayes' Rule is used to calculate the probability of all possible next moves once a reference point and a direction of travel is given.
Time Criterion	The TC algorithm imposes the time of cell crossing (accurate to the nearest hour) into the DC algorithm. It assumes that people trend to have regular daily activities; the collected probability describes the movement.
Hierarchic Position Prediction [8]	HPP is based on movement history and instant RSSI. Movements can be estimated by current location, velocity and cell geometry.
Profile Based Next-cell Prediction [15]	PB is based on the movement history and the classification of locations. Mobility can be purely random, mostly random, purely deterministic, and mostly deterministic.

VI. Conclusion

Existing mobility management methods do not take advantage of all the information that is possible to get from networks and users' behaviors. In other words, the information contained in users' motion pattern is not used to reduce signaling amount and to improve the bandwidth utilization. On the other hand, location prediction can effectively help reservation schemes provide advance reservation during handoff.
A mobility model and NLP predictor are proposed for predicting the future locations of a MH, and prove the effectiveness and efficiency of the proposed NLP by trajectory simulation. Many factors' relationships demonstrated by the designer that are important to practical location prediction. The simulation results confirm that NLP can achieve high prediction accuracy. The results are encouraging.

References

[1] S. K. DAS and S. K. SEN, "Adaptive Location Prediction Strategies Based on a Hierarchical Network Model in a Cellular Mobile Environment," IEEE The Computer J. vol.42, pp.473-486, 1996.
[2] C. Oliveiray, J.B. Kimz and T. Suday, "An Adaptive Bandwidth Reservation Scheme for High-Speed Multimedia Wireless Networks," IEEE JSAC., vol.16, no.6, pp.858-874, Aug. 1998.
[3] D.A. Levine, I.F. Akyildiz, and M.Naghshineh, "The shadow cluster concept for resource allocation and call admission in ATM-based wireless networks," IEEE/ACM Trans. Networking, vol.6, pp.1-10, Feb. 1997.
[4] G.S. Kuo, P.C. Ko and M.L. Kuo, "A Probabilistic Resource Estimation and Semi-Reservation Scheme for Flow-Oriented Multimedia Wireless Networks," IEEE Commun. Mag., pp.135-141, Feb. 2001.
[5] S. Tabbane, "Location Management Methods for Third-Generation Mobile Systems," IEEE Commun. Mag., vol.35, pp.72-84, Aug. 1997.
[6] J. Biesterfeld, E. Ennigrou, K. Jobmann, "Location Prediction in Mobile Networks with Neural Networks," Proc. the Intl. Workshop on Applications of Neural Networks to Telecom., pp.207-214, June 1997.
[7] C. H. Choi, M. I. Kim, T. J. Kim, S. J. Kim, "Adaptive Bandwidth Reservation Mechanism Using Mobility Probability in Mobile Multimedia Computing Environment," Proc. IEEE LCN 2000, pp.76-85
[8] T. Liu, P. Bahl, and I. Chlamtac, "Mobility Modeling, Location Tracking, and Trajectory Prediction in Wireless ATM Networks," IEEE JSAC., vol.16, no.6, pp.922-936, Aug. 2001.
[9] M.D. Austin and G.L. Stuber, "Velocity adaptive handoff algorithm for microcellular systems," IEEE Trans. Veh. Technol., vol.43, pp.549-561, Aug. 1994.
[10] M.D. Austin, and G.L. Stuber, "Direction Biased Handoff Algorithms for Urban Microcells," Kluwer Wireless Personal Commun., vol.3, no.3, pp.287-298, 1996.
[11] R. Viayan and J.M. Holtzman, "A model for analyzing handoff algorithms," IEEE Trans. Veh. Technol., vol.42, pp.351-356, Aug. 1993.
[12] D. Hong and S.S. Rappaport, "Traffic model and performance analysis for cellular radio telephone systems with prioritized and nonprioritized handoff procedures," IEEE Trans. Veh. Technol., vol. VT-35, no.3, pp.77-92, Aug 1986.
[13] J. Chan, A. Seneviratne, "A Practical User Mobility Prediction Algorithm for Supporting Adaptive QoS in Wireless Networks," Proc. ICON'99, pp.104-111, Sept. 1999.

[14] J. Chan, S. Zhou, A. Seneviratne, "A QoS Adaptive Mobility Prediction Scheme for Wireless Networks," Proc. IEEE GLOBECOM'98, vol.3 pp.1414-1419, Nov. 1998.
[15] V. Bharghavan and J. Mysore, "Profile Based Next-cell Prediction in Indoor Wireless LAN," Proc. IEEE Singapore International Conference on Networking, Apr. 1997.
[16] Chin-Teng Lin and C.S. George Lee, "Neural Fuzzy Systems – A Neuro-Fuzzy Synergism to Intelligent Systems," Prentice Hall Intl. Ed., pp237

Appendix: Complexity Analysis of NLP

From equation (9-12), the time complexity to compute *each* y_i is $\boldsymbol{O(l*m)}$. Furthermore the total time complexity to compute *all* $y_i=(y_1,..y_i..,y_n)$ is not $\boldsymbol{O(l*m*n)}$ but $\boldsymbol{O(l*m + l*n) = O(l*(m + n))}$. The reason is that $(z_1,..z_i..,z_n)$ had been computed to determine the value of y_1. The solid line with arrow in Figure 19 helps readers to better understand the computation of y_1. There is no need to re-compute $(z_1,..z_i..,z_n)$ again and again to determine the value of $(y_2,..y_i..,y_n)$. If it is based on the computed $(z_1,..z_i..,z_n)$, the computation complexity of $(y_2,..y_i..,y_n)$ will be $\boldsymbol{O(l*n)}$. Such that the total time complexity will be $\boldsymbol{O(l*m + l*n) = O(l*(m + n))}$.

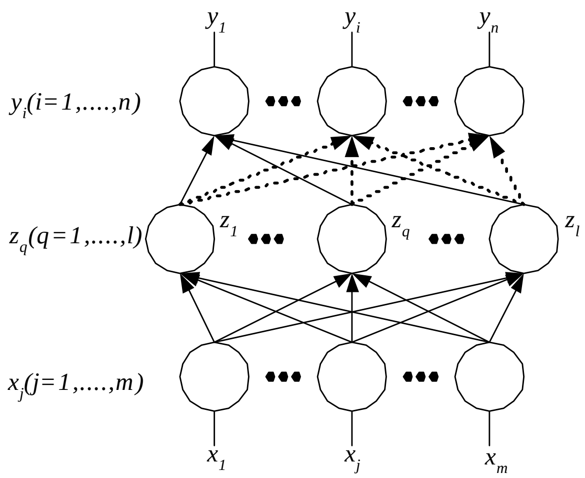


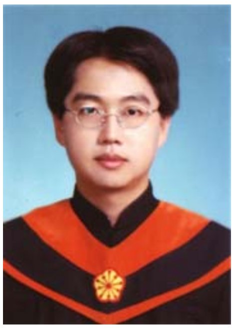
Fig. 19. Three-layer neural network.[16]

$$net_q = \sum_{j=1}^{m} v_{qj} x_j \tag{9}$$

$$z_q = a(net_q) = a\left(\sum_{j=1}^{m} v_{qj} x_j\right) \tag{10}$$

$$net_i = \sum_{q=1}^{l} w_{iq} z_q = \sum_{q=1}^{l} w_{iq} a\left(\sum_{j=1}^{m} v_{qj} x_j\right) \tag{11}$$

$$y_i = a(net_i) = a\left(\sum_{q=1}^{l} w_{iq} z_q\right) = a\left(\sum_{q=1}^{l} w_{iq} a\left(\sum_{j=1}^{m} v_{qj} x_j\right)\right) \tag{12}$$



Shiang-Chun Liou received the B.S. degree in computer science & engineering from Yuan-Ze University in 1995, and the M.S. degree in engineering science from National Cheng-Kung University in 1997, Taiwan. Shiang-Chun Liou is a lecturer in Leader University and he is currently working toward the Ph.D. degree in engineering science at National Cheng-Kung University. His current research interests include mobile communication, multimedia transmission and industrial communication.

Yueh-Min Huang was born in Taiwan, R.O.C., in 1960. He received the B.S. degree in engineering science from National Cheng-Kung University, Tainan, Taiwan, in 1982 and the M.S. and Ph.D. degrees in electrical engineering from the University of Arizona, Tucson, in 1988 and 1991, respectively. He joined the Department of Engineering Science, National Cheng-Kung University, as an Associate Professor in 1991 and became a Professor in 1999. His research interests include distributed multimedia systems, data mining, and real-time systems. Dr. Huang is a Member of the IEEE Computer Society, the Taiwanese Association for Artificial Intelligence, and the Chinese Fuzzy Systems Association. He received the 1996 and 1998 Acer Long-Term Awards for Best M.S. Thesis Supervision.