Institut Eurécom
Department of Mobile Communications
2229, route des Crêtes
B.P. 193
06904 Sophia-Antipolis
FRANCE

Research Report RR-06-171

The Challenges of Predicting Mobility

August 1st, 2006
Last update April 25th, 2007

Jérôme Härri, Christian Bonnet and Fethi Filali

Tel : (+33) 4 93 00 81 00
Fax : (+33) 4 93 00 82 00
Email : {Jerome.Haerri,Christian.Bonnet,Fethi.Filali}@eurecom.fr

[1]Institut Eur´ecom's research is partially supported by its industrial members: BMW Group Research & Technology - BMW Group Company, Bouygues T´el´ecom, Cisco Systems, France T´el´ecom, Hitachi Europe, SFR, Sharp, STMicroelectronics, Swisscom, Thales.

The Challenges of Predicting Mobility

Jérôme Härri, Christian Bonnet and Fethi Filali

Abstract

In this report, we describe the challenges facing mobility prediction in mobile networks and the potential benefit from its usage. We first describe telecommunication areas, which are subject to, and potentially suffering from, the mobility of network terminals. We then provide a related work on prediction models illustrating previous successful attempts aimed at taming mobility in cellular and mobile networks. Finally, we dissert on the potential of the adaptation of such techniques to mobile ad hoc networks by describing some successful application to routing and location management in ad hoc networks.

Index Terms

Mobility and resource management, mobility prediction, prediction models, Kalman filters, auto-regressive models, particle filters, performance, cellular networks, mobile networks, MANET.

Contents

List of Figures

1 Introduction

Since the creation of modern telephony, telecommunication networks have been developed in a static way, the mobility of users being negligeable with respect to the new capacity to connect two remote customers. At the eve of the Internet, we started to see the worldwide generalization of those static networks. However, at the same time, the customers demand for a larger flexibility toward nomadic patterns appeared, which placed mobility management within static networks as an important improvement factor.

In parallel, wireless networks were developed which was the response of telecommunication operators for a growing demand of seamless mobility and wireless connections. Cellular networks such as 1G, 2G or 3G have been designed with a clear objective to offer a national mobile communication coverage, quickly increased to a worldwide coverage with the introduction of roaming capabilities. However, with the generalization, of data traffi c on cellular networks, as well as the increasing demand for improved throughput and security, mobility became a more serious subject.

Mobility is indeed a serious factor contributing to the performance of mobile telecommunication networks. It limits the capacity to maintain a connection, or to guaranty a quality of service between two customers. Moreover, the signifi cant increase of the telecommunication customers dramatically increased the effect of mobility on the network maintenance.

Studies have been produced on the effect of mobility on telecommunication networks. Information Theory showed that mobility is able to increase the network capacity by increasing the network spatial diversity, a feature actually long known by epidemiologists studying virus propagations. However, this improvement comes at the cost of unbounded delays making this improved capacity unusable on real network deployments, and which explains in part why communication protocols are not taking advantage of the increased spatial diversity for communication improvements.

In most technologies used nowadays, networks are subject to terminal mobility. This effect may be compared to a blind person evolving in our universe and trying to discover its own representation with its stick. Our universe is indeed a knowledge plane acquired with experience, while mobile and fi xed network stations are trying to blindly discover this universe using periodic transmission of beacon messages. Some protocols have been designed to reduce this drawback, yet without being able to jump the fence and resolve it.

In this section, we present concrete examples of methods successfully developed for telecommunication networks in order to limit the effect of user mobility. Then, in the second step, we re-introduce[1] a breach in the way modern telecommunication network could be designed by illustrating that mobility may actually

[1] We use the term ”re-introducing”, as this way of thinking had already been proposed in the past for the 2G and 2.5G cellular networks, but later forgotten in analysis of ad hoc and wireless telecommunication networks.

be tamed by predicting instead of being subject to it. For that matter, we describe prediction models that are available, then we illustrate the application fields those models may be used. Our objective is to clearly demonstrate and convince the reader that efficient solutions exist which provides a better vision of the mobility of network terminals.

2 Undergoing Mobility

As described in the previous section, telecommunication networks have always been subject to mobility. A major task for telecommunication engineers is therefore to design technics reducing this drawback. For example, in cellular networks GSM/UMTS, mobility management is handled differently whether the mobile terminal in active communication or not.

For example, the only way to contact a mobile terminal in idle mode is by paging it. In order to save network resource when the mobile terminal is not connected, the base station only keeps a coarse vision of the zone where the mobile terminal is. Accordingly, if the base station does not have any precise information on where a mobile terminal is, it needs to page the whole network with all the latency incurred by this method. Therefore, in order to reduce the drawbacks of this approach, GSM/UMTS systems developed a hierarchical structure called *Paging Area (PA)* including several *Location Areas* for the GSM or *Routing Areas* for the UMTS. Thanks to this, the system limits its paging's scope to the PA containing the last LA/RA where the mobile terminal has last been attached. By using this process, the system is able to save network resources and delay.

Instead, when the mobile terminal is actively communicating, the base station needs to keep a very precise vision on the region where the terminal is located in order to reserve network resources for future cell handovers. This procedure is critical for the seamless functionality of cellular networks as no drop calls should occur resulting from handovers. For that objective, the mobile terminal periodically samples then transmits RSSIs of all base station beacons it receives to its connected base station in order to obtain a coarse relative position estimates. Then, the mobile terminal and the attached base station may coordinate with the next base station in order to anticipate the handover and reserve the required network resources.

In the IP world, provisions have also been created in order to deal with mobility or nomadism of IP terminals. The IPv4 and IPv6 networks developed algorithms, called *Fast Handover*, limiting the packet losses generated by changes of covering zones. Alerts are triggered when a node is approaching a new access router, which creates alternate routes faster and re-routes packets even before the real hand-over actually takes place. However, this system is resource consuming as it requires a periodic tracking of access routers.

Ad hoc networks also had to quickly develop efficient methods to handle terminals mobility. Globally, five different categories of protocols were designed:

- **Proactive Protocols–** Similarly to static networks, those protocols build routing tables providing a path to any accessible destination on the network. Periodic beacon messages are triggered in order to adapt the backbone to topology changes at the cost of a higher energy consumption and channel occupancy. The two flagships in proactive routing protocols are the *Wireless Open Shortest Path First (W-OSPF) [1,2]* and the *Optimized Link State Routing (OLSR) [3,4]*. Indeed, after having developed many protocols, the community slowly started to converge to those two protocols, which are also the only two candidate to the IETF standard track RFC for proactive routing in Mobile Ad Hoc Networks (MANETs). Yet, the more the mobility increases, the harder it becomes to maintain the routing tables. Accordingly, this approach has shown not to be very adapted to fast mobile networks. Recent results also pointed out the relationship between performance and density, arguing that proactive routing could only be efficient on dense networks.

- **Reactive Protocols–** In order to limit the waste of resources, reactive networks only open routes on demand. Thanks to this limitation, the mobility of nodes not involved in the opened route does not influence network management. However, the mobility of nodes belonging to the opened route reduces the performance of reactive networks. Local repairs are possible in the case of a route failure and, in order to reduce the latency of a broken path, reactive networks also use periodic beacon messages. In this category, the *Direct Source Routing (DSR) [5]* protocol and the *Ad hoc On Demand Distance Vector Routing (AODV) [6]* are two potential candidate, although that the IETF recently chose a modified and improved version of AODV called *Dynamic MANET On-demand (DYMO) [7]* as the only candidate to IETF standard track RFC for reactive routing in MANETs.

- **Geographic Routing–** It is a stateless approach where no backbone or route is generated. Instead, geographic information of the destination and intermediate nodes are used in order to wisely choose the best candidate to forward a packet toward the intended destination. Those protocols are based on two functions: *the greedy forwarding* and the *recovery*. Indeed, each node receiving a packet will try to chose the best candidate among its neighbors with the maximum progress toward the destination node. This is the *greedy forwarding* phase and *Most Forward within Radius [8]* is the technique most widely used in order to find the best progress. But in some cases, the packet falls in some local maxima, where not any single node in the neighborhood may bring any potential progress toward the destination. Accordingly, a recovery phase is triggered, where the packet is sent back until an alternate candidate is found. This is the *recovery phase* and use mostly *Face Routing [9] pp. 389-394* to circumvent the local maxima. The first and still pioneer protocol in this field is the *Greedy Perimeter Stateless Routing (GPSR) [10]* protocol, but some extension and improvements in the two phases have been suggested ([9] Sect. 12.4). Nodes mobility still alters the precision of geo-localization information, potentially reducing the performance of the geographic forwarding approach. The strong requirement of the availability of a geo-localization system was the major justification for the IETF for not pushing this approach for standardization. Yet, the stateless feature of geographic routing made them good candidates for routing in Vehicular Ad Hoc Networks, where GPS systems are commonly accepted.

- **Fish-Eye Routing–** In order to deal with the lack of precision of geographic information, mobility is handled in a different way whether the destination node is far or close from the intermediate or sender node.

  – *Locally*: Frequent position updates of all neighboring nodes are triggers as mobility has a significant local influence.

  – *Remote*: Only a coarse mobility maintenance is triggered as the remote mobility does not have a significant influence on a local decision.

  The *Fisheye State Routing (FSR) [11]* protocol or the *Landmark Routing (LANMAR) [11]*, two proactive approach, are two example, where a node keeps up to date state information about all nodes in its inner circle (or landmark), while the accuracy of such information decreases as the distance increases. Even if a node does not have an accurate state information about distant nodes, packets will be routed correctly because the route information becomes more and more accurate as the packet gets closer to the destination. Another proactive protocol in this category is called *Distance Routing Effect Algorithm for Mobility (DREAM)*. It is based on *location information*, and adapt its location update to both mobility rate and distance. Finally, a reactive approach called *Location-Aided Routing (LAR) [11]*, also based on *location information*, has been developed, where each node maintains the location about nodes it is aware of with respect of the distance. The farer is the node, the larger is area and then, on demand, orients route requests toward the area where the destination node is.

- **Hybrid Routing–** This is the last category of protocols which mixes the proactive approach for local routing and reactive even geographic approach for distance routing. Most of the protocols developed in this category either create local zones, clusters, or trees and uses a reactive routing strategy to route between them. The *Zone Routing Protocol [12]*, or the Hybrid Ad Hoc Routing Protocol [13] are examples of this approach.

Although some techniques have been developed to reduce the impact of nodes mobility, it still has a major impact on the performance of routing protocols. And similarly to the topology management approaches previously described, all are subject to mobility and non negligeable resources are dedicated to maintaining the stability of the network backbone or routes with respect to mobility. These resources could be better used if the mobility could be used as an asset instead of a drawback.

3 Predicting Mobility

An alternative to the methods described in the previous section is to try to predict users mobility. Indeed, by again considering the example of blind persons, what differentiate us from them is first our global long range vision, and second our capacity to predict and anticipate the evolution of our environment. Similarly, mobility prediction techniques could be used in order to improve the management of mobile networks.

Definition 1 (Mobility Prediction) *Capacity to evaluate a future position given past positions.*

Mobility Prediction is actually a very ancient technique used by the first sailors to navigate on seas and oceans. In marine literature, this technique is better known as **Dead Reckoning**. Using instruments measuring:

- the initial point
- the azimuth, or headings (Astrolab, Sextant, Compas)
- the speed (Chip log, Tachometer, Anemometer, Doppler sonar)
- the time (Astrolab, Chronograph)

the dead-reckoning technique is able to obtain the current position and the distance travelled since the last known position. Inertial system are able to improve the precision of dead-reckoning techniques for systems that are not able to receive satellite signals. Nowadays, a large variety of navigational methods are still based on dead-reckoning, varying from under-water navigation, spatial navigation, missile guidance and tracking. More generally, in any domain where a knowledge of the trajectory taken by a system is vital, mobility prediction is used.

In telecommunication network management, resources are shared in order to benefit to the widest set of users. And those resources are allocated depending on the density of users. Yet, mobility makes this management random and inefficient. The knowledge of the trajectory taken by users may be very useful in order to improve the resource management of mobile telecommunication networks. This is also a significant motivation for the study of mobility prediction techniques in the field of telecommunication networks.

3.1 Available Localization Techniques

On periodic position reassessments, mobile terminals using algorithms based on mobility prediction techniques must acquire their position. It is therefore necessary to obtain a sporadic access to a geo-localization system. Three categories of geo-localization algorithms exists:

- Satellite Systems (GPS or Galileo): It is a widely diffused system, which guaranties a precise localization (± 1m) at a low cost. However, the acquisition time may be long (>30s), it also consumes a non negligeable energy, and requires access to satellite signals.
- Beacon Systems (GSM): The precursor of GPS localization, and an alternative to situation when the GPS signal is not available. However, the precision cannot challenge the GPS system.
- Hybrid Systems:
  - Inertial Systems: Contain a set of accelerometers, gyroscopes and guidance algorithms able to provide the velocity, orientation, and angular velocity of a mobile system by measuring the linear and angular accelerations applied to the system in an inertial reference frame. If calibrated on known positions and velocities, the inertial system is then able to estimate a mobile system complete trajectory.
  - GPS Systems: Compute position, velocity, acceleration, and use complex mobility prediction techniques when the signal is not available.
  - GSM Systems: Uses all available techniques or triangulation or multilateration in cellular networks: Angle of Arrival (AOA), Time Difference of Arrival (TDOA), Enhanced Cell Identifi cation (E-CID), Uplink Time Difference of Arrival (E-TDOA), Enhanced Observed Time Difference (E-OTD), or A-GPS.

3.2 Mobility Prediction Models

Mobility prediction models has nothing new, and working on this fi eld could look like trying to reinvent the wheel. Indeed, they have initially been developed for tracking purposes in the 60s in cellular Networks since 1995. More complex models have later been used in order to be applied to cellular systems requiring a quality of service such as the Wireless ATM network. The complexity and the precision of those models culminated around the year 2001, but unfortunately were forgotten afterward as illustrated in Fig. 1[2]. In fact, it took a long time to the mobile ad hoc network community to understand that mobility prediction was as important as it used to be for cellular network. But then, only simplistic models were re-introduced, as if the whole past literature has either been forgotten, or judged too complex for the needs. However, we started to see a growing popularity in complex iterative models in recent work and expect this popularity to further increase in the near future. This section aims at recalling and putting back into the light the different orientations taken by the community in order to tame the mobility of mobile terminals.

[2]This fi gure has been obtained based on the number of google hits using keywords *Trajectory, Prediction, Tracking, Mobility* and estimated based on the complexity of each solutions and the initial publication year.

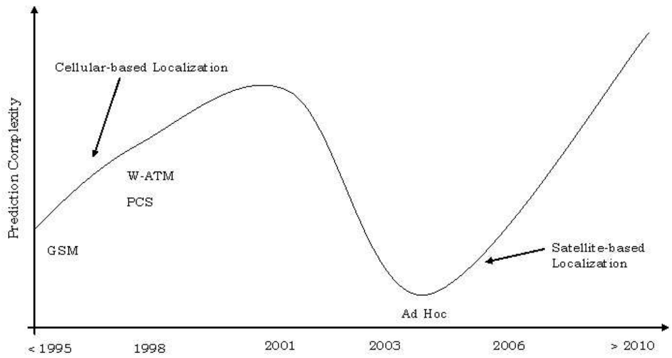


Figure 1: Evolution of the Popularity of Prediction Techniques

3.2.1 Deterministic Models

Deterministic mobility prediction models may be a first order model, only considering the position and a fixed velocity, but more higher order models have also been designed, including acceleration and a time-varying velocity.

The mostly known and used deterministic model is the first order kinetic model illustrated in Fig. 2(a).

$$\overrightarrow{Pos}(i+1) = \begin{pmatrix} X_{i+1} \\ Y_{i+1} \end{pmatrix} = \begin{pmatrix} X_i \\ Y_i \end{pmatrix} + \begin{pmatrix} V_x^i \\ V_y^i \end{pmatrix} \cdot (t - t_i) \qquad (1)$$

A direct application of (1) is to compute the kinetic distance or the estimated connection time between two nodes P and Q. The kinetic distance is computed as follows

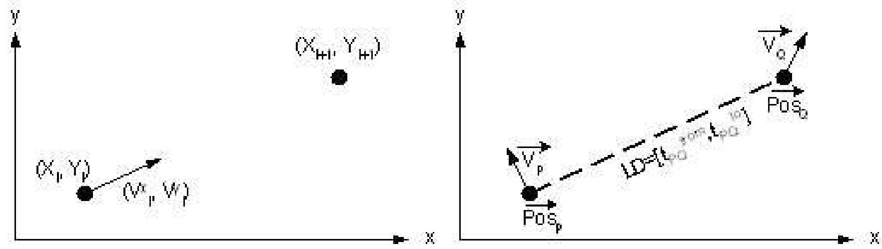
$$\begin{aligned} D_{PQ}^2(t) &= D_{QP}^2(t) = \|\overrightarrow{Pos}_Q(t) - \overrightarrow{Pos}_P(t)\|_2^2 \\ &= \left[\begin{pmatrix} X_Q - X_P \\ Y_Q - Y_P \end{pmatrix} + \begin{pmatrix} V_Q^x - V_P^x \\ V_Q^y - V_P^y \end{pmatrix} \cdot t \right]^2 \\ &= a_{PQ}t^2 + b_{PQ}t + c_{PQ}, \end{aligned} \qquad (2)$$

Considering r as nodes maximum transmission range, as long as $D_{PQ}^2(t) \leq r^2$, nodes P and Q are neighbors. Therefore, solving

$$\begin{aligned} D_{PQ}^2(t) - r^2 &= 0 \\ a_{PQ}t^2 + b_{PQ}t + c_{PQ} - r^2 &= 0, \end{aligned} \qquad (3)$$

gives t_{PQ}^{from} and t_{PQ}^{to} as the time intervals during which nodes i and j remain neighbors (see Fig 2(b)).

In cases where the velocity is not constant, a second order prediction model based on the Euler motion law is used.



(a) First Order Prediction Model

(b) Link Duration (LD) between nodes P and Q

Figure 2: First Order Prediction Model and its Application to Link Duration

$$\begin{aligned} \overrightarrow{V}_{i+1} &= \overrightarrow{a_i} \cdot t + \overrightarrow{V}_i && \text{(4a)} \\ \overrightarrow{Pos}_{i+1} &= \frac{1}{2}\overrightarrow{a_i} \cdot t^2 + \overrightarrow{V}_i \cdot t + \overrightarrow{Pos}_i && \text{(4b)} \end{aligned}$$

Although vehicular motions involve impulsive forces (such as sudden braking), a constant acceleration is usually accepted in high speed mobility networks. However, a piecewise constant acceleration is used in practice. In both cases, (4) may be used to predict a future position based on some kinetic information. (4) may be solved by substitution.

$$\begin{aligned} \overrightarrow{Pos}_{i+1} &= \frac{1}{2} \cdot \frac{\left(\overrightarrow{V}_{i+1} - \overrightarrow{V}_i\right)}{t} \cdot t^2 + \overrightarrow{V}_i \cdot t + \overrightarrow{Pos}_i \\ &= \frac{\left(\overrightarrow{V}_{i+1} + \overrightarrow{V}_i\right)}{2} \cdot t + \overrightarrow{Pos}_i && \text{(5a)} \end{aligned}$$

Accordingly, position predictions are calculated using a velocity one step ahead, which forces us to have two samples of past velocities and two piecewise constant accelerations in order to predict the future position.

$$\begin{aligned} \overrightarrow{V}_i &= \overrightarrow{V}_{i-1} + \overrightarrow{a}^{(i-1)\to i} \cdot (t_i - t_{i-1}) && \text{(6a)} \\ \overrightarrow{V}_{i+1} &= \overrightarrow{V}_i + \overrightarrow{a}^{i\to(i+1)} \cdot (t_{i+1} - t_i) && \text{(6b)} \\ \overrightarrow{Pos}_{i+1} &= \frac{\left(\overrightarrow{V}_{i+1} + \overrightarrow{V}_i\right)}{2} \cdot (t_{i+1} - t_i) + \overrightarrow{Pos}_i && \text{(6c)} \end{aligned}$$

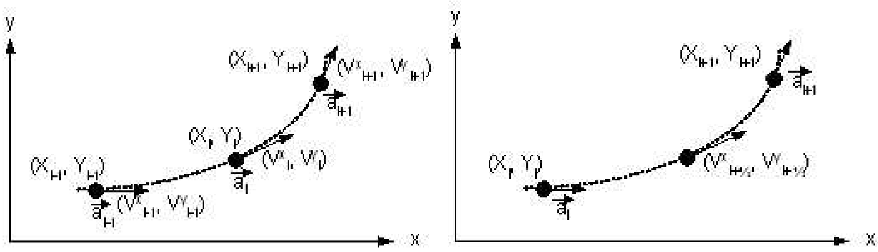
where $\overrightarrow{a}^{(i-1)\to i}$ and $\overrightarrow{a}^{i\to(i+1)}$ are the constant acceleration during the time intervals $[t_{i-1}, t_i]$ and $[t_i, t_{i+1}]$ respectively.

If the acceleration is constant between two sampling intervals, the velocity increases linearly with time and the approximation $\frac{\|\overrightarrow{V}_{i+1}\|+\|\overrightarrow{V}_i\|}{2} \approx \| \overrightarrow{V}_{i+\frac{1}{2}} \|$ is exact. If not, we need to sample the velocity at the mid-interval and use a variation from 4 called the *Feynman-Verlet* model. The *leap-frog* algorithm may be appropriately used in this case.

$$\overrightarrow{V}_{i+\frac{1}{2}} = \overrightarrow{a}_i \cdot t_i + \overrightarrow{V}_{i-\frac{1}{2}} \quad (7a)$$
$$\overrightarrow{Pos}_{i+1} = \overrightarrow{V}_{i+\frac{1}{2}} \cdot t + \overrightarrow{Pos}_i \quad (7b)$$

Changes in position are calculated using a velocity that is half a step ahead in time. Likewise, changes in velocity are calculated using an acceleration which is half a step ahead in time. Position and acceleration are therefore in-phase, while velocity is out of phase with position and acceleration.

The *leap-frog* algorithm owes its simplicity to the fact that stepping the velocity half step out of phase with the position and acceleration provides midpoint values for both (7a) and (7b) and thus provides more accurate results than the Euler model. Fig. 3 illustrates both approaches.



(a) Euler Model   (b) Feynman Model

Figure 3: Second Order Constant Acceleration Prediction Models

3.2.2 Stochastic Models

Stochastic models do not aim at obtaining an exact prediction, but rather a correct one with high probability. Stochastic models may be easily used to add an uncertainty to deterministic predictions. But a more important use is to model unknown parameters in the state equations or to take into account the model's prediction error. In many cases, it is both. For example, tracking-based auto-regressive processes (AR) use white noise to model the AR prediction errors, and the estimation of the states, such as position or velocity, is often accomplished using Kalman Filters. Even if position or velocity are obtained without error, the AR process still provides predictions with some errors. Now, if errors are added to the positions or velocities, the performance may decrease drastically. Accordingly, in most applications, joint optimization is applied to obtain good predictions. In the rest of the section, we provides examples and related work using stochastic prediction models.

Approaches for mobility tracking mostly rely on *Autoregressive processes* [14, 15], *Kalman Filtering* [16–19], *semi-Hidden Markov* [20–22] models, or *Particle Filtering* [23–26]. Two measurements have been mostly used in the literature, the Received Signal Strength Indicator (RSSI) or the Time or Arrival (TOA), but GPS positioning is experiencing a growing interest from the community as a mean to reduce the measurement error.

The first and most straightforward model is to weight a deterministic prediction by the probability the prediction still exists. It is defined as follows

$$Pred^{stoch}(t) = Pred^{det}(t) \cdot e^{-\beta(t-t_{sample})} \tag{8}$$

where

$Pred^{det}(t)$	:	Deterministic Mobility Prediction at time t
$e^{-\beta(t-t_{sample})}$	:	Stochastic validity of the prediction parameters
β	:	Stability of the mobility parameters (also called Predicability)
t_{sample}	:	Latest sampling time of the mobility parameters

As mentioned in the beginning of this section, *Autoregressive (AR) Models* also falls in the stochastic class. A white Gaussian noise with zero mean ϵ is used to model the AR prediction error, and the mobility state of the process is estimated using Kalman or Particle Filters.

An auto-regressive model of order p defines the n^{th} value as a weighted sum of the p previously measured ones and is defined as

$$x_n = \alpha_0 + \sum_{i=1}^{p} \alpha_i x_{n-p} + \epsilon_n \tag{9}$$

where ϵ_n is an independent identically distributed noise with zero mean.

Creixell and Sezaki [14] proposed to model pedestrian trajectories using first order auto-regressive process (AR(1)) for the velocity and the azimuth. They used Least Square Lattice filters (LSL) to solve their model and obtained fairly good predictions up to 10 simulation step ahead.

Zaidi and Mark [15] also used a first order auto-regressive model (AR(1)) but used Yule-Walker formulation in order to estimate α and error coefficients of the AR(1) process. Unlike [14], the mobility state are not obtained from mobility traces, but are measured using RSSI (Received Signal Strength Indicator) or TOA (Time of Arrival) and then approximated using Kalman Filters. They validated their approach by comparing it against real sets of data.

Another model is called the *Gauss-Markov* prediction model and has been proposed in [27]. It first models a node's velocity as a time-correlated Gauss-Markov random process. In discrete time, it computes the predicted velocity based on the previous value and a Gaussien iid process

$$v(n) = \alpha v_{n-1} + (1-\alpha)\mu + \sigma\sqrt{1-\alpha^2} w_{n-1} \tag{10}$$

with the Gauss-markovien auto-correlation process

$$R_v(\tau) = E\left[v(t)v(t+\tau)\right] = \sigma e^{-\beta|\tau|} + \mu^2 \tag{11}$$

where

$$\begin{array}{rcl} \alpha & = & e^{-\beta|\tau|} \\ \beta & : & \text{Memory size} \\ \sigma^2 & : & \text{Variance of the v(t) process} \\ \mu & : & \text{Expectation of the v(t) process} \\ w_n & : & \text{Gaussien IID Process} \end{array}$$

Their numerical results have demonstrated the importance of the performance gain of prediction-based approaches, but also confirmed that the performance of such approach is directly proportional to the predictability of a node's mobility pattern. That was also our intuition and was the justification of our predictability analysis in [28].

While deterministic models are able to model quite fairly first order or second order kinetic models with constant accelerations, a velocity subject to an unknown acceleration, or known but non-constant acceleration requires the use of more complex stochastic models. Many studies of position tracking in wireless networks exist, most of them deploying some form of Kalman filtering to the position tracking problem. However, in recent years, it has been noted that the sequential Monte Carlo processing filters, better known as the Particle Filter, can provide an improved performance in the non-linear and non-Gaussian noise tracking problem [29].

Central to all navigation and tracking applications is the motion model to which various kind of model based on filters can be applied. Models that are linear in the state dynamics and non linear in the measurements are often considered:

$$x_{t+1} = Ax_t + B_u u_t + B_f f_t \tag{12a}$$
$$y_t = h(x_t) + e_t \tag{12b}$$

where

$$\begin{array}{rcl} x_t & : & \text{state vector} \\ u_t & : & \text{measured input} \\ f_t & : & \text{unmeasurable input or faults toward the measured input} \\ y_t & : & \text{measurement} \\ e_t & : & \text{measurement error} \end{array}$$

An independent distributions may usually be assumed for f_t, e_t, x_0 with known probability densities p_{e_t}, p_{f_t}, and p_{x_0}, respectively, not necessarily Gaussian. The difference between the applications based on (12) mainly lies in the different means to obtain the measurement equation (12b).

Liu and al. [16] proposed a mobility model for wireless ATM networks based on a dynamic linear system model in wich the mobility state consists of the position, velocity and acceleration of the mobile terminal. Originally introduced by Singer [30], the system can capture a wide range of realistic user mobility patterns. The measurement is based on an estimated position obtained by the RSSI (received signal strength indicator) from three different base stations, and the state vector is given by

$$\dot{x}(t) = Ax(t) + Bu(t) + Cr(t) \tag{13}$$

where

$$A = \begin{bmatrix} \Theta & 0 \\ 0 & \Theta \end{bmatrix} \qquad B = C = \begin{bmatrix} \Phi & 0 \\ 0 & \Phi \end{bmatrix}$$
$$\Theta = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \qquad \Phi = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$$

and where

$$\begin{aligned} x(t) = \left[X(t)\dot{X}(t)Y(t)\dot{Y}(t)\right]^T &\quad : \quad \text{Node } u \text{ mobility vector} \\ u(t) = [u_x(t)u_y(t)]^T &\quad : \quad \text{Node } u \text{ deterministic acceleration command} \\ r(t) = [r_x(t)r_y(t)]^T &\quad : \quad \text{Node } u \text{ random acceleration} \end{aligned}$$

The structure of the model, illustrated in Fig. 4, manages to replace a time varying acceleration with a semi-Markov based acceleration commands and a random acceleration component. This filter is resolved using Kalman Filtering techniques.

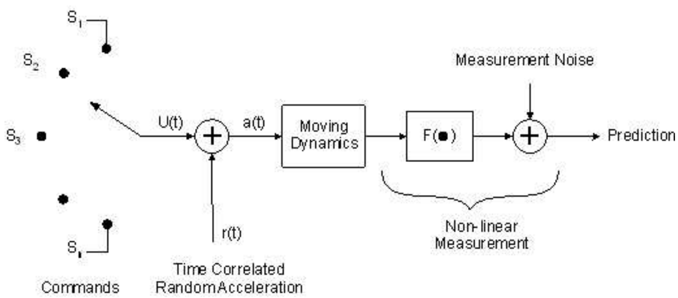


Figure 4: Structure of the Stochastic Micro-Prediction Model

A key observation is that the process u_n is a semi-Markov process. Therefore, accurate estimation of u_n should exploit its semi-Markov characterization. Yu and Kobayashi [20, 21] developed an efficient algorithm for estimating the parameters of a hidden semi-Markov Model (HSMM), a generalization of a similar approach for Hidden Markov Models (HMM), and its application to position tracking. Zaidi and Mark [22] also used the same idea to obtain the acceleration command u_n with a HSMM estimator while using a Kalman Filter to estimate the mobility states. Thanks to this hybrid approach, their solution outperformed Liu's work [16] in terms of prediction errors by a factor of 5.

Zaidi et al. [19] later generalized their approach and proposed to first preprocess the RSSI with an average Filter to obtain coarse position estimates, and second to decouple the mobility state estimates $x(t)$ from the estimation of the discrete command process $u(t)$. They illustrated how their approach was able to follow mobile trajectories more accurately than in Liu's work.

Pathirana et al. [17] proposed a modification to Liu's work and used a Robust Extended Kalman Filter (REKF) approach in order to improve the prediction accuracy, processing efficiency, and more important, to include non-linearities to the model. Accordingly, no assumption are made on the measurement equation or the system dynamics, and thus could be able to better model sharp turns and log-normal or Nakagami propagation models popular in the modeling of Vehicular Network.

Another mean to solve the motion model described in (12), without using REKF or more complex systems, is by means of Bayesian recursive filtering, also called Particle Filtering. The optimal Bayesian Filter in the case of (12) is given below and is composed of a prediction step and an update step. If the set of available observations at time t is given by

$$Y_t = \{y_0, \cdots, y_t\},$$

then the Bayesian solution to compute the posterior distribution $p\left(x_{t+1}|Y_t\right)$ of the state vector, given past observations, is given by

$$\begin{aligned}
p\left(x_{t+1}|Y_t\right) &= \int p\left(x_{t+1}|x_t, Y_t\right) p\left(x_t|Y_t\right) && \text{(16a)}\\
&= \int p\left(x_{t+1}|x_t\right) p\left(x_t|Y_t\right) dx_t && \text{(16b)}\\
&= \int p_{f_t}\left(B_f^{\dagger}(x_{t+1} - Ax_t - B_u u_t)\right) p\left(x_t|Y_t\right) dx_t && \text{(16c)}\\
p\left(x_t|Y_t\right) &= \frac{p\left(y_t|x_t\right) p\left(x_t|Y_{t-1}\right)}{p\left(y_t|Y_{t-1}\right)} && \text{(16d)}
\end{aligned}$$

where we assume that both the initial probability density of state p_0, and the density $p\left(x_t|Y_t\right)$ at time step t are known and $p\left(y_t|Y_{t-1}\right) \approx c_t$.

In the case the motion model is as 12a and the update equation is as 12b, (16) may be rewritten

$$p\left(x_{t+1}|Y_t\right) = \int p_{f_t}\left(B_f^{\dagger}(x_{t+1} - Ax_t - B_u u_t)\right) p\left(x_t|Y_t\right) dx_t \quad (17a)$$

$$p\left(x_t|Y_t\right) = \frac{p_{e_t}\left(y_t - h(x_t)\right) p\left(x_t|Y_{t-1}\right)}{c_t} \quad (17b)$$

The particle filter can be considered as an approximation to a sequential solution to the above equations. It achieves this by representing the posterior density with some random weighted samples, called the *Particles*. A typical particle filter algorithm consists of 5 steps that we shortly describe next.

1. *Initialization*: Generate $x_0^i \sim p_{x_0}$, $i = 1, \cdots, N$. Each sample of the state vector is referred to as a *particle*.

2. *Measurement Update*: At each particle position, the assigned weight of each particles is updated and normalized according to a likelihood function (based for example on RSSI cumulative distribution).

$$w_t^i = w_{t-1}^1 p(y_t|x_t^i)$$

$$w_i = \frac{w_i}{\sum_i w_t^i}$$

   This is the *update step* of the Bayesian recursive filtering.

3. *Resampling*: P particles are replaced from the set of particles based on the weights. This step is necessary in order to avoid a high concentration of probability mass at a few particles.

4. *Prediction* Move the particle forward according to the adopted Model (12a for instance). This step is therefore the *prediction step* of the Bayesian recursive filtering. The particles are now referred to as *predicted particles*.

5. Let $t := t = 1$ and iterate item 2).

There is a large literature of successful use of Particle Filtering methods to solve positioning, tracking or navigation, and it is hard to be exhaustive. The major difference between different approaches are usually the *measurement step* or the *Resampling Phase*.

For example, Yang and Wang [23] also illustrated the inaccuracy of Liu's work and proposed an alternative estimation scheme based on a sequential Monte Carlo (SMC) Filtering. The SMC can achieve better performance than the Liu's filtering, but is computationally intensive and hence might not be suited for real-time trajectory predictions. Zaidi et al. [19] later showed that the SMC was outperformed by their Modified Kalman Filtering approach.

Gustafsson et al. [24] proposed a framework for positioning, navigation and tracking problems using particle filters. They showed a clear improvement in performance in real-time, off-line, on real data and in simulation environments compared with existing Kalman filter-based solutions in term of convergence time and precision. By using Rao-Blackwellization, authors also managed to reduce the increased computational complexity of the Particle filter approach compared to Kalman Filters.

Mihaylova et al. [25] two other Sequential Monte Carlo algorithms, a Particle Filter and a Rao-Blackwellised Particle Filter, have also been presented. In contrast to previous work [16, 19, 23], the mobility tracking is formulated as an estimation problem of a *hybrid system*, where a base state vector is continuously evolving, and a mode state vector which may undergo abrupt changes. This formulation together with the Monte Carlo approach showed it could reduce the computational complexity and provide efficient mobility tracking.

Sha et al. [26] described another Particle Filtering approach for position tracking in Wi-Fi networks under the assumption of log-normal fading and with intermittent GPS information signaling. They obtained a factor 2 improvement against a stand alone Wi-Fi-based localization and could obtain real-time positioning in hybrid Wi-Fi and GPS systems.

3.2.3 History-based Models

Those models are usually used to predict the terminal macro-mobility, or the cell to cell mobility. Indeed, repetition of routine movements allows to more easily learn the users preferred paths.

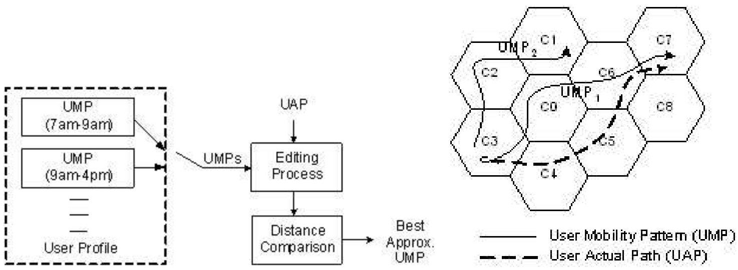
One method to characterize users mobility regularities is to record a set of *User Mobility Patterns* stored in a *profile* for each user and indexed by the occurrence time (see Fig. 5(a)). The major difficulty is to assess the sensitivity between the UMP and the *User Actual Path (UAP)*. Indeed, is a UAP which diverges from the UMP by a single cell a small variation of the same path, or a totally new path not reported in the profile ?

Different approaches have been proposed [16, 31–33] in the past. We illustrate here the solution presented in [16], where the authors successfully used approximate pattern-matching techniques to find the UMP that fits best to a UAP. For example, if a UMP is described by a cell sequence $(a_1 a_2 \cdots a_{i-1} a_i a_{i+1} \cdots a_n)$, then the authors modelled the regular movement of a mobile user as an edited UMP by allowing the following legal options:

- *inserting* a cell c at position i of the UMP gives UAP: $(a_1 a_2, \cdots a_{i-1} c a_i a_{i+1} \cdots a_n)$
- *deleting* the cell a_i at position i of the UMP gives UAP: $(a_1 a_2, \cdots a_{i-1} a_{i+1} \cdots a_n)$
- *changing* a cell a_i to another cell c gives UAP: $(a_1 a_2, \cdots a_{i-1} c a_{i+1} \cdots a_n)$

Figure 5(b) gives an example a UMP $c_3 c_0 c_6 c_7$ and its edited UAP $c_3 c_4 c_5 c_6 c_7$, which can be obtained by *changing* c_0 to c_4 and *inserting* c_5.

The degree of resemblance of a UAP with a UMP is measured by the *edit distance* a finite string comparison metric. The simplest way to find this distance is by determining the smallest number of *insertion*, *deletion* and *changes* by which two cell sequences can be made alike. If the *edit distance* is less than a matching threshold t, an approximately matched UMP is found, indicating the general moving intention of the user and the macro-prediction may be done accordingly. For example, in Fig. 5(b), the UMP_1 has clearly a smaller *edit distance* than UMP_2 compared to the UAP. UMP_1 is therefore selected as general moving intention of the user.



(a) Macro-model

(b) Example with 2 Potential UMPs

Figure 5: Global Mobility Model

In the next example, another way to benefit from mobility patterns repetition is by modeling by them by sequence of stationary events generated using a *Markovien process of order m*. In other words, the new event may be generated as a function of the *m* previous events.

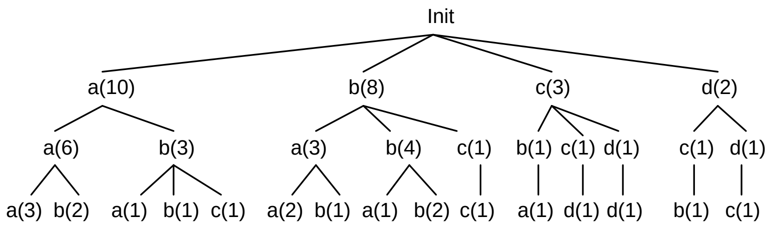
$$P\left[V_{l+m+1} = v_{l+m+1} | V_1 = v_1 \cdots V_l = v_l \cdots V_{l+m} = v_{l+m}\right] \tag{18a}$$
$$= P\left[V_{l+m+1} = v_{l+m+1} | V_l = v_l \cdots V_{l+m} = v_{l+m}\right] \tag{18b}$$

where V_i are the states of the system, which may be represented by a cell, or an occupied road segment.

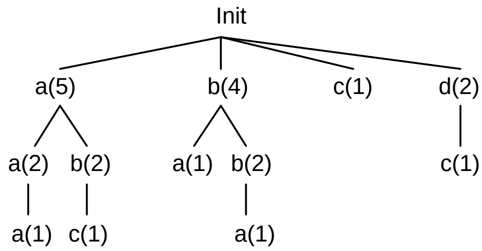
A representation of (18) may be obtained by a *trie* or a *digital search tree*, where every node represents a context $V_k = v_k | V_{k-1} = v_{k-1} \cdots V_1 = v_1$ and stores its last symbol along with the relative frequency of its appearance at the context of the parent nodes. Obviously, the depth of the trie is the order of the Markovien process and, as we move down the trie, we restrict our uncertainty to finally converge to a next event prediction when we reach a leaf. The performance of the prediction is therefore the trie's ability to add a new event to the frequency of an already existing node (reducing the uncertainty) and not to create a new branch (an unpredicted event).

The *Uncertainty* of a new event based on a sequence of past events is called the *Entropy* in Information Theory, and the optimal prediction of the future state may then be obtained from algorithms minimizing this entropy. The Lempel-Zif (LZ78) algorithm is a good choice in order to generate an optimal dictionary of observed paths and a reduced search trie.

Figure 6 illustrates an example of the trie representation of a movement history $aaababb\ bbbaabccddcbaaaa$ with a second order Markov Process and its improvement using the LZ78 algorithm. This algorithm creates the dictionary a, aa, b, ab, bb, bba, abc, c, d, dc, ba, aaa and only adds a new branch by concatenating a new entry v with a symbol w already contained in it.



(a) Classical Trie



(b) Lempel-Zif Trie

Figure 6: Trie Representation of a Movement History modeled by a Markovie Process of order 2

The idea of using the LZ78 algorithm in order to reduce the uncertainty has been originally presented by Bhattacharya and Das [34] for Location Management under the name *LeZi-Update*. An extension to Handover prediction in Wireless Networks, which has been introduced in [35], is presented next.

By representing the state sequences as $N, H_1, H_2, \cdots H_n, E$, where N is a new call, H_i is the i^{th} handoff, and E is the end of call, we can generate the prediction tree illustrated in Fig. 7. Each sequence of events $N, H_1, H_2, \cdots H_n, E$ during the lifetime of a call corresponds to a substring in the Ziv-Lempel algorithm.

Each node builds a tree based on the sequence of events. For example, in Fig. 7, the Lempel-Zif algorithm found 3 substrings which ended at time slot 2, 3 substrings which contained a handover to cell b also at time slot 2, and where all of them ended right after the handover, or 15 substrings which contained a handover at time slot 1 etc...

When the mobile requests a new call in cell a in the time interval 9:00-9:01 a.m., we can use the statistics preserved in the node's mobility trie to predict the probabilities of the next possible events of this mobile. From the root's point of view, it will terminate the call without handoffs in the 2nd time slot with probability of $3/56$, handoff to cell b in the 2nd time slot with probability of $2/56$. Then, depending on the next event, we go down the tree following the sequence of events in order to refine the predictions. If one prediction error occurs, the tree is updated.

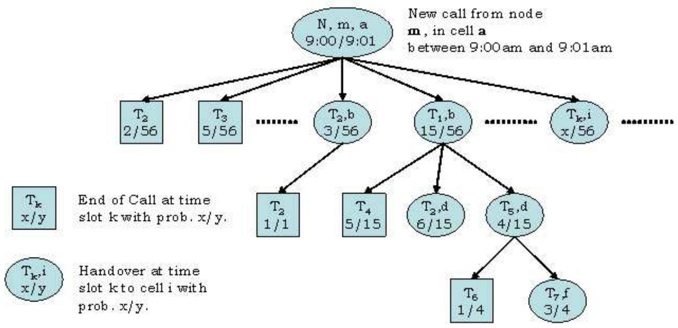


Figure 7: Example of a Lempel Zif tree predicting the time slot of either the end of call or a handoff

This kind of repetition also allows to successfully use fuzzy logic algorithms. In the following example, authors in [36] used a Neuro-Fuzzy Inference Model (NFIS), which is based on an `IF`, `THEN}`rule whose consequence is a real number.This model provides the inference structure that avoids the time-consuming process of defuzzification in an inference procedure. The form of fuzzy `IFTHEN` rules is as follows:

$$Rule\ i: \quad IF\ x_1 = A_1^i\ and\ \cdots x_j = A_j^i,\ \ THEN\ y = \omega_i \tag{19}$$

where

$x_1 \cdots x_d$: input variable

A_j^i : A fuzzy set for input variable x_j in the i^{th} fuzzy rule

ω_j : A real number for output variable in the i^{th} fuzzy rule

Given the real-value input vector $\vec{x} = [x_1, x_2 \cdots x_d]$, the real-value output of the fuzzy model is inferred as follows:

$$NFIS = f\left(\vec{x} = [x_1, x_2 \cdots x_d]\right) = \frac{\sum_{i=0}^{n} \mu_i \omega_i}{\sum_{i=0}^{n} \mu_i} \tag{20}$$

where

$\mu_i = \prod_{j=1}^{d} \mu_{A_j^i}(x_j)$: Fuzzy membership function of the Fuzzy set A_j^i

$\mu_{A_j^i}$: Fuzzification function (triangular, trapezoid, Gaussian)

Then, the prediction is as follows:

$$NFIS\left(S_k^t, S_k^t, \cdots S_k^{t-(d-1)}\right) \rightarrow S_k^{t+1} \tag{21}$$

where S_k^t is the state k a time t. S_k^t may contains a set of parameters such as velocity, acceleration or azimuth.

In [37], the sectorization-based prediction model has been proposed which has been applied to cellular networks. It is mostly a refinement from the basic regular path prediction model where a next cell is predicted based on a sequence of previous visited cells. Depending on predefined sectors in a cell, a node will be more likely to move the cell adjacent to its sector, or move to another sector. Fig. 8 illustrates the sectors in a cellular network.

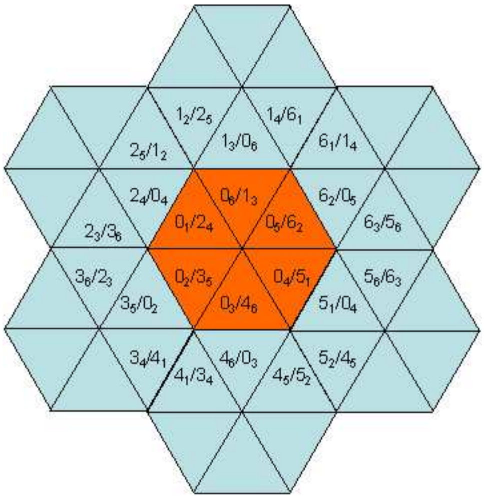


Figure 8: The Cell Sector Numbering Schema

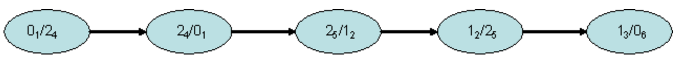


Figure 9: Sectorized Mobility

Based on the cell sector numbering schema, a history-based sectorized mobility is generated as illustrated in Fig. 9, where the probability to be at position m after X movements is given by

$$P_X(m) = \frac{X! p^{\frac{1}{2}(X+m)} (1-p)^{\frac{1}{2}(X-m)}}{\left[\frac{1}{2}(X+m)\right]! \left[\frac{1}{2}(X-m)\right]!} \tag{22}$$

where p is the probability to leave the cell.

Another approach introduced in [38] is the *Shadow Cluster* model, which is also a refinement from the basic regular path prediction model. Its concept is that any active wireless device establishes an influence on cells in the vicinity of its location and its direction of travel. The cells currently being influenced are said to form a *Shadow Cluster* because the region of influence follows the movement of the active device like a shadow. Fig. 10 illustrates this approach, where the shaded areas compose the shadow cluster centered in cell C.

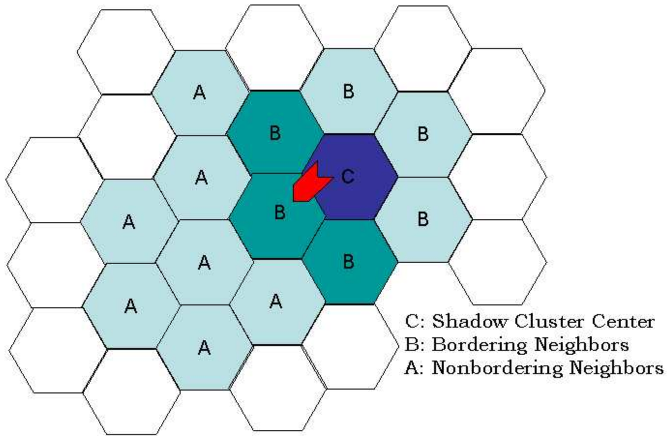


Figure 10: Shadow Clusters Produced by an active mobile terminal

Complex stochastic techniques are used to compute the active mobile probabilities to generate the shadow clusters. In [39], Akyldiz and Wang further improved the *Shadow Cluster* approach to consider aggregate history and a stochastic model of cell residence time to shrink the region considered for shadow clusters.

Finally, another particular class of history-based model uses Neural Networks. In those models, information is gathered in order to train the neural network, which then is able to predict a particular future state of the network. Based on the sequence of input vectors during the training period, back-propagation is used to update and improve the weights of the neural networks layers. Depending on the needed complexity, such neural network may have several hidden layers. Fig. 11 illustrates an example of the multi-later neural network with back-propagation.

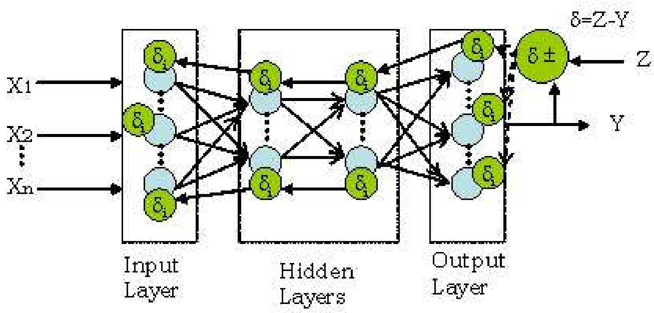


Figure 11: A Multi-later Neural Network with Back-propagation

where

X_i	:	State of node i
Y	:	Decision (Handover, Link duration)
Z	:	Real Value during training
δ_i	:	Weight correction during training

In Capka and Boutaba [40], the moving trajectory of a mobile node is determined as a sequence of base stations the node has been attached to. The neural network is trained with sequences observed in the past in order to detect the current movement pattern and improve network management.

Shang et al. [41] developed a clustering-based protocol using wavelet neural network, where a wavelet function replaces the Sigmoid function in conventional neural networks. They showed that this approach resulted in more stable clusters than LowID or MaxConn[3].

3.2.4 Hierarchic Models

This last category includes the most precise models ever developed at this time. Indeed, hierarchic models usually include a micro-prediction algorithm coupled with a macro-prediction schema. Most of the time, we find stochastic models for the micro-prediction, and history-based models for the macro-prediction. Those models are not only capable of predicting with a very high precision the sequence of cells a user will use in the future, but also the time it will reach the limit of each cell.

4 Network Algorithms using Prediction Models

In this section, we describe the divers application domains where mobility prediction schemes has been successfully adapted to mobile ad hoc networks.

Figure 12 illustrates domains where prediction models could be applied in mobile ad hoc networks. In most of those domains, protocols have been developed which significantly improved the network performance.

Mobility prediction techniques have been successfully applied to the following domains:

- Connection Management:
  - KADER [42]: This protocol generates a connected forest using a non-periodic maintenance strategy. Its Performance is similar to other topology control algorithms, yet at a drastic reduction of the maintenance overhead.

[3] LowID is a clustering technique where a node with the lowest ID is elected as cluster head, while MaxConn elects a node with the maximum connectivity as clusterhead

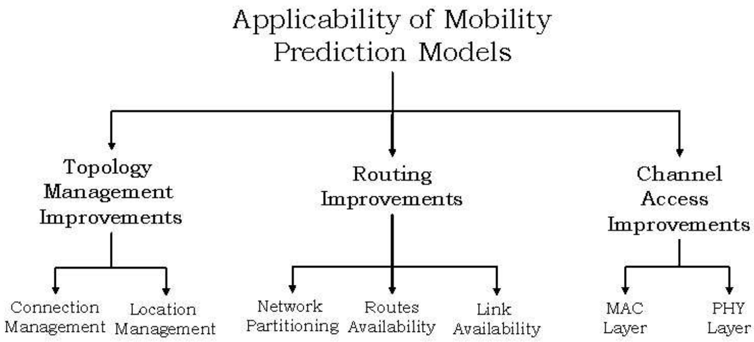


Figure 12: Classification of the Applicability of Prediction Techniques

  - Kinetic MultiPoint Relays (KMPR) [43]: The KMPR protocol elects MPR nodes depending on their predicted nodal degree. The KMPR protocol is able to reduce the MPR protocol maintenance overhead by 60% and the delay by 25

- Location Management:

  - Dead Reckoning-Based Location Service [44]: This model adjusts the periodic dissemination of geographic information based on a first order deterministic mobility prediction model.
  - Mobility Prediction-based GLS [45]: Improves the Grid Location Server (GLS) by adapting the periodic location maintenance with two prediction models *deterministic first order* and *history-based first order Markovien*.
  - Predictive Location Service (PLS) [46]: This approach only uses a first order deterministic prediction model, but manages to reduce the location errors and the maintenance overhead of GLS.

- Link Availability:

  - Mobility Prediction-based Position-based Forwarding (MP-PBF) [47]: This approach improves the accuracy of the general PBF protocol by relaying packets depending the predicted position of the intermediate nodes with respect to the predicted position of the destination.
  - Predictive Location Aided Routing (P-LAR) [48] : This approach sectorizes nodes mobility. The cost of routes establishment is largely lower than LAR.
  - Prediction-based Link Availability (PB-LA) [49]: Uses a deterministic first order prediction model to efficiently predict the link duration between two nodes. When used in conjunction with the DSR protocol, the performance is significantly improved.

- Context Aware Routing (CAR) [50]: Uses a Stochastic model based on Kalman Filters in order to predict mobility in sporadically connected networks. It illustrated its benefit compared with traditional epidemic routing.

- Route Availability:

  - Distance Vector with Mobility Prediction (DV-MP) [51]: Represents the link cost as the predicted link duration for Distance-Vector approaches, and improves their performance with respect to conventional Distance-Vector protocols or LAR.
  - Kinetic Minimum Spanning Trees (KMST) [52]: KMST uses a stochastic prediction model in order to build a Spanning Tree using a non-periodic maintenance strategy.
  - Dead-Reckoning Model (DRM) [53]: The DRM improves the performance of DSR by using the prediction of links duration instead of the hop count as the cost metric.
  - Reliable On-Demand Routing Protocol (RORP) [54]: This approach opens routes with weight set to the minimum link duration of each link comprised in the route. Then, the source chooses the route based on the maximum link duration.
  - AODV MOvement Prediction Routing (AODV-MOPR) [55]: In AODV-MOPR, nodes selected to establish a route from a source to a destination node are selected depending on their similar direction and velocity. This generates a 7% improvement in AODV route stability.
  - Kinetic Link State Routing (KLSR) [56]: Used in conjunction with KMPR, KLSR is able to use the actual and predicted future MPR selectors in order to build an optimal routing table containing not only actual but also future optimal paths, without requiring the periodic broadcast of Topology Control (TC) messages.

5 Conclusion

Telecommunication networks have long been subject to the effect of user mobility. In order to reduce this drawback on cellular networks (PCS, GSM), mobility prediction models have been created and successfully tested. Among other examples, this approach was seen as a way to provide some kind of quality of service to Wireless ATM networks. However, the prediction approach lost its popularity as those telecommunication networks were replaced by new systems such as 3G or wi-fi networks.

The subject reclaimed its popularity when mobility became again a major source of waste of network resource. For instance, predicting mobility in mobile ad hoc networks is seen as a mean to make such system scalable.

A large literature reading illustrates the advantage of mobility prediction models for mobile ad hoc network. However, unlike their counterpart in cellular networks, almost none of them use complex schemes. At the eve of mesh and vehicular networks, it would be interesting to reintroduce such approach. Moreover, the effect of prediction models on the physical or the Mac layer have not been studied yet.

References

[1] "Wireless open shortest path first," http://hipserver.mct.phantomworks.org/ietf/ospf.

[2] E. B. *et al.*, "Ospf mpr extension for ad hoc networks," February 2007, internet Draft, http://www.ietf.org/internet-drafts/draft-baccelli-ospf-mpr-ext-03.txt (work in progress).

[3] P. J. *et al.*, "Optimized link state routing protocol (olsr)," request for Comments, http://www.ietf.org/rfc/rfc3626.txt.

[4] T. C. *et al.*, "The optimized link state routing protocol version 2," February 2007, internet Draft, http://www.ietf.org/internet-drafts/draft-ietf-manet-olsrv2-03.txt (work in progress).

[5] D. J. *et al.*, "The dynamic source routing protocol (dsr) for mobile ad hoc networks for ipv4," request for Comments, http://www.ietf.org/rfc/rfc4728.txt.

[6] C. P. *et al.*, "Ad hoc on-demand distance vector (aodv) routing," request for Comments, http://www.ietf.org/rfc/rfc3561.txt.

[7] I. C. *et al.*, "Dynamic manet on-demand (dymo) routing," February 2007, internet Draft, http://www.ietf.org/internet-drafts/draft-ietf-manet-dymo-08.txt (work in progress).

[8] L. K. H. Takagi, "Optimal transmission ranges for randomly distributed packet radio terminals," *IEEE Transactions on Communications*, vol. 32, no. 3, pp. 246–257, March 1984.

[9] I. S. H. Frey, "Geographic and energy aware routing in sensor networks," in *Handbook of Sensor Networks: Algorithms and Architectures*, I. Stojmenović, Ed. Wiley, 2006, ch. 12, pp. 381–415.

[10] *et al.*. B. Karp, "Greedy perimeter stateless routing (gpsr)," http://www.icir.org/bkarp/gpsr/gpsr.html).

[11] Y.-B. Ko and N. H. Vaidya, "Location-aided routing (lar) in mobile ad hoc networks," in *Proc. of the 4th annual ACM/IEEE international conference on Mobile computing and networking (MOBICOM'98)*, 1998, pp. 66–75.

[12] Z. H. *et al.*, “Zone routing protocol (zrp),” July 2002, internet Draft, http://tools.ietf.org/id/draft-ietf-manet-zone-zrp-04.txt(work in progress).

[13] N. Nikaein, C. Bonnet, and N. Nikaein, “Hybrid ad hoc routing protocol (harp),” in *Proc. of the International Symposium on Telecommunications (IST'01)*, 2001.

[14] W. Creixell and K. Sezaki, “Mobility prediction algorithm for mobile ad hoc network using pedestrian trajectory data,” in *Proc. of the IEEE International Region 10 Conference (TENCON'04)*, November 2004.

[15] Z. Zaidi and B. Mark, “Mobility estimation based on an autoregressive model,” 2004, submitted to IEEE Transactions on Vehicular Technology, Jan. 2004. (Pre-print) Available at URL: http://mason.gmu.edu/zzaidi.

[16] T. Liu, P. Bahl, and I. Chlamtac, “Mobility modeling location tracking and trajectory prediction in wireless atm networks,” *IEEE Journal on Selected Areas in Communications*, vol. 16, no. 6, pp. 922–936, 1998.

[17] P. Pathirana, A. Savkins, and S. Jha, “Robust extended kalman filter based technique for location management in pcs networks,” *Elsevier Computer Communications*, vol. 27, pp. 502–512, 2004.

[18] I. G. *et al.*, “Enhancement to rss based indoor tracking systems using kalman filters,” in *Proc. of the Global Signal Processing Conf. (GSPx)*, April 2003.

[19] Z. Zaidi and B. L. Mark, “Real-time tracking algorithms for cellular networks based on kalman filtering,” *IEEE Transactions on Mobile Computing*, vol. 4, no. 2, pp. 195–208, March-April 2005.

[20] S.-Z. Yu and H. Kobayashi, “An integrated mobility and traffic model for resource allocation in wireless networks,” in *3rd ACM Workshop on Wireless Mobile Multimedia*, 2000, pp. 39–47.

[21] ——, “A hidden semi-markov model with missing data and multiple observation sequences for mobility tracking,” *ACM Signal Processing*, vol. 83, no. 2, pp. 235–250, February 2003.

[22] B. L. Mark and Z. Zaidi, “Robust mobility tracking for cellular networks,” in *Proc. of IEEE International Conference on Communications (ICC'02)*, May 2002.

[23] Z. Yang and X. Wang, “Joint mobility tracking and hard hand-off in cellular networks via sequential mote carlo filtering,” in *Proc. of the IEEE INFOCOM*, June 2002.

[24] F. G. *et al.*, “Particle filters for positioning, navigation, and tracking,” *IEEE Transactions on Signal Processing*, vol. 50, no. 2, pp. 425–437, February 2002.

[25] L. M. *et al.*, "Mobility tracking in cellular networks with sequential monte carlo filters," in *Proc. of the IEEE International Conference on Information Fusion*, July 2005.

[26] Z. Shah and A. Malaney, "Particle filters and position tracking in wi-fi networks," in *Proc. of the 63rd Vehicular Technology Conference (VTC'06 spring)*, vol. 2, 2006, pp. 613–617.

[27] B. Liang and Z. Haas, "Predictive distance-based mobility management for pcs networks," *IEEE Transactions on Networking*, vol. 11, no. 5, pp. 1–15, October 2005.

[28] J. Härri and C. Bonnet, "A lower bound for vehicles' trajectory duration," in *Proc. of the 62nd Vehicular Technology Conference (VTC'05 fall)*, 2005.

[29] J. Krumm, "Probabilistic inferencing for location," in *Proc. of Location-Aware Computing*, October 2003.

[30] R. Singer, "Estimating optimal tracking filter performance for manned maneuvering targets," *IEEE Transactions Aerospace and Electronic Systems*, vol. 6, pp. 473–483, July 1970.

[31] G. Liu and G. Maguire, "A class of mobile motion prediction algorithms for wireless mobile computing," vol. 1, no. 2, 1996, pp. 113–121.

[32] *A regular path recognition method and prediction of user movements in wireless networks*, 2001.

[33] J.-M. François and G. Leduc, "An entropy-based knowledge spreading and application to mobility prediction," in *Proc. of the 1st ACM/e-NEXT International Conference on Future Networking Technologies (CoNext'05)*, 2005.

[34] A. Bhattacharya and S. Das, "Lezi-update: An information-theoretic approach to track moble users in pcs networks," *ACM Wireless Networks (WINET)*, vol. 8, no. 2-3, pp. 121–135, 2002.

[35] F. Yu and V. Leung, "Mobility-based call admission control and bandwidth reservation in wireless cellular networks," *Elsevier Computer Networks*, vol. 38, pp. 577–589, 2001.

[36] J. G. *et al.*, "Restoration scheme of mobility databases by mobility learning and prediction in pcs networks," *IEEE Journal on Selected Area in Communications (JSAC)*, vol. 19, no. 10, pp. 1962–1973, 2001.

[37] R. Chellappa, A. Jennings, and N. Shenoy, "The sectorized mobility prediction algorithm for wireless networks," in *Proc. of the Int'l Conference on Information and Communication Technologies*, 2003.

[38] D. Levine, I. Akyildiz, and M. Naghshineh, "A resource estimation and call admission algorithm for wireless multimedia networks using the shadow cluster concept," *IEEE/ACM Transactions on Networking*, vol. 5, no. 1, pp. 1–12, February 1997.

[39] I. Akyildiz and W. Wang, "The predictive user mobility profi le framework for wireless multimedia networks," *IEEE/ACM Transactions on Networking*, vol. 12, no. 6, pp. 1021–1035, December 2004.

[40] J. Capka and R. Boutaba, *Mobility Prediction in Wireless Networks*, ser. Lecture Notes in Computer Science. Springer, 2004, vol. 3271, pp. 320–333.

[41] Y. Shang, W. Guo, and S. Cheng, *Clustering Algorithm Based on Wavelet Neural Network Mobility Prediction in Mobile Ad Hoc Network*, ser. Lecture Notes in Computer Science. Springer, 2005, vol. 3498, pp. 391–396.

[42] J. Haerri, N. Nikaein, and C. Bonnet, "Trajectory knowledge for improving topology control in mobile ad-hoc networks," in *Proc. of the 1st ACM/e-NEXT International Conference on Future Networking Technologies (CoNext'05)*, 2005.

[43] J. Haerri, F. Filali, and C. Bonnet, *On the application of mobility predictions to multipoint relaying in MANETs: kinetic multipoint relays*, ser. Lecture Notes in Computer Science. Springer, 2005, vol. 3837, pp. 143–155.

[44] V. Kumar and S. Das, "Performance of dead-reckoning based location service for mobile ad hoc networks," *ACM Wireless Communications and Mobile Computing Journal*, vol. 4, no. 2, pp. 189–202, March 2004.

[45] S. S. *et al.*, "A comparative study of mobility prediction schemes for gls location service," in *Proc. of the IEEE Vehicular Technology Conference (VTC'04)Conference*, 2004.

[46] X. Luo, T. Camp, and W. Navidi, "Predictive methods for location services in mobile ad hoc networks," in *Proc. of the 19th IEEE International Parallel and Distributed Processing Symposium (IPDPS'05)*, 2005.

[47] T. K. *et al.*, "Dead-reckoning for position-based forwarding on highways," in *Proc. of the 3rd International Workshop on Intelligent Transportation (WIT 2006)*, 2006.

[48] C. Doss, R. A. Jennings, and N. Shenoy, "Mobility prediction based routing for minimizing control overhead in mobile ad hoc networks," in *Proc. of the International Conference on Wireless Networks (ICWN'04)*, 2004.

[49] S. Jiang and D. He, "A prediction-based link availability estimation for routing metrics in manets," *IEEE/ACM Transaction on Networking*, vol. 13, no. 6, pp. 1302–1312, 2005.

[50] M. Musolesi, S. Hailes, and C. Mascoloy, "Adaptive routing for intermittently connected mobile ad hoc networks," in *Proc. of the 6th IEEE IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks (WoWMoM'05)*, 2005.

[51] W. Su, S. Lee, and M. Gerla, "Mobility prediction in wireless networks," in *IEEE Military Communication Conference (MILCOM)*, vol. 1, 2000, pp. 491–495.

[52] C. Gentile, J. Haerri, and R. E. V. Dyck, "Kinetic minimum-power routing and clustering in mobile ad-hoc networks," in *Proc. of the IEEE Vehicular Technology Conference (VTC'02 Fall)Conference*, 2002.

[53] A. Agarwal and S. R. Das, "Dead reckoning for mobile ad hoc networks," in *Proc. of the 2003 IEEE Wireless Communications and Networking Conference (WCNC'03)*, 2003.

[54] N. Wang and S. Chang, "A reliable on-demand routing protocol for mobile ad hoc networks with mobility prediction," *Elsevier Computer Communications*, vol. 29, pp. 123–135, 2005.

[55] H. Menouar, M. Lenardi, and F. Filali, "A movement prediction-based routing protocol for vehicle-to-vehicle communications," in *Proc. of the 1st International Vehicle-to-Vehicle Communications Workshop (V2V-COM'05)*, 2005.

[56] J. Haerri, F. Filali, and C. Bonnet, "Kinetic link state routing," Institut Eurécom, Technical Report 07-196, 2007.