

# Predictive Mobile IP Handover for Vehicular Networks

by

Alexander Magnano

Thesis submitted to the  
Faculty of Graduate and Postdoctoral Studies  
In partial fulfillment of the requirements  
For the M.A.Sc. degree in  
Electrical and Computer Engineering

School of Electrical Engineering and Computer Science  
Faculty of Engineering  
University of Ottawa

© Alexander Magnano, Ottawa, Canada, 2016

## **Abstract**

Vehicular networks are an emerging technology that offer potential for providing a variety of new services. However, extending vehicular networks to include IP connections is still problematic, due in part to the incompatibility of mobile IP handovers with the increased mobility of vehicles. The handover process, consisting of discovery, registration, and packet forwarding, has a large overhead and disrupts connectivity. With increased handover frequency and smaller access point dwell times in vehicular networks, the handover causes a large degradation in performance. This thesis proposes a predictive handover solution, using a combination of a Kalman filter and an online hidden Markov model, to minimize the effects of prediction errors and to capitalize on advanced handover registration. Extensive simulated experiments were carried out in NS-2 to study the performance of the proposed solution within a variety of traffic and network topology scenarios. Results show a significant improvement to both prediction accuracy and network performance when compared to recent proposed approaches.

## Acknowledgements

Foremost, I would like to express my sincere gratitude to my supervisor, Prof. Azzedine Boukerche for his inspiring guidance, extraordinary patience and technical assistance throughout my entire masters degree. His insightful guidance and critical comments have played an essential role in my research work and the completion of this thesis. His financial support and sense of humor have not only made my life easier, but have also increased my motivation to go deeper in my academic research. I shall treasure for my whole life this experience as a student with such a distinguished supervisor. My earnest thanks must also go to Dr. Xin Fei, a great mentor and friend over the past two years. His rigorous scholarship, constructive advices and truthful encouragement were really helpful for my learning and life experience. This thesis could not have been finished without the help and support from him. Special thanks go to my colleagues at PARADISE Research Laboratory for all the fun times and collaboration. The thorough discussions and great ambiance in the lab facilitated my work daily. Last but not least, my deepest gratitude to my parents and my friends, who have stood by me throughout my degree. Their unconditional love, endless support and continuous encouragement serve as the most powerful drive for my achievements.

## Publications

- A. Magnano, X. Fei, A. Boukerche, "Predictive Mobile IP Handover for Vehicular Networks," IEEE International Conference on Local Computer Networks, Oct 2015, pp 338-346.
- A. Magnano, X. Fei, A. Boukerche, "Movement Prediction in Vehicular Networks," IEEE Global Communications Conference 2015 (Accepted)
- A. Magnano, X. Fei, A. Boukerche, "Predictive Handover for Mobile IP in Vehicular Networks," IEEE Transactions on Vehicular Technology 2016 (Final Production)
- A. Boukerche, A. Magnano, N. Aljeri, "Mobile IP Handover for Vehicular Networks: Methods, Models and Classifications," EECS, Technical Report, University of Ottawa (in Progress-Main Contact: Prof. A. Boukerche)

## Glossary

<b>AP</b> Access Point	<b>MAP</b> Mobility Anchor Point
<b>V2V</b> Vehicle-to-Vehicle	<b>FBU</b> Fast Binding Update
<b>I2V</b> Infrastructure-to-Vehicle	<b>HAck</b> Handover Acknowledgment
<b>CoA</b> Care-of-Address	<b>FBAck</b> Fast Binding Acknowledgment
<b>HA</b> Home Agent	<b>LMA</b> Local Mobility Anchors
<b>FA</b> Foreign Agent	<b>MAG</b> Mobile Access Gateways
<b>HMIP</b> Hierarchical Mobile IP Handover	<b>PBU</b> Proxy Binding Update
<b>FMIP</b> Fast Mobile IP Handover	<b>PBA</b> Proxy Binding Acknowledgment
<b>PMIP</b> Proxy Mobile IP Handover	<b>EM</b> Expectation-Maximization
<b>KF</b> Kalman Filter	<b>SUMO</b> Simulation of Urban Mobility
<b>HMM</b> Hidden Markov Model	<b>NS-2</b> Network Simulator 2

# Table of Contents

List of Tables	x
List of Figures	xi
<b>1 Introduction</b>	<b>1</b>
1.1 Background . . . . .	1
1.2 Problem . . . . .	3
1.3 Contribution . . . . .	4
1.4 Thesis Organization . . . . .	6
<b>2 Related Work</b>	<b>8</b>
2.1 Hierarchical Handover . . . . .	8
2.1.1 Mobility Anchor Point Transition Costs . . . . .	10
2.1.2 Optimizing Mobility Anchor Point Selection . . . . .	11
2.1.3 Reducing Mobility Anchor Point Overheads . . . . .	12
2.1.4 Summary . . . . .	13
2.2 Fast Handover . . . . .	13
2.2.1 Improving Reliability . . . . .	17
2.2.2 Reducing Delay . . . . .	18

2.2.3	Summary . . . . .	19
2.3	Proxy Handover . . . . .	21
2.3.1	Reduce LMA Transition Costs . . . . .	24
2.3.2	Reducing LMA Overhead . . . . .	25
2.3.3	Summary . . . . .	26
2.4	Hybrid Handovers . . . . .	28
2.4.1	FMIP and HMIP . . . . .	28
2.4.2	FMIP and PMIP . . . . .	30
2.4.3	Summary . . . . .	31
2.5	Predictive Handover . . . . .	33
2.5.1	Probability Analysis . . . . .	35
2.5.2	Movement Projection . . . . .	40
2.5.3	Pattern Matching and Hybrid . . . . .	45
2.5.4	Neighbor Discovery . . . . .	48
2.5.5	Summary . . . . .	49
2.6	Further Improving Mobile IP . . . . .	49
<b>3</b>	<b>Motivation</b>	<b>51</b>
<b>4</b>	<b>System Modeling</b>	<b>53</b>
4.1	Hidden Markov Model . . . . .	53
4.2	HMM Learning . . . . .	55
4.2.1	Online Learning . . . . .	58
4.2.2	Initial HMM Estimation . . . . .	59
4.3	Movement Projection . . . . .	61

<b>5</b>	<b>Prediction Method</b>	<b>64</b>
5.1	Combining the Kalman Filter and the HMM . . . . .	65
5.2	Prediction Algorithm . . . . .	66
<b>6</b>	<b>Predictive Handover Protocol</b>	<b>68</b>
6.1	Neighbor Discovery and HMM Updating . . . . .	69
6.2	Predictive Handover Protocol . . . . .	69
<b>7</b>	<b>Performance Evaluation</b>	<b>75</b>
7.1	Environment Setup . . . . .	75
7.1.1	Road and Traffic . . . . .	76
7.1.2	Network . . . . .	77
7.2	Parameter Optimization . . . . .	80
7.3	HMM Learning Analysis . . . . .	82
7.4	Prediction Analysis . . . . .	85
7.4.1	Urban with AODV . . . . .	85
7.4.2	Highway and Urban/highway . . . . .	87
7.4.3	OLSR and GPSR . . . . .	88
7.4.4	Vehicle Location Predictions . . . . .	89
7.5	Network Performance . . . . .	91
7.5.1	Urban . . . . .	92
7.5.2	Highway and Urban/highway . . . . .	99
<b>8</b>	<b>Conclusion and Future Work</b>	<b>102</b>
8.1	Conclusion . . . . .	102
8.2	Future Work . . . . .	103



APPENDICES	105
A An Illustrative Example	106
References	108

# List of Tables

2.1	Hierarchical Handover Approaches . . . . .	9
2.2	Hierarchical Handover Performance Comparison . . . . .	14
2.3	Fast Handover Approaches . . . . .	15
2.4	Fast Handover Performance Comparison . . . . .	20
2.5	Proxy Handover Approaches . . . . .	22
2.6	Proxy Handover Performance Comparison . . . . .	27
2.7	HMIP, FMIP, and PMIP Hybrid Handover Approaches . . . . .	29
2.8	Hybrid Handover Performance Comparison . . . . .	32
2.9	Predictive Handover Approaches (Probability Analysis) . . . . .	34
2.10	Predictive Handover Performance Comparison (Probability Analysis) . . . . .	37
2.11	Predictive Handover Approaches (Movement Projection) . . . . .	38
2.12	Predictive Handover Performance Comparison (Movement Projection) . . . . .	41
2.13	Predictive Handover Approaches (Pattern Matching and Hybrid) . . . . .	43
2.14	Pattern Matching and Hybrid Performance Comparison . . . . .	45
7.1	Simulation Parameters . . . . .	79
7.2	Observations $M$ versus Prediction Accuracy $P$ . . . . .	81
7.3	Threshold Values . . . . .	82
7.4	HMM-KF Parameters . . . . .	83

# List of Figures

1.1	Mobile IP handover registration . . . . .	2
2.1	HMIP handover architecture . . . . .	10
2.2	Proactive fast handover protocol . . . . .	13
2.3	Reactive fast handover protocol . . . . .	16
2.4	Proxy handover architecture . . . . .	21
2.5	Proactive PMIP handover . . . . .	23
2.6	Reactive PMIP handover . . . . .	24
2.7	Movement and statistical variable differences . . . . .	33
4.1	Deriving the hidden Markov model . . . . .	56
4.2	Modeled Kalman filter testing . . . . .	63
5.1	Prediction method overview . . . . .	64
5.2	HMM observations with(d) and without(c) the Kalman filter . . . . .	66
6.1	Predictive handover protocol . . . . .	71
6.2	Prediction error protocol . . . . .	74
7.1	Road networks imported into SUMO . . . . .	78
7.2	Testing the HMM learning in the mobile IP network . . . . .	83

7.3	Prediction accuracies versus traffic density . . . . .	86
7.4	Prediction accuracies versus traffic density . . . . .	91
7.5	Latency results versus vehicle density . . . . .	93
7.6	FPMIP-PT and C-HMIP compared latency results . . . . .	95
7.7	Throughput results versus vehicle density . . . . .	97
7.8	FPMIP-PT and C-HMIP compared throughput results . . . . .	98
7.9	Packet drop rate versus vehicle density . . . . .	99
7.10	FPMIP-PT and C-HMIP compared packet drop results . . . . .	100
7.11	Performance averages across all environments . . . . .	100
A.1	Problematic scenarios (left) resolved by proposed prediction method (right)	106

# Chapter 1

## Introduction

Rapid expansion of wireless networks has given rise to the development of vehicular networks for such services as infotainment, road safety, and traffic management. This development has included advanced routing protocols for effective inter-vehicular communication; however, providing IP services to vehicles remains challenging. In this chapter, we first provide an overview of mobile IP and vehicular networks. This is followed by the problems of the mobile IP handover within a vehicular network, and details on our contribution and the thesis organization.

### 1.1 Background

Vehicular networks are a developing technology aimed to provide high-speed wireless communication between vehicles. These networks, composed of vehicles and road-side access points (APs), are similar to other wireless networks, but with some key differences required for handling the increased movement of vehicles. One such difference is the use of vehicle-to-vehicle (V2V) communication in addition to infrastructure-to-vehicle (I2V) communication. Having APs cover the entire area of a vehicle's potential movement is unrealistic, but gaps between coverage are resolved with V2V. There is also the development of IEEE802.11p and numerous routing protocols to handle rapidly shifting network topologies. However, vehicular networks continue to have many challenges that prevent

performance from reaching closer numbers to other wireless networks.

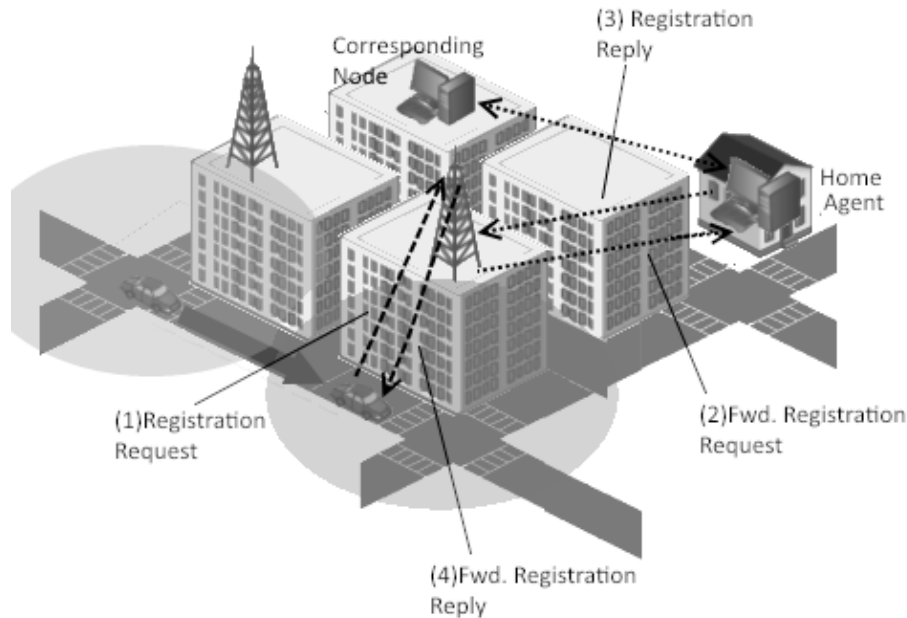


Figure 1.1: Mobile IP handover registration

Mobile IP is the procedure used for providing Internet access to wireless nodes as they connect to different access points. When a moving node begins transitioning between AP coverages, it conducts a mobile IP handover to continue its IP session through the new AP. The mobile IP handover procedure consists of two main steps: access point discovery and registration. Discovery, triggered by a weakening signal with the mobile node's current AP, is the process of the mobile node finding the closest neighbor AP to commit to. This involves the mobile node listening on various channels for AP ad messages which contain registration information. The mobile node chooses an AP to commit to based on the signal strengths of the different ads, and then initiate registration with that AP, as visualized in Figure 1.1. Registration consists of the mobile node setting up a care-of-address (CoA) with the new AP and notifying its home agent (HA) of the new location. The HA then updates the location for the mobile node and begins forwarding IP packets to the new CoA.

Completion of the handover process can be very slow, causing latencies up to one second [23]. During this time, the mobile node is unable to receive any IP packets. Common contributions to this delay include the time it takes the mobile node to discover the nearby

APs, and the potentially large communication distance between the mobile node and its HA, which increases the time to exchange packages during registration.

## 1.2 Problem

The problems addressed in this thesis are centered around performance issues of the mobile IP handover and poor handover prediction accuracy within vehicular networks. While the handover creates less of an issue for slow moving nodes, its high costs are unsuitable for the increased and frequent movement of vehicles. Providing an uninterrupted IP connection, which is required for many IP services, mandates smooth transitions between Internet access points, but the mobile IP handover's expensive procedure causes service interruption during the AP transfer. Because of this handover cost, a vehicle that is frequently transferring between APs will suffer constant interruptions and large performance degradation. For example, consider a vehicle moving at  $30m/s$  that switches to a new AP every  $300m$ , spending approximately ten seconds within each AP. The handover would consume up to 10% of the vehicle's dwell time before IP services would be accessible again. Additionally, every 10 seconds the IP services will be interrupted, making them unusable.

Different methods to improve the mobile IP handover have been investigated, but improved performances have yet to meet IP service requirements. Popular approaches include the fast handover (FMIP) [35], the hierarchical handover (HMIP) [34], and the proxy handover (PMIP) [52]. These approaches focus on reducing the latency costs of the discovery and registration processes, but mobile IP requirements make it difficult to reduce performance costs enough for vehicular network compatibility. Additionally, if there is densely populated AP, the packet overhead of the handover starts causing large packet drop rates. Most approaches, however, add more overhead rather than reduce to improve the latency issues. Properly balancing these different performance metrics remains a challenge because improving upon one often involves sacrificing the other.

A potential solution to these issues is a predictive handover, because a correctly predicted handover can conduct the majority of the procedure in advance to produce a near-

ideal handover performance. In addition, the additional overhead produced for the prediction occurs before the handover, which has less consequence on the vehicle's connectivity. However, predictive approaches have had less popularity due to the difficulty of accurately predicting a handover. To capitalize on advance handover benefits, a predictive handover scheme must achieve a high success rate. Otherwise, the performance improvement is too small to outweigh the extra costs for conducting a prediction. While vehicle movement is limited by road restrictions, some aspects of a vehicular network are still problematic for predicting a handover. One such issue is the difficulty of encompassing the large variety of road and intersection scenarios to maintain prediction reliability. Another problem is the road restrictions forcing vehicles to move in indirect directions, misleading prediction algorithms. Other than prediction accuracy, there are also the issues of handling a prediction error to minimize performance degradation, exchanging network information essential to making a prediction, and approaching an advanced handover to achieve best performance. All of these problems should be addressed to meet the performance requirements of a vehicular network.

### 1.3 Contribution

In this thesis, a predictive handover using a robust prediction method is proposed to improve mobile IP performance within vehicular networks. By improving the reliability of the prediction method, the frequency of advanced handovers will reach a point where a vehicle will experience few interruptions in its IP services. The central contributions of this thesis are described as follows.

- A prediction method is proposed that combines vehicle movement projection with stochastic probability analysis to determine the next handover. Vehicle projections are made using a Kalman filter (KF) [66], which are then fed into a hidden Markov model (HMM) [67] as observations. In combining the probabilistic and temporal data, accuracy is maintained in a wider variety of situations, thus improving reliability and accuracy. The KF and HMM are modeled to resolve previously problematic



situations, such as misleading vehicle movement. Both of these tools have low calculation costs, ensures that the prediction is made efficiently for the small time frame required to complete an advanced handover in a vehicular network.

- An online incremental learning method for the HMM is modeled and adjusted to the mobile IP network to continuously learning the system probabilities and determine the most likely neighbor AP for handover prediction. In addition, we propose an initial HMM learning approach that takes advantage of the mobile network to improve HMM convergence rates and include network metrics in probability calculations. The latter improves network performance by having vehicles favor handovers with APs that will provide better coverage.
- A information protocol within mobile IP is proposed for APs to gather necessary neighbor information and HMMs to learn system probabilities. The protocol uses information packets sent over the network between vehicles and APs. The approach is self-maintaining, requiring no initial maintenance and allowing each AP to adapt to their individual environments. This way, the protocol can be easily implemented and self-maintained. The method advances a neighbor discovery approach previously proposed by A. Mishra et al. [19].
- A predictive handover protocol is proposed and designed to capitalize on the benefits of a successfully predicted handover and to minimize costs of a prediction error. This includes handling problems such as ensuring early registration is completed before the handover occurs and detecting when a prediction error has occurred. By addressing these issues, performance is further improved beyond just the improved accuracy.
- Extensive simulations experiments are ran and illustrated to test the prediction and network performance of the proposed predictive method in comparison with other recent methods. The simulations are done using data from multiple road environments, three different MAC layer routing protocols, and across a variety of vehicle densities. This provides insight to the robustness of each handover approach and the effects that different road environments and routing protocols have on the network performance.

## 1.4 Thesis Organization

The rest of the thesis is organized as followed:

- Chapter 2 reviews related literature that discusses improving the mobile IP performance. We organize the literature into different categories and analyze the differences between them, as well as literature that combines multiple categorical approaches. We also discuss the benefits and issues of the methods discussed within the literature, and explain the relationships between these methods.
- Chapter 3 reviews the problems discussed in the related literature, and from these problems we explain the theoretical concept and reasoning behind our approach. This is followed by an illustrated example showing how our approach addresses and resolves issues raised in other literature.
- Chapter 4 describes the mathematical tools used for the proposed predictive approach. This includes mathematically modeling the Kalman filter and hidden Markov model, and deriving equations used for the prediction and online HMM learning.
- Chapter 5 proposes the prediction algorithm used for determining the next most likely AP. This includes details on combining the derived mathematical tools and optimizing variable parameters to perform better within a vehicular network. Discussion of the decisions we made are also provided, explaining the benefits of our chosen approach.
- Chapter 6 proposes our predictive handover protocol and details how the prediction method is implemented within the mobile IP network. This includes details on how neighboring APs are discovered, how information is retrieved for HMM learning, where the HMM and Kalman filter are located, the protocol followed for both correct and incorrect predictions, and detection of a prediction error.
- Chapter 7 describes the evaluation environment and analyzes the proposed solution compared to other methods. This includes separate analysis of the prediction accuracy and the network performance to provide more insight into the effectiveness of

our approach. We simulate a large set of comparison methods, road networks, and vehicular network routing protocols to test the robustness of our approach.

- Section 8 concludes the thesis, where we review our proposed approach and a summary of what we derived from simulation analysis. This is followed by a discussion on potential research topics to continue the work presented this thesis.

# Chapter 2

## Related Work

Literature on improving mobile IP handover performance is reviewed. Surveys on handover literature can be found in [32] and [33], with a survey specifically on the used algorithms in [114–116]. The handover methods proposed by the literature can be organized into the four categories: hierarchical, fast, proxy, and predictive; all of which are focused on resolving one or more of the problems in the mobile IP handover. In Section 2.1, we provide a summary of the hierarchical handover and related literature that propose methods of improving it. We then detail the fast handover by the same approach in Section 2.2. This is followed by the proxy handover in Section 2.3 and hybrid approaches of the hierarchical, fast, and proxy approaches in Section 2.4. In Section 2.5, we analyze the literature of the different predictive method, which we then follow with our conclusion in Section 2.6.

### 2.1 Hierarchical Handover

H. Soliman et al. [34] introduce the hierarchical mobile IP handover, which utilizes mobility anchor points (MAP) to decrease registration costs for vehicles that are distant from their HA. An overview of the HMIP architecture is shown in Figure 2.1.

A MAP acts a temporary local home address for a vehicle within a foreign network. The MAP is located within close distance of the vehicle, allowing registration between APs to be completed through only local communication. This prevents communication with a

Table 2.1: Hierarchical Handover Approaches

Article	Approach	Advantages	Drawbacks
H. Soliman et al. [34] (2003)	Adds MAPs to act as local HAs	Reduces registration tunneling	Large latency costs for MAP transitions
H. Teng et al. [1] (2011)	Context messaging between MAPs	Smoother MAP transitions	AP discovery costs, still suffers if MAPs are distant
P. Nath et al. [2] (2012)	MAPs use AP tables, uses vehicle movement for management	Smoother AP and MAP transitions	Low scalability due to maintaining tables, movement unreliable
K. Kawano et al. [3] (2002)	Adds a hierarchical tree structure to MAPs	Improved MAP coordination, balancing MAP loads	Increased overhead for MAP coordination
T. You et al. [4] (2003)	Vehicles connect to a primary and secondary MAP	Improves reliability and consistency of handover performance	Increased overhead and network loads
J. Lee et al. [7] (2003)	Adds IP paging extension	Reduces unnecessary MAP and vehicle communication	Increased latency delay when MAP needs to forward IP packets to vehicle
M. Yi et al. [5] (2003)	Use dwell-time estimation to reduce binding updates	Reduces packet overhead for vehicle tracking	Large performance costs when estimation error occurs
E. Mirzamany et al. [6] (2012)	Two-layer MAPs for intra-domain handovers	Reduced intra-domain latency, reduced inter-domain frequency	Costly inter-domain transitions, increased complexity

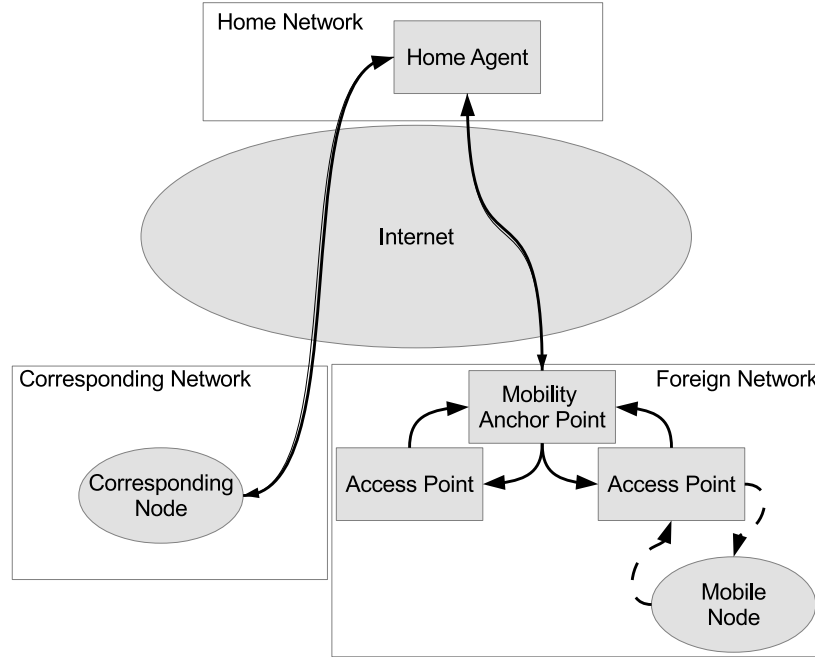


Figure 2.1: HMIP handover architecture

distant HA that would cause large registration latency. Each MAP covers a set number of APs within an region, and so the vehicle's HA must only be contacted upon switching between MAPs. By reducing the frequency of HA communication, the average registration latency is also reduced. The issues that remain within the HMIP approach are the MAP transition costs, optimizing a vehicle's MAP selection to prevent frequent MAP transitions, and reducing the MAP overheads.

### 2.1.1 Mobility Anchor Point Transition Costs

The costly procedure of a vehicle switching between MAPs may create inconsistent handover performances. When a switch happens, the handover procedure becomes more expensive than the standard mobile IP handover, due to the latency and overhead costs for contacting the HA and also setting up the new MAP. To address this problem, H. Teng et al. [1] add a context message protocol between anchor points. The context messages allow MAPs to communicate information directly to each other during a MAP transition. This reduces the required registration exchange between the vehicle and MAP, and additionally

lowers the registration delay since one MAP can directly begin forwarding IP packets to the vehicle's new MAP. One issue of this approach is the additional network load produced by the context messages.

Another approach that uses context messages to reduce transition costs is proposed by P. Nath et al [2]. In their approach, the MAPs maintain tables of all nearby APs and that AP's related MAP. The vehicle's current MAP uses the vehicle's general movement to estimate which AP it will connect to next. The MAP will then check its tables and see if the vehicle is about to leave to another domain. If it is the case, the MAP will send context messages to the new MAP to reduce the registration costs of MAP transitions. This allows for smoother MAP transitions when successful; however, the problem is two fold. First, having MAPs maintain tables on all nearby APs can be difficult to maintain or update and is not scalable if AP density is high. Second, a vehicle's movement is unreliable, which will cause many incorrect assumptions about which AP is next.

### 2.1.2 Optimizing Mobility Anchor Point Selection

There is also the issue of a vehicle connecting to a MAP that is not optimal for performance. This would occur when a vehicle moving on the line between two MAP domains. K. Kawano et al. [3] introduce a multilevel HMIP approach which adds a tree structure to MAPs that allows improved MAP coordination and better distribution of network loads. This shows to improve MAP performance, but suffers from the increased overhead required for MAP coordination. Another approach to this issue is by T. You et al. [4], who propose a robust HMIP where a vehicle registers with a primary and secondary MAP at the same time. If failure to connect with the primary MAP occurs, or if the vehicle quickly transitions away from that MAP, it can quickly recover by switching to the secondary MAP. This reduces the recovery time that occurs when the wrong MAP is selected. However, registering with two MAPs incurs additional overhead and network loads to the MAPs.

### 2.1.3 Reducing Mobility Anchor Point Overheads

When vehicle densities increase, the high overhead and network loads for MAPs can put large stress on the network. This issue is amplified by methods adding more overhead and MAP loads to resolve other HMIP issues. J. Lee et al. [7] add an IP paging extension to HMIP to reduce the amount of vehicle tracking otherwise required by the MAP. The central idea is the MAP pages for the vehicle's location upon receiving packets destined for it. By having a paging system, the MAP only needs to keep close track of active vehicles. This reduces the unnecessary tracking otherwise required for vehicles that are idling within a MAP's domain. One issue of this approach is the additional delay before a vehicle receives the first wave of IP packets upon becoming active.

Another method to decrease overhead and load costs is discussed by M. Yi et al. [5], who propose a dwell-time estimation method to reduce binding updates, which are used for vehicle tracking. The method estimates how long a vehicle will remain within the MAP's region and the current AP's range by observing the vehicle's current mobility. Binding updates can then be sent according to this estimation instead of at predefined increments, thus reducing the packet exchanges required for the MAP to track a vehicle. However, large performance costs can occur if a vehicle changes location in less time than what was estimated.

E. Mirzamany et al. [6] propose using a two-level MAP system to distribute the MAP load without affecting communications with the corresponding node. The two-level system is composed of global MAPs (GMAP) and local MAPs (LMAP), where GMAPs are located at the standard MAP location, and LMAPs are setup between the GMAP and APs. This allows each MAP to have a lower load, since LMAPs handle fewer APs than a standard MAP, and the GMAP only has to communicate with the vehicle during LMAP transitions. Another benefit of this approach is it reduces intra-domain handover latency, since LMAPs are even more local and with lower MAP loads, their response will be near-instant. One issue with this approach is it would incur additional overhead upon a domain transfer, as the vehicle would then have to establish connection through an LMAP and GMAP.



### 2.1.4 Summary

In summary, the HMIP's reduction of HA communication improves the handover registration costs when a vehicle is within a single MAP's region. However, some HMIP aspects, such as switching between MAP regions and its additional overhead, remain problematic and limit its improvement to network performance. In addition, HMIP approaches do not address the performance issues of the handover's discovery process.

## 2.2 Fast Handover

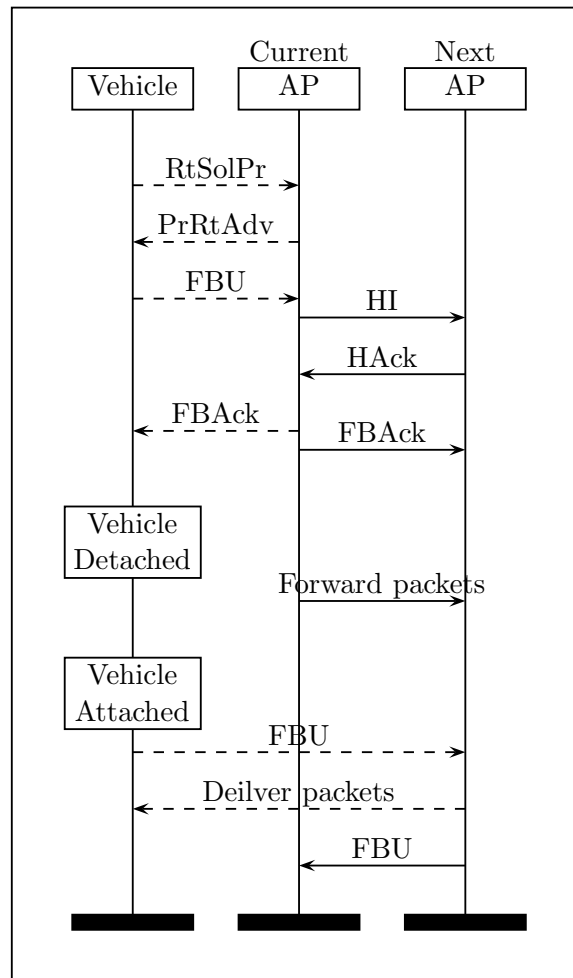


Figure 2.2: Proactive fast handover protocol

Table 2.2: Hierarchical Handover Performance Comparison

Article	Handover Latency	MAP Transition Cost	Overhead	Network Load	Consistency
H. Soliman et al. [34] (2003)	Medium	Large	Medium	Medium	High
H. Teng et al. [1] (2011)	Medium	Low	Medium	High	Medium
P. Nath et al. [2] (2012)	Medium	Low	Low	High	Low
K. Kawano et al. [3] (2002)	Medium	Medium	High	Low	High
T. You et al. [4] (2003)	Medium	Low	High	High	High
J. Lee et al. [7] (2003)	Medium	Medium	Low	Low	Low
M. Yi et al. [5] (2003)	Medium	Large	Low	Low	Medium
E. Mirzamany et al. [6] (2012)	Low	Medium	Low	Medium	Low

Table 2.3: Fast Handover Approaches

Article	Approach	Advantages	Drawbacks
G. Tsirtsis et al. [35] (2003)	MAC triggers to initiate handover	Minimize AP discovery costs	Unreliable cross-layer communication
M. Boutabia et al. [10] (2013)	Ensure FBack message delivery with media independent protocol	Improve reliability of FMIP	Additional overhead, only improves FBack reliability
A.S. Sadiq et al. [11] (2014)	Increase accuracy of signal strength prediction	Improve reliability of FMIP	Additional overhead and calculation, only improves AP selection
H. Huang et al. [12] (2009)	Pre-binding update packets exchanged for establishing earlier connection	Increases FMIP success rate	Increased overhead and latency
N. V. Hanh et al. [39] (2008)	AP initiates HA registration instead of vehicle	Reduces registration latency	Amplifies consequences of an FMIP error
H. Kim et al. [13] (2006)	Neighbor AP info found in advance, used to send early binding update packets	Improves FMIP reliability and reduces handover latency	Increases overhead for added packets and neighbor discovery
S. Kim et al. [31] (2011)	Adds packet buffering in APs after MAC triggers	Reduces handover latency	High overhead and large memory costs for buffering

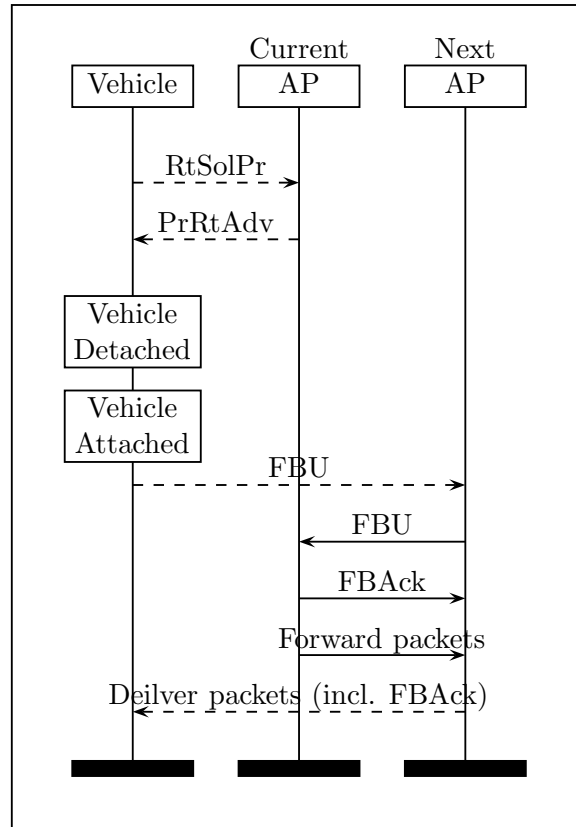


Figure 2.3: Reactive fast handover protocol

The fast handover, introduced by G. Tsirtsis et al. [35], utilizes MAC layer triggers to notify the mobile IP layer of an upcoming handover. Since the MAC layer handover occurs first, the early trigger allows the vehicle to discover its upcoming AP and reserve a CoA before the mobile IP handover starts. Reserving a CoA in advanced allows earlier packet-forwarding to the new AP, which produces a smoother transition with fewer packets dropped. However, MAC triggers can often be incorrect, and communication between layers is unreliable. Thus, the fast handover relies on proactive and reactive protocols followed when either the handover is successful or unsuccessful.

The steps of the two fast handover protocols are shown in Figures 2.2 and 2.3, where dashed lines represent wireless communication and solid lines represent communication over the wired network. Both protocols begin with a router solicitation for proxy advertisement (*RtSolPr*) and a proxy router advertisement (*PrRtAdv*) message. These messages are exchanged every time the vehicle receives a new MAC-layer advertisement from a neigh-

boring AP. The vehicle sends a RtSolPr message to its current AP, requesting the related IP information with the newly discovered AP's IP information. In a proactive situation, the vehicle chooses which AP it will connect to next before it disconnects with its current AP. It will send a fast binding update (FBU) message to its current AP, which will then establish a new CoA with the next AP using a handover initiate (HI) and a handover acknowledge (HACK) message. The current AP then sends a fast binding acknowledgment (FBACK) message to both the next AP and vehicle for confirmation. The current AP then forwards the IP packets to the next AP, which delivers the vehicle upon its reconnection. In the case of a reactive protocol where the vehicle fails to send the FBU before disconnecting, it sends the FBU to the next AP, which then forwards the FBU to the previous AP, and the same handover procedure is followed.

A proactive fast handover can reduce latency and packet drop rates, but there still exists a few performance issues that prevent the fast handover from meeting IP requirements. These problems include the unreliability of the proactive protocol and reducing the delay of the proactive handover exchange.

### 2.2.1 Improving Reliability

An issue that arises in the FBACK message is it is also sent through the MAC layer. This results in the FBACK message to not always be delivered, which causes the vehicle to incorrectly assume a failure. This is particularly a problem within vehicular networks, as their frequent movement makes them regularly susceptible to interference [106]. A solution to this problem is proposed by M. Boutabia et al. [10], who utilize the media independent protocol to ensure delivery of the FBACK message. Since the media independent protocol is not linked to a specific layer, it can communicate with both the MAC and network layers. This allows it to more reliably check packet delivery, thus reducing the chances of a fast handover failure.

There is also the issue of choosing the wrong neighboring AP because of unreliable MAC signal readings. This issue is approached by A.S. Sadiq et al. [11], who propose using a curve fitting model to accurately interpret changing signal strengths of neighboring APs.

This prevents any brief signal disruptions causing the vehicle to choose the wrong AP, and also stops the MAC trigger from occurring at an inappropriate time. Since a vehicle may be moving fast, frequent small disruptions could occur, showing how this approach could be useful within the context of vehicular networks. A similar approach is proposed by S. Samarah et al. [90]. Their method uses sequential patterns instead of a curve fitting model to track the signal strengths. However, an issue with these approaches is the proper setting of the curve fitting model, which could prove to be difficult with the variety of environments a vehicle network may be operating in. A different method to ensuring correct AP selection is discussed by A. Boukerche et al. [89,92,111] and Y. Ren [102,112], who both introduce agent-based trust methods and reputation management schemes. This would provide vehicles insight into which AP is more reliable to connect to.

Another approach that focuses on improving reliability of the fast handover is investigated by H. Huang et al. [12]. They add a pre-binding update scheme to establish connection with the next AP before the fast handover packets are exchanged. With a pre-established connection, the fast handover exchange has a much higher chance of success. In addition, the approach add a packet forwarding trigger that is separate from the fast handover. This approach waits for confirmation of a successful AP connection before redirecting packets to the new AP. Thus, the trigger prevents a fast handover error from forwarding packets to the wrong AP. An issue with this approach is it does not address the case of a pre-binding packet consistently failing to transfer, possible if a certain channel is very crowded. A method that addresses this issue is proposed by A. Boukerche et al. [93,94], who use an efficient distributed algorithm to better distribute bandwidth within an AP. Similarly, T. Antoniou et al. [100] utilize a variable transmission range protocol to manage each APs bandwidth and reduce the number of dropped packets.

### 2.2.2 Reducing Delay

In addition to unreliability, the fast handover process can also often fail to finish within the handover time frame, especially within a vehicular network where the faster moving vehicles reduce the AP transition time. N.V. Hanh et al. [39] address this issue by simplifying the

fast handover procedure to ensure registration completion before the AP transition. The approach has the AP reserve the CoA instead of the vehicle, thus the AP can initiate the registration process earlier and the number of exchanged packets is reduced. However, in reducing the costs of the fast handover, the consequences for an error are amplified. By having the AP initiate registration earlier without more confirmation, both the chance and number of packets forwarded to the wrong AP increase.

S. Kim et al. [31] also approach the issue of the fast handover delay by implementing registration buffers inside the APs to reduce packet exchanges. These buffers are used by APs to store neighboring AP registration information, which is shared by AP advertisements sent over the network. By having this information available in advance, vehicles can skip packet exchanges otherwise required to initiate the early registration and reserve a CoA. The reduced exchanges improves the FMIP latency costs for handling vehicular movements. However, this approach suffers similar unreliability problems as [39].

Both reliability and latency are addressed by H. Kim et al. [13], who consider the methods proposed in [31] and [12]. In this approach, similar AP advertisements sent over the network are used to discover neighboring AP information. This information is used by the vehicle to send early binding update packets, which are used to improve the reliability of the MAC layer trigger, and to improve the response time of the FMIP registration. The consequence of this approach is the additional binding update and advertisement packets add up to large overhead costs.

### 2.2.3 Summary

Considerable improvements have been made to certain aspects of the fast handover, but usually at an expense of another performance issue. For example, in [10] the reliability is improved, but overhead and delay is increased due to the increased exchange required for ensure successful message exchanges. In addition, the fast handover does not address the potentially large registration latencies, which can additionally cause a proactive handover failure.

Table 2.4: Fast Handover Performance Comparison

Article	Handover Latency	Packet Drop Rate	Overhead	Network Load	Consistency
G. Tsirtsis et al. [35] (2003)	Medium	Medium	Medium	Medium	Low
M. Boutabia et al. [10] (2013)	Medium	Low	High	Medium	Medium
A.S. Sadiq et al. [11] (2014)	Medium	Low	High	Medium	Medium
H. Huang et al. [12] (2009)	Medium	Medium	High	High	High
N. V. Hanh et al. [39] (2008)	Low	Medium	Medium	Medium	Low
H. Kim et al. [13] (2006)	Low	Medium	High	High	High
S. Kim et al. [31] (2011)	Low	Low	High	High	Low



## 2.3 Proxy Handover

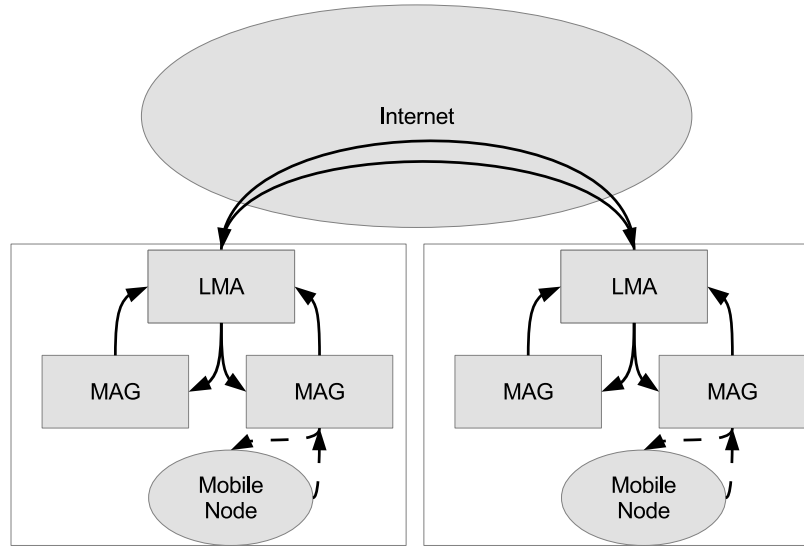


Figure 2.4: Proxy handover architecture

The proxy handover takes a different approach than the fast and hierarchical handover. Instead of a mobile node-based approach, PMIP is network-based, where the network manages the vehicle and conducts the handover in full. PMIP is first proposed by S. Gundavelli et al. [52], who introduce local mobility anchors (LMAs) and mobile access gateways (MAGs) that manage IP mobility for the vehicles. An overview of the new architecture is illustrated in Figure 2.4. The LMA acts similarly to an HA, but keeps more detailed tracking on each individual vehicle. Additionally, it is guaranteed to be more local than an HA, as it only manages a specific network domain. LMAs communicate with MAGs, which can be compared to MAPs within the HMIP protocol but which conduct additional processes. MAGs track a vehicle's movement as it moves between APs, and conducts the handover with the LMA for the vehicle when a MAG transition occurs. Instead of the handover re-registering and the vehicle generating a new CoA, the process instead involves the LMA updating the vehicle's location within its cache. The goal behind this approach is to minimize wireless communication. Since it is more expensive and

Table 2.5: Proxy Handover Approaches

Article	Approach	Advantages	Drawbacks
S. Gundavelli et al. [52] (2008)	Introduces LMAs and MAGs for APs to manage handovers	Minimizes wireless communication	Expensive domain transfers
K. Lee et al. [53] (2010)	Adds intermediate MAGs to handle domain transfers	Reduces inter-domain transfers to similar costs as intra-domain	Determining placement of intermediate MAGs is problematic
S. Ro et al. [54] (2015)	Introduces overlap-MAGs to maintain IP connection	Improves handover consistency	Added complexity for determining placement and number of overlap-MAGs
N. Neumann et al. [55] (2009)	IP Packet forwarding between LMAs	Reduces latency of domain transfers	Circuitous packet routes
T. Chiba et al. [56] (2008)	Optimize packet routes by recognizing close MAGs	Reduces LMA communication	Increases penalty for MAG transitions
H. Jung et al. [57] (2011)	Increases MAG communication with broadcasting messages to reduce LMA loads	Reduces LMA communication and overhead	Additional MAG overhead and increases domain transfer costs
S. Son et al. [58] (2014)	Use vehicle speed to determine handover priority	Reduces overhead and network load	Vehicle speed is unreliable, costly errors

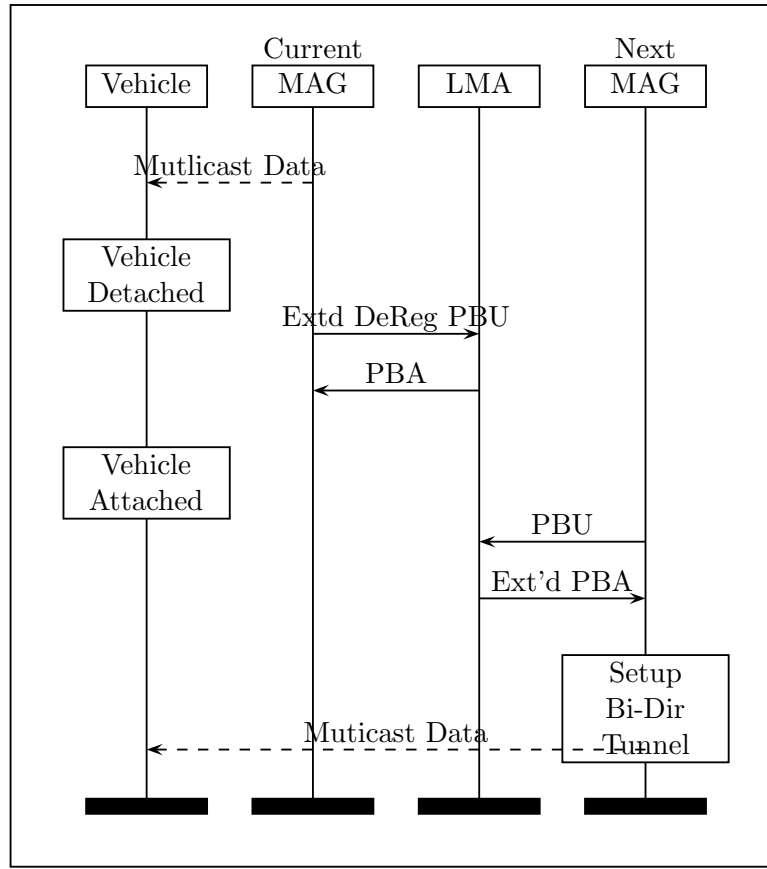


Figure 2.5: Proactive PMIP handover

limited than wired communication, minimizing wireless exchanges reduces latency and performance costs.

Similar to FMIP, PMIP also has a proactive and reactive protocol. These protocols are summarized within Figures 2.5 and 2.6, respectively. In the case of PMIP, the proactive protocol is followed when the current MAG successfully recognizes that the vehicle is about to leave the MAG's signal range. Upon this realization, the current MAG notifies the LMA with an proxy binding update (PBU) message indicating that the vehicle is unregistering with that MAG, and to update the LMA with information on the vehicle. The LMA replies with that MAG, and to update the LMA with information on the vehicle. The LMA replies with a proxy binding acknowledgment (PBA) message to confirm. Once the vehicle has connected with the next MAG, the MAG sends a PBU to the LMA, which then completes the handover process by forwarding the information provided by the current MAG and setting up a tunnel with the next MAG.

In the event the current MAG does not recognize the vehicle disconnecting, the han-

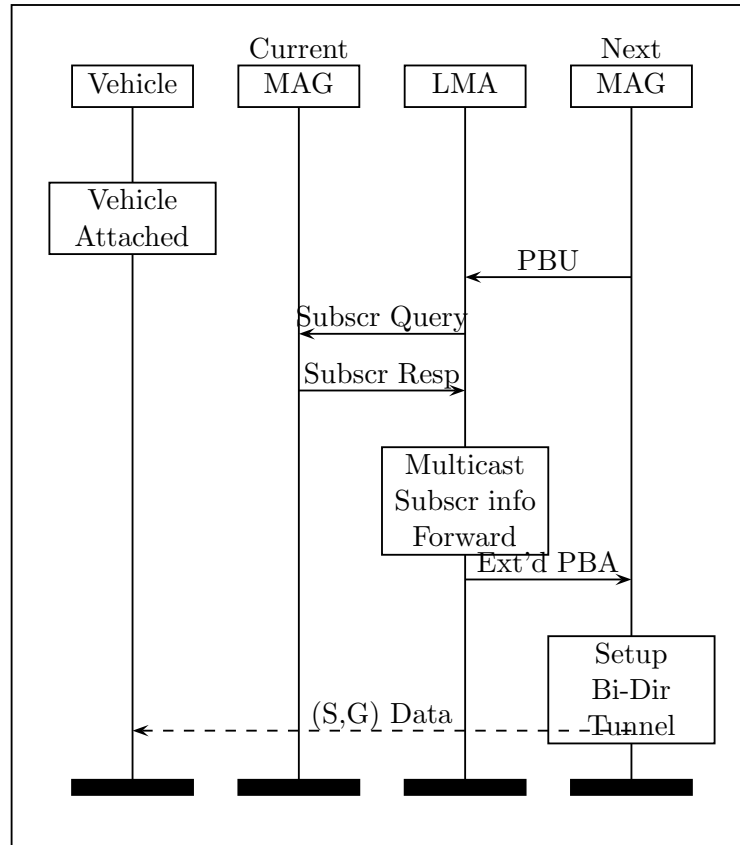


Figure 2.6: Reactive PMIP handover

handover protocol then begins with the next MAG sending a PBU to the LMA. Upon receiving this PBU, the LMA then requests IP information pertaining to that vehicle from the current MAG using the "subscr query" message. Upon receiving the response, it forwards the information to the new MAG and completes the handover. The central issues faced by the PMIP handover are LMA transitions and LMA overhead.

### 2.3.1 Reduce LMA Transition Costs

A problem with the proxy handover is the costly handling of a vehicle moving between the domain of two LMAs. This requires a complete reconfiguration with the new LMA that does not occur until the vehicle has already connected with the new LMA's domain. K. Lee et al. [53] address this issue by introducing intermediate-MAGs, which keep the vehicle connected as it transitions between LMAs. These MAGs are located between two LMA domains, and are connected to both. This allows the vehicle to conduct the handover with

the new LMA while still connected with its original LMA. The vehicle thus never loses connection and the latency cost is reduced to similar values as in intra-domain handovers. A similar approach is proposed by S. Ro et al. [54], who use overlap-MAGs between LMAs, but also take advantage of this overlap period to conduct route optimization. The problem that arises in these approaches is determining the placement of the intermediate-MAGs. To do this requires accurate knowledge of LMA and MAG placement, and understanding of vehicular behavior within the network.

An alternative approach to handling inter-domain transitions is introduced by N. Neumann et al. [55], who propose LMA communication for forwarding IP packets between domains. The first LMA that a vehicle connects to becomes its session mobility anchor (SMA), which acts similarly to an HA within standard mobile IP. The SMA follows standard proxy handover procedure until a vehicle transfers into a new domain. The new LMA then establishes a tunneling route with the LMA for forwarding IP packets directed to that vehicle.

### 2.3.2 Reducing LMA Overhead

With all IP packets traveling through the LMA before being redirected to the appropriate vehicle, there arises the need for route optimization. For example, if two vehicles are physically close to each other, standard PMIP will still have packets routed all the way to the LMA. T. Chiba et al. [56] and L. Villas et al. [113] address this need by proposing multiple route optimization techniques, which are designed to improve PMIP performance by updating circuitous packet routes. One of the proposed methods uses the LMA to recognize if two corresponding vehicles have MAGS close to one another. If true, the LMA notifies the MAGs, which then send the packets directly to each other instead of redirecting to the LMA. The other approaches are a MAG query system, where neighboring MAGs communicate with each other to optimize their route, and a binding-cache system where the LMA sends MAGs regular updates of vehicles within that domain. These systems include both inter-domain and intra-domain systems.

Reducing the amount of data transmission through the LMA is also addressed by H.

Jung et al. [57], who propose multiple MAG communication methods. These methods are similar to the MAG query system of [56], but with an increased focus on removing LMA communication rather than path optimization. Instead of determining the location of a corresponding vehicle by communicating with the LMA, the MAGs communicate with each other through broadcasting messages. The packets are then directly sent between the corresponding MAGs. This removes unnecessary distanced LMA communication, and additionally reduces the load on the LMA that otherwise has to maintain its entire domain.

Instead of increasing MAG communication, S. Son et al. [58] propose removing unnecessary LMA communication by estimating the urgency for LMA communication and changing packet rates accordingly. The urgency value is calculated by the speed at which the vehicle is moving. If a vehicle is moving quickly, they will be committing a handover at much more frequent rate, and so will require more regular LMA checks. This allows removal of unnecessary LMA communication with vehicles that are moving slowly or not at all and that will not switch APs for a longer time. The central issue with this approach is the unreliability of depending on a vehicle's speed. If a slow vehicle moves just on the edge of APs, causing frequent handovers, the lowered LMA communication could cause large delays and packets dropped due to the uninformed LMA. A. Boukerche et al. [109] propose a similar approach to removing unnecessary communication with the vehicles, however suffers from the same issues of unreliable vehicle movement causing errors.

### 2.3.3 Summary

The PMIP handover approach attempts to minimize wireless overhead from the mobile IP handover, reducing network performance costs. However, implementing PMIP causes issues within the wired network, such as LMA transitions and the required network load for tracking each vehicle. Approaches that attempt to resolve these issues often resort to including some wireless communication with the vehicle. This can indicate that a hybrid of PMIP that includes reduced, but not minimal, wireless communication could perform better.

Table 2.6: Proxy Handover Performance Comparison

Article	Handover Latency	Domain Transfer Cost	Overhead	Network Load	Consistency
S. Gundavelli et al. [52] (2008)	Medium	Medium	Low	Medium	Low
K. Lee et al. [53] (2010)	Medium	Low	Medium	High	Medium
S. Ro et al. [54] (2015)	Medium	Low	Medium	High	Medium
N. Neumann et al. [55] (2009)	Medium	Low	High	High	Medium
T. Chiba et al. [56] (2008)	Medium	High	Low	Low	Low
H. Jung et al. [57] (2011)	Medium	Low	Medium	Medium	Low
S. Son et al. [58] (2014)	Medium	Low	Medium	Medium	Low

## 2.4 Hybrid Handovers

In this section, we explore the multiple methods which have been investigated to combine the FMIP with HMIP and PMIP methods and benefit from the performance improvements of both. This is possible since the FMIP operates on the wireless end of operations, and HMIP and PMIP operate within the wired network structure, making them compatible.

### 2.4.1 FMIP and HMIP

R. Hsieh et al. [62] recognize the complimentary nature of FMIP and HMIP, and propose the Seamless handover, which utilizes both approaches. In addition to the two methods working together, the hierarchical approach further amplifies the fast handover's benefits by providing additional time for the fast handover to confirm success through because of the quicker MAP registration procedure. The additional time increases the fast handover's success rate and minimizes the chances for a incorrect failure assumption. A similar approach that utilizes both the MAC layer and MAP is proposed by Z. Zhang et al. [63, 64], who use IP addresses to distinguish between MAP domains, and conduct intra-domain handovers done only with MAC protocol.

The method proposed by R. Farahbakhsh et al. [8] similarly uses both the hierarchical and fast handover, but adds MAP context messaging for the hierarchical handover to take advantage of the fast handover. Upon receiving a MAC trigger, the vehicle initiates its current MAP to forward registration information while the vehicle conducts the handover. The additional time provided by the fast handover, in addition to the context transfer between MAPs, allows a much smoother inter-MAP transition.

A similar idea to [8] is proposed by L. Zhang et al. [9], who propose setting up bi-directional tunnels between APs before the handover occurs. This is done by having the two APs establish a tunneling route when the MAC layer trigger of the FMIP first occurs. The APs can then exchange registration information as the vehicle transitions between them. Additionally, the tunnel reduces packet-drop rates and allows additional time for the vehicle to begin receiving IP packets through the new AP. This method improves both MAP



Table 2.7: HMIP, FMIP, and PMIP Hybrid Handover Approaches

Article	Approach	Advantages	Drawbacks
R. Hsieh et al. [62] (2003)	Uses HMIP and FMIP together	Reduces costs of entire handover	Limited by IP requirements, has additional complexity
Z. Zhang et al. [63, 64] (2013)	Uses MAC for intra-domain, IP for inter-domain	Minimizes inter-domain handover latency	Still suffers in inter-domain handovers, additional MAC management
R. Farahbakhsh et al. [8] (2009)	Adds AP context transfer protocol that utilizes MAC trigger	Further reduces registration latency	High costs and slow recovery if FMIP fails
L. Zhang et al. [9] (2011)	Setup bi-directional tunnels between APs at MAC trigger	Reduced packet drop rate, increased reliability	Increased overhead
L. Zhuang et al. [40] (2011)	Use decision engine for MAP selection	Improved handover performance consistency	Added complexity and overhead
H. Yokota et al. [41] (2010)	Uses FMIP to pre-establish tunneling between MAGs	Reduces packet drop rate and latency	Increased overhead, depends upon vehicle communication
A. Moravejosharieh et al. [42] (2014)	Use GPS threshold triggers to initiate MAG tunnels	Lower packet drop rate and latency	Large error cost and inconsistent
Y. Wang et al. [43] (2009)	Uses MAC trigger to initiate MAG route optimization	Reduces latency from route optimization	Increased overhead and wireless communication
S. Moon et al. [44] (2011)	Neighbor MAGs maintain tunnels, tunnel initiated with FMIP	Reduces latency of registration costs in MAG switches	Higher overhead to maintain tunnels
C. Huang et al. [45] (2015)	Use MAC trigger to initiate route optimization	Reduces latency from route optimization	Increased overhead and wireless communication

inter-domain and intra-domain handovers, thus also increasing reliability. The proposed approach also includes a backup procedure if a fast handover fails. This procedure allows IP packet forwarding between APs immediately after the vehicle establishes connection with the new AP. While the procedure has less performance improvement, it still reduces the delay for IP packet exchanges.

A problem caused by combining the FMIP and HMIP is the increased probability of poor MAP selection caused by an FMIP's incorrect assumption. A solution to this issue is proposed by L. Zhuang et al. [40], who add a decision engine to improve upon MAP selection and increase handover consistency. The decision engine uses the vehicle's mobility and acquired network information to ensure the vehicle connects with the MAP that will provide the best performance. This minimizes the occurrence of false assumptions made by FMIP that otherwise cause reduction to HMIP performance, in addition to general improvement to HMIP performance.

## 2.4.2 FMIP and PMIP

The other hybrid approach is proposed by H. Yokota et al. [41], who introduce the proxy-based fast handover. This implements the fast handover within the proxy architecture to reduce the packet loss and latency costs of PMIP. However, different from FMIP and HMIP, the vehicle is not involved within the PMIP handover. Thus, the FMIP and PMIP require more adjustments to be used together. This is approached within [41] by having the vehicle forward the MAC layer trigger to the AP. The AP then pre-establishes a tunnel with the next MAG instead of the vehicle conducting an early registration protocol. Upon the vehicle connecting with the new MAG, the old MAG can send the new MAG registration information and also begin forwarding IP packets immediately. The tunnel thus reduces the overhead and packet-drop rate of the handover process. A. Moravejosharieh et al. [42] propose a similar method as [41], but use GPS signals instead of MAC triggers. The vehicles send periodic GPS updates to the AP, which then derives which MAG the vehicle is moving towards and how soon the vehicle will switch MAG domains. Using GPS triggers instead of MAC triggers allows for a much earlier handover initiation; however, this method

is inconsistent because its performance relies on vehicles not making any changes to their direction or speed.

A slightly different approach to combining the FMIP and PMIP is proposed by Y. Wang et al. [43], who take advantage of the FMIP to improve the PMIP route optimization. They initiate the routing optimization procedure when the MAC trigger occurs. This way, the route optimization is able to complete by the time the handover occurs. By having the better route set up for the handover, latency is reduced and overall performance is increased. Additionally, the advanced route optimization is used to buffer packets at the new AP to reduce the packet drop rate. An issue of this approach, however, is it does not resolve the previously discussed issues of the FMIP and PMIP. The problems faced of each individual approach is not resolved by the other's benefits.

S. Moon et al. propose the FPMIP-PT [44] to also enhance the fast-proxy mobile IP handover by reducing latency times. The FPMIP-PT involves each MAG pre-configuring tunnels with its neighboring MAGs, separate from the handover. The tunnels between its neighbors are then activated when the handover is initiated, requiring less time and costs than having to establish a new tunnel. This reduces the overall registration latency when MAG changes occur, thus making the handover more consistent.

Instead of focusing on early-initiation of proxy procedures, the method proposed by C. Huang et al. [45] utilizes the fast handover to improve MAG selection. In this approach, the MAC layer information is used by the current MAG to decide the next best MAG for the vehicle to connect to. Since the MAC information arrives early and provides insight into network performances, the MAG can make a well educated selection. In addition, once the new MAG is chosen, the LMA begins multicasting packets to both MAGs. The multicasting reduces the packet drop rate, but also largely increases overhead.

### 2.4.3 Summary

Overall, hybrid approaches improve the handover latency and packet drop costs compared to the single approaches. However, hybrid approaches do suffer from additional complexity and often times additional overhead. There also is the issue that all of these approaches

Table 2.8: Hybrid Handover Performance Comparison

Article	Handover Latency	Packet Drop Rate	Overhead	Network Load	Consistency
R. Hsieh et al. [62] (2003)	Medium	Medium	Low	Medium	Low
Z. Zhang et al. [63, 64] (2013)	Medium	Medium	Low	Medium	Low
R. Farahbakhsh et al. [8] (2009)	Medium	Low	Medium	High	Medium
L. Zhang et al. [9] (2011)	Medium	Low	High	High	Medium
L. Zhuang et al. [40] (2011)	Medium	High	Low	Low	Low
H. Yokota et al. [41] (2010)	Medium	Low	Medium	Medium	Low
A. Moravejosharieh et al. [42] (2014)	Low	Low	Medium	Medium	Low
Y. Wang et al. [43] (2009)	Low	Low	High	High	Low
S. Moon et al. [44] (2011)	Low	Low	High	High	Low
C. Huang et al. [45] (2015)	Low	Low	High	High	Medium

are reactionary and initiate at the time of a transition between APs, resulting in a small time frame to complete the handover procedure before network performance drops. This is particularly problematic when there is high network traffic causing higher packet drop rates and for vehicles that have much smaller transition times between APs.

## 2.5 Predictive Handover

In contrast to the fast, hierarchical, and proxy methods, the predictive handover aims to conduct the process in advance instead of directly modifying the mobile IP architecture. This is done by predicting which AP the vehicle will connect to next, before the vehicle begins transitioning between APs. By knowing the next AP so far in advanced, the handover process can easily be completed ahead of time to provide a smooth AP transition. This resolves the issues faced by the other approaches, as mentioned in Section 2.4.3. The largest issue the predictive approach is the unreliability of the advanced handover, mostly caused by poor AP prediction performance. Methods for handover prediction can be categorized as probability analysis, pattern-matching, and movement projection. Figure 2.7 illustrates the different variables to be considered (statistical and movement) for handover prediction.

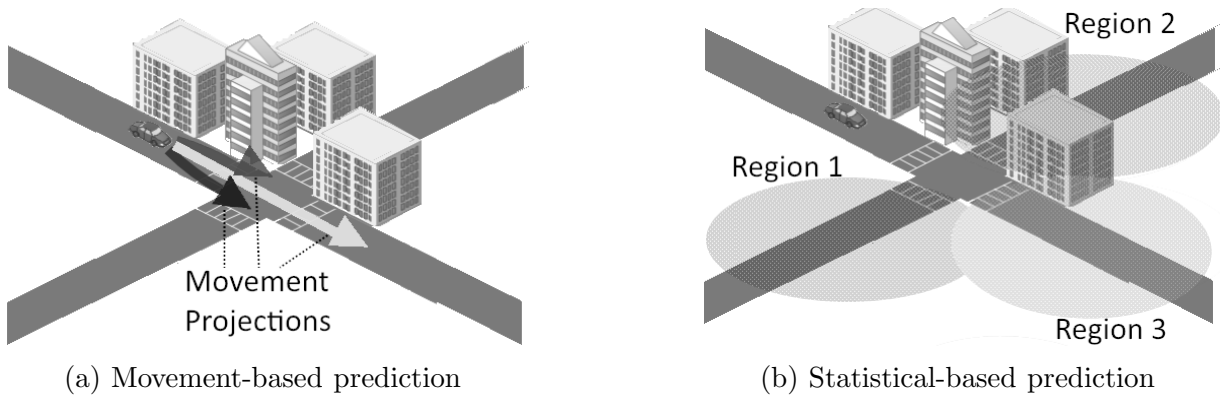


Figure 2.7: Movement and statistical variable differences

Table 2.9: Predictive Handover Approaches (Probability Analysis)

Article	Approach	Advantages	Drawbacks
F. Lassabe et al. [59] (2006)	Use Markov renewal processes to compare vehicle's AP history	Accurate in consistent scenarios	Cannot adjust to new information
H. Kim et al. [14] (2009)	Analyzes vehicle's physical movement history	Physical movement can provide more insight than AP history	Errors when physical history is misleading
M. Kyriakakos et al. [18] (2003)	Adds learning automaton to improve from errors	Improves performance consistency	Suffers from misleading information
Z. Becvar [21] (2009)	Uses AP's handover history	Less likely to have misleading information	Cannot distinguish between vehicles
N.V.D. Wijngaert et al. [16] (2005)	Predict the top 3 most-likely APs	Higher accuracy	Large overhead costs
M. Al Masri et al. [17] (2014)	Predict session activity to determine handover timing	Reduces packet drop rate	Added complexity not justified by small benefits
S. Pack et al. [50] (2004)	Use a handover database for prediction and dwell times	Ensures consistent handovers if dwell time is small	Added overhead and complexity, requires accurate data
I.F. Akyildiz et al. [22] (2004)	Weights vehicle history against AP history	Improved accuracy and consistency	Limited by each method's best performance in a scenario
P. Fazio et al. [48] (2013)	Use vehicle's location history within AP range	Reduced calculation and less misleading	Cannot differentiate similar AP probabilities

### 2.5.1 Probability Analysis

Statistical analysis for predictive handovers most commonly uses a probability modeling or pattern matching technique to determine the next AP. Probability modeling and pattern matching both apply previously gathered information for the prediction. The main differences between these two approaches are what information is gathered and how it is interpreted. Probability modeling considers sample statistical information of vehicle movement for prediction.

One approach to probability modeling is proposed by F. Lassabe et al. [59], who use Markov renewal processes for prediction. The Markov renewal processes are used to model the probabilistic relationship between APs based on what previous APs a vehicle has connected to. First, the Markov renewal processes are trained with a set of sample vehicle data to determine the probability values. After training, the final values are used to calculate a vehicle's most likely next-AP based on what previous APs it has connected to. This approach is expanded upon by A. Boukerche et al. [110], who use the Markov models to predict AP congestion levels. A problem with these approaches is the situation where a vehicle's AP history is misleading. With roads restricting the directions a vehicle can go, it may often have to take indirect routes, which will often cause prediction errors. In addition, various AP coverage methods [91, 101, 103, 104] could lead to AP connections that do not match with vehicle's history.

In an attempt to resolve the issue of misleading history, Z. Becvar [21] reduces the size of the system being predicted. Instead of considering a vehicle's entire AP history, only the vehicle's current AP and its probability relationships to neighboring APs are used. This greatly simplifies the prediction requirements by reducing the probability calculations and the AP memory a vehicle would otherwise maintain. However, a problem that arises in this approach is its inability to distinguish between individual vehicles.

M. Kyriakakos et al. [18] also choose to consider only local AP variables as done in [21], but include the previous AP as well to provide some distinction between vehicles. They also introduce a learning automaton to improve long-term performance by adjusting probability variables. The automaton updates the variables according to a trial-and-error method,

which adds weight to neighbor AP probabilities if the prediction is correct, and removes weight from the predicted AP if it is wrong. The prediction result is retrieved by vehicles informing the learning automaton by communicating over the network. Two learning automatons are used, one for global AP probabilities and one that keeps track of individual vehicle path results. The goal of adding the second automaton is to learn each vehicle's paths, since vehicles are more likely to follow the same path as they have previously. The learning automatons show improvement to accuracy as time passes, but require additional overhead for communication prediction results. In addition to overhead, the second learning automaton requires a large database to maintain probability information for every passing vehicle.

N.V.D. Wijngaert et al. [16] use a similar approach as [21], but extend the prediction to determine the next three most-likely APs. It is shown that predicting the next three APs instead of only one greatly improves the accuracy, partly because a vehicle will most commonly have around that many realistic options. The problem of this method, however, is the large overhead increase. To conduct an early handover with three APs requires a default of two APs to waste resources. Once APs begin reaching saturation in high-density traffic scenarios, this additional overhead will have large performance costs, reducing the benefits of handover prediction.

A separate method to using probability analysis to improve the handover is proposed by M. Al Masri et al. [17]. In their approach, they use a Markov model to probabilistically model the session activity, and then use this information to determine the best moment to conduct the handover. If the handover is conducted when network activity is low, overall performance will be less affected. However, a vehicle's activity is very unpredictable, making this approach unreliable. In addition, there is also the risk of predicting a late handover timing that will cause more performance degradation than the standard handover.

The idea presented by [16] is expanded upon by S. Pack et al. [50], who propose a handover database used for prediction. The database records which neighbor AP the vehicle moves to and the dwell time the vehicle spends within that AP. First, the most likely APs are derived based on the handover frequency, as done in [16]. Next, the recorded



Table 2.10: Predictive Handover Performance Comparison (Probability Analysis)

Article	Processing Cost	Memory Cost	Over-head	Network Load	Consistency
F. Lassabe et al. [59] (2006)	Medium	Medium	Low	Low	Low
H. Kim et al. [14] (2009)	High	Medium	Low	Low	Medium
M. Kyriakakos et al. [18] (2003)	Medium	Medium	Medium	Medium	Medium
Z. Becvar [21] (2009)	Low	Medium	Low	Low	Low
N.V.D. Wijngaert et al. [16] (2005)	Medium	Medium	High	High	High
M. Al Masri et al. [17] (2014)	Medium	Low	Medium	Low	Low
S. Pack et al. [50] (2004)	Medium	High	High	Medium	Medium
I.F. Akyildiz et al. [22] (2004)	High	High	Medium	Low	Medium
G. Yavas et al. [47] (2005)	Medium	Low	Low	Low	Low
P. Fazio et al. [48] (2013)	Low	Low	Low	Low	Low

Table 2.11: Predictive Handover Approaches (Movement Projection)

Article	Approach	Advantages	Drawbacks
E. Hernandez et al. [23] (2004)	Utilizes GPS movements to project next handover	Temporal vehicle information resolves ambiguous probabilities	Difficult to predict sudden movement changes
W. Su et al. [37] (2001)	Predicts the dwell time using vehicle movement	Reduces packet loss and latency from packet forwarding	Difficult to predict sudden movement changes
R. Gunasekaran et al. [38] (2014)	Adds route database to dwell time prediction	Reduces packet loss and packet overhead	Large performance costs if error occurs
F. Fang et al. [26] (2004)	Replaces GPS with network measurements	Remove GPS reliability	Unreliable within a mobile IP network
S. Bhaskar et al. [27] (2015)	Use signal strength measurements to predict	Reduced requirements, improves timing	Unreliable due to inconsistent, noisy measurements
A.S. Sadiq et al. [28] (2013)	Use both network and movement measurements	Improves reliability + network performance	Poor performance if sudden behavior changes
M. Almulla et al. [29] (2014)	Adds turn detection with angle calculation	Improves accuracy and turn detection	Unreliable when sudden changes or misleading behavior
H. Park et al. [46] (2005)	Early handover conducted and managed with HA	Reduces handover packet loss and latency	Large overhead for multi-casting packets, increased costs if prediction error

dwelling times and network information is used to determine how many APs the vehicle should predict. If a dwell time is small for the predicted AP, the algorithm then extends the number of APs for early registration to ensure a smoother connection. By using a more dynamic approach to determining the number of APs for the vehicle to commit to, additional accuracy is attained without also producing too much overhead.

The method proposed by I.F. Akyildiz et al. [22] uses both the handover history method from [21] and the AP history method from [59] in parallel. After calculating the next most likely AP using both methods, the two results are then weighed against each other based on their determined reliability. These reliabilities are calculated based on how confident each method is with the information used in its prediction. The proposed approach thus outputs whichever of the two methods perform better in a specific situation. Since these methods have strengths in different scenarios, an overall improvement to prediction is observed. But, this parallel approach is still limited by the maximum individual performances of the other methods, improving performance by only a small amount.

G. Yavas et al. [47] propose a similar movement matching method to [49], but uses vehicle regional movements instead of trajectories. The method divides the roads into small, discrete segments, and then compares a vehicle's recent segment history to previous vehicle segment history. The approach then assumes the vehicle will next move to the road segments of the closest matching segment path. By dividing the road into discrete segments instead of directly considering a vehicle's continuous movement, the number of possible observations is greatly reduced. This reduces the calculation cost for comparing path-history because of the fewer potential combinations and the simpler math required for comparing discrete values. However, a setback is that the method then does not consider vehicle speed or acceleration, which can often be telling of a vehicle's intentions.

A half-way point of using trajectories and regions is proposed by H. Kim et al. [14], who predict the next AP by considering a vehicle's AP history and its general direction. At each handover, the vehicle's location and AP is recorded. When more than one measurement set is recorded, the angle between the two locations is calculated to determine a vehicle's general projection. The angle information is then compared to previous vehicles who have

connected to the same set of APs. The vehicle is then assumed to next connect to the same APs as the vehicle that has the closest-matching angles. By using a bit of regional history and general direction, some improvement can be seen in prediction performance. Although, this approach suffers from a similar issue as [59], where road restrictions can often cause a vehicle to move in indirect paths. A road that causes the vehicle to move in an indirect path will disrupt the AP to angle pattern matching, producing prediction errors.

P. Fazio et al [48] propose analyzing a series of location measurements while the vehicle is within a single AP, instead of using AP history and direction analysis. They use distributed Markov chains to calculate the next most probable AP based on what path the vehicle is taking. This requires much less calculation than [14], since it only requires local location information. Additionally, there is the benefit of local location information being less misleading than long-term history analysis because temporal data is more likely to represent a vehicle's next movement. Despite these benefits, however, there is the large issue of being unable to differentiate between two APs of similar probabilities. For example, a T intersection can result in two vehicles with the same path, but with different resulting direction.

## 2.5.2 Movement Projection

Movement projection approaches use temporal data related to a vehicle's current movement to predict its future location, opposed to deriving the most probable future movement through considering recorded statistical information. The benefit of this approach is it does not suffer from the issues as a statistical approach. Misleading history is not a problem because only temporal data is considered, and two options with similar probabilities can be distinguished using temporal data that differentiates between these options.

An example of movement projection is proposed by E. Hernandez et al. [23], who use the vehicle's location and velocity to determine its future AP. The next AP is chosen by projecting the vehicle's location and finding the AP that provides coverage to that location. The method also introduces the concept of ghost agents, which are entities that are added

Table 2.12: Predictive Handover Performance Comparison (Movement Projection)

Article	Processing Cost	Memory Cost	Over-head	Network Load	Consistency
E. Hernandez et al. [23] (2004)	Low	Low	Medium	Low	Low
W. Su et al. [37] (2001)	Low	Low	Medium	Low	Low
R. Gunasekaran et al. [38] (2014)	Low	Medium	Low	Low	Low
F. Fang et al. [26] (2004)	Low	Low	Medium	Low	Low
SBhaskar et al. [27] (2015)	Low	Low	Low	Low	Low
A.S. Sadiq et al. [28] (2013)	Medium	Low	Medium	Low	Medium
M. Almulla et al. [29] (2014)	Medium	Low	Medium	Low	Medium
H. Park et al. [46] (2005)	Low	Low	High	Medium	Low

to APs to be used after a prediction is made. These agents reserve resources and initiate an early registration procedure before the handover begins to reduce handover latency times.

W. Su et al. [37] extends the prediction introduced by [23] to also use a vehicle's location and speed to project a vehicle's remaining dwell time for its current connection. Projecting the remaining dwell time is used to determine the handover's timing, thus allowing packet-forwarding to the new AP to begin without requiring a trigger from the vehicle. This reduces packet-loss that would otherwise occur from misdirected packets, and additionally reduce the latency caused by a vehicle waiting for its forwarded packets. An approach that expands upon [37] is proposed by A. Bamis et al. [105], who aim to reduce the processing costs of the projection by categorizing vehicles into mobility classes and determining the dwell time based on their class.

These projection approaches have high prediction accuracy when the vehicle movement is consistent, but begin to suffer large performance consequences in situations where changes in movement are common. Both [23] and [37] are unable to predict if a vehicle is about to turn or rapidly change in speed. Therefore, if one of these events occurs, a prediction error will occur almost every time. For [37], where handover timing is also predicted, an additional spike in packets-dropped will also occur. One attempt to remove these issues is by R. Gunasekaran et al. [38], who propose also cross-referencing the vehicle's route according to a road database. This reduces the number of possibilities for being misled; however, a vehicle that behaves differently from the expected route will suffer very large performance degradation.

F. Fang et al. [26] propose removing the reliability on GPS measurements by deriving the vehicle's movement and the next AP through use of temporal network measurements instead. This is done by the vehicle interpreting its current AP's and surrounding AP's change in signal strengths to estimate its movement. Instead of projecting its specific movement, it predicts what AP it will most likely connect to next based on the projected signal strengths. This is not to be confused with the fast handover, which also uses signal measurements to recognize an upcoming handover. The predictive method takes current signal strength changes and estimates what AP it will connect to next, whereas the fast handover

Table 2.13: Predictive Handover Approaches (Pattern Matching and Hybrid)

Article	Approach	Advantages	Drawbacks
G. Jeney et al. [24] (2009)	Uses database to compare movement to previous vehicles	Better detection of turns and movement changes	High memory and overhead costs
W. Hu et al. [49] (2004)	Compares partial trajectories to database	More consistent, unlikely to be misled by probabilities	Large processing and storage costs for comparing trajectories
G. Yavas al. [47] (2005)	Compare vehicle regional movements to database	Reduced calculation and memory costs	Less reliable without projection consideration
A. Bohlooli et al. [15] (2011)	Matches vehicle's turn choices at intersections	Reduces processing costs, predicts longer segments of vehicle's path	Incorrect predictions have large costs, lack of temporary data reduces accuracy
E.K. Paik et al. [30] (2003)	Weights movement projection against handover history	Small improvement to accuracy	Limited by individual performances, does not justify added complexity
T. Liu et al. [25] (2002)	Projection for short-term prediction, and regional matching for long-term	Projection improves pattern choosing, consistency improved	Accuracy still limited by individual approaches' accuracies

waits for the AP change to already begin occurring. The problem of this approach, however, is the signal interference of the environment can easily cause noise problems much worse than GPS measurements. This noise can then cause prediction errors when the method tries to project the signal changes. In addition, AP coverage ranges are relatively small, often leaving the vehicle to not have enough data to determine its next AP.

The method proposed by S. Bhaskar et al. [27] also utilize network measurements to project the next AP. However, the method does not attempt to derive the physical movement with the signal; instead, they analyze the changing signal strengths to determine the next AP. This removes the requirement of having multiple nearby APs and reduces the calculation costs for prediction, but maintains enough information for predicting the handover timing. The consequence of this approach is it amplifies the noise issues also faced by [26]. Since it analyzes fewer signals than [26], disruptions in the signal will cause more regular and more disruptive prediction errors. In addition, the dependency on fewer signals is less reliable due to the natural inconsistency of a signal reading. One approach that attempts to improve upon this is proposed by A. Boukerche et al. [88,96], who propose an event-driven and query-based protocol to ensure a high delivery of packets.

A.S. Sadiq et al. [28] introduce an approach for using both signal strength and movement projection to predict the next best AP. This approach includes the use of a vertical handover with an intelligent network selection scheme to resolve the issues of the shorter AP transmission ranges. However, the used projection approach does not consider potential sudden changes in the vehicle's behavior. M. Almulla et al. [29] aimed to resolve this issue by adding a turn detection scheme that examines a vehicle's movement angles to determine if a turn is about to occur. This method improves recognition of a vehicle turning, but still suffers when abrupt movement changes occur.

The method proposed by H. Park et al. [46] uses a similar GPS-based prediction approach as [23], but has a different protocol for handling the prediction. Once the next AP has been predicted, the vehicle uses binding update packets, similar to the fast handover, to notify the next AP and prepare the handover in advanced. The binding update is sent to the HA, which then establishes a connection with the next AP and multi-casts the packets



Table 2.14: Pattern Matching and Hybrid Performance Comparison

Article	Processing Cost	Memory Cost	Over-head	Network Load	Consistency
G. Jeney et al. [24] (2009)	High	High	Medium	Low	Medium
W. Hu et al. [49] (2004)	High	High	Low	Medium	Medium
G. Yavas et al. [47] (2005)	Medium	Low	Low	Low	Low
A. Bohlooli et al. [15] (2011)	Low	Low	Low	Low	Low
E.K. Paik et al. [30] (2003)	Low	Low	Medium	Low	Medium
T. Liu et al. [25] (2002)	High	Medium	Low	Medium	Medium

to both the old and new AP. After the vehicle connects with the next AP, a notification is sent to the HA to stop sending packets to the old AP. This way, the packets dropped during the handover are greatly reduced, and the handover latency is reduced. However, this approach requires large overhead for the multi-casting, which can potentially occur for extended periods of time if the prediction occurs prematurely.

### 2.5.3 Pattern Matching and Hybrid

Pattern matching and hybrid approaches are discussed together since pattern matching uses either probability analysis or movement projection as the patterns. Pattern matching records individual vehicle movements and then matches the current vehicle's movement to the closest related recorded movement. The approach then assume the current vehicle will continue to follow the recorded related pattern. Pattern matching can often provide more insight into a vehicle's future movement, but has higher calculation costs than probability modeling approaches. An example of pattern matching is the method proposed by W. Hu et al. [49], who match partial trajectories to determine a vehicle's future location. At each time interval, a trajectory is calculated based on the vehicle's current movement.

After a series of trajectories are collected, they are compared to a database of previous vehicle trajectories and the most common paths related to those trajectories. A problem of this approach, however, is the high cost of comparing a vehicle's trajectories to an entire database when there are time-restrictions to provide an output. Also, the large variety of potential vehicle movements can make it difficult to match trajectories to a specific pattern.

G. Yavas et al. [47] propose a similar movement matching method to [49], but use vehicle regional movements instead of trajectories. The method divides the roads into small, discrete segments, and then compares a vehicle's recent segment history to previous vehicle segment history. The approach then assumes the vehicle will next move to the road segments of the closest matching segment path. By dividing the road into discrete segments instead of directly considering a vehicle's continuous movement, the number of possible observations is greatly reduced. This reduces the calculation cost for comparing path-history because of the fewer potential combinations and the simpler math required for comparing discrete values. However, a setback is that the method then does not consider vehicle speed or acceleration, which can often be revealing in regard to a vehicle's intentions.

G. Jeney et al. [24] compare a vehicle's GPS measurements to a database of previous to predict the next AP. A series of GPS measurements recorded by the vehicle are sent over the network to be compared to a database. The database is then used to determine the closest matching vehicle, and return the next most-likely AP. In addition to the GPS prediction, the vehicles also record network information that is stored within the database. This includes dwell times, signal-to-noise ratios, and handover timing. By also having access to the network information, the handover timing can be predicted, and packets can be forwarded appropriately to further reduce handover costs.

Instead of using GPS measurements, A. Bohlooli et al. [15] take advantage of the road movement restrictions in their pattern matching. A vehicle can only move straight except at road junctions, where there is then turn possibilities. Thus, the method uses vehicle turning directions and the length of road segments for prediction. A vehicle's turning sequence is recorded and compared to previously recorded turn sequences. The

closest-matching sequence is then used to predict the vehicle's future turn decisions. With the additional information on road lengths, the vehicle's path between road junctions is predicted with minimal calculation. The method proposed by R. Batista et al. [97] also uses sequences matching, but uses a z-align method to optimize the pattern matching performance. The benefit of this approach is it provides details on a vehicle's entire path with only having to do pattern comparisons at each intersection. But, an incorrect turn prediction will lead to large errors in path prediction. Additionally, this approach also suffers from not considering individual vehicle speed and acceleration. This makes two similarly-probable path options difficult to distinguish between.

Statistical analysis and movement projection prediction approaches have shown to perform accurately under certain circumstances and also have problematic scenarios. The strengths and weaknesses compliment one another, providing. One example of using these approaches together is found in E.K. Paik [30], who conducts movement projection and probability individually, then weighs them against each other based on a confidence variable. They also introduce a threshold value to ensure a prediction is not made based on faulty network data. The threshold represents the distance the vehicle should travel before making a prediction, otherwise, signal interference could cause the vehicle to prematurely output a prediction. By having the vehicle wait a certain distance, enough information is acquired to produce a more reliable prediction. Although a major issue of having a distance threshold is if a vehicle changes direction in an unexpected way. This could result in a much shorter distance till the handover, and the threshold preventing any prediction from being made.

T. Liu et al. [25] use movement projection for short-term prediction and a pattern-matching cell prediction for long-term prediction. First, the movement projection is used to determine the next most-likely AP by considering the vehicle's current movement while within the area of an AP. After the vehicle has moved through multiple APs, its overall AP pattern is compared to previous vehicle AP history. This is used to determine a more generalized projection of the vehicle's movement. Determining this generalized path is done with a similar learning method described by A. Boukerche et al. [87, 108] These

two operations are conducted separately, as done in [30], except the pattern matching approach will sometimes use the movement projection's predicted AP to help differentiate between similar pattern predictions. By including some overlap between the two prediction methods, the prediction is more informed and accuracy is improved.

### 2.5.4 Neighbor Discovery

Beyond addressing prediction accuracy, other aspects of the predictive handover have also been investigated in literature. One such aspect is discovering neighbor AP information, which is required for the vehicle to begin an early handover registration before actually reaching the AP. A popular approach for discovering neighbors was developed by A. Mishra et al. [19], who introduce neighbor graphs that are populated through the use of context message communication between vehicles and APs. Whenever a vehicle conducts a handover between two APs, it sends two context messages: the first is sent to its new AP with information on its previous AP, and the second is sent to the previous AP providing information on its new AP. The APs are then able to find out about neighboring APs that a vehicle may potentially connect to next. A large benefit of this approach is it only relies on the network information. This allows the neighbor graphs to be built without outside influence, and still allows APs to discover neighbors that are potentially far away. A different method for neighbor discovery is proposed by A. Boukerche et al. [98], who have the nodes send out context messages over the network to find one another. The issue with this approach, however, is the additional network load required to conduct the process.

An example prediction approach that uses the neighbor discovery method is proposed by S. Hadjiefthymiades et al. [20], who also introduce a datagram relocation coordinator to conduct the prediction and manage resources. The coordinator utilizes the vehicle's past movement and the neighbor graphs to predict the most likely APs for the handover. The coordinator then initiates packet buffering at these APs, minimizing packet loss. By having neighbor information and predicting multiple neighbor APs, the approach provided a reliable improvement to packet loss. However, the approach requires high overhead to buffer packets at multiple neighbors, and begins to experience large performance degradation as

the vehicle density is increased.

### 2.5.5 Summary

Literature on predictive handovers has mostly focused on proposing methods to improve the prediction accuracy. Using probability analysis, movement projection, or pattern matching, the methods attempt to determine which AP the vehicle will connect to next. However, issues remain within each of these approaches, which are often overlooked depending on the mobility model [95, 99, 107] the prediction method is tested against. First, probability analysis experiences difficulty when two AP options have similar probabilities due to its inability to differentiate between the vehicles. Next, movement projection fails when there are sudden turns, as it is unable to consider a probability of a vehicle changing direction. Last, pattern matching requires very high complexity and calculation requirements to maintain and compare many different patterns, while still experiencing, to a lesser extent, the issues of the other approaches.

## 2.6 Further Improving Mobile IP

As mobile technology expands, the demand for improved Internet connectivity increases. Mobile IP is the most widely deployed approach for providing IP services to wireless devices, however the mobile IP handover is an expensive process with performance costs too great for such environments as vehicular networks. A large amount of research has been committed to resolving the handover issues such as HA communication, AP selection, and packet overhead. In this chapter, we categorized this research into the hierarchical, fast, proxy, and predictive handovers, and presented these approaches found within the literature of mobile IP research. We also provide discussion of the benefits and drawbacks of each approach, and their relation to one another.

While these approaches have exhibited improvements to the standard mobile IP handover, each method still has drawbacks that prevent it from providing a smooth AP tran-

sition. Possible directions for future research to attain a handover with low latency and packet drop rate include:

**Addressing Inconsistencies** All approaches suffer from inconsistencies that reduce the performance reliability. In HMIP, MAP transitions remain a point of issue. FMIP faces the unreliability of using MAC layer communication, and the possibility of assuming the wrong AP for advanced handover. PMIP suffers consistency issues similar to both HMIP and FMIP, due to it having LMA transitions and relying on MAGs to accurately detect AP transitions. Of all approaches, the predictive handover suffers the largest inconsistencies due to the difficulty of accurately predicting the next AP in advance. Providing a reliable handover is particularly important for fast moving vehicles, where the frequency of AP transitions largely increases the possibility of handover failures.

**Improving Scalability** Some approaches have achieved greatly reduced latency with consistency, but at the cost greatly increasing overhead. This works if there is a very low density of mobile nodes, but APs will reach saturation quickly, and performance quickly drops as node population increases. Potential exists in researching scalable approaches to these methods, which would then be able to provide smoother AP transitions in any condition.

**Further Exploration of Hybrid Approaches** Hybrid approaches have shown potential for resolving current mobile IP issues, as different methods are often complimentary to one another. However, there still are many possible combinations that have not been fully explored. For example, predictive methods being utilized with HMIP, FMIP, or PMIP approaches.

# Chapter 3

## Motivation

In this chapter, the reasoning behind the proposed approach is explained in terms of the issues faced by the related literature. We propose our predictive handover method, and discuss how this solution addresses the existing problems.

The central problem of non-predictive handover approaches is the requirement of mobile IP mandating a minimum performance cost. This cost is still higher than IP service requirements, and is an unaddressed problem in both fast and hierarchical approaches. Both approaches are initiated when the vehicle begins its transition between access points, which limits improvement in performance to the minimum cost [32]. If the process is conducted in advance, the required steps can be completed before performance degradation occurs. However, initiating a handover early requires a correct prediction of the AP to which the vehicle will connect to next.

Accurately predicting a handover requires a robust method capable of handling a variety of traffic and road scenarios, while also differentiating between the intentions of individual vehicles. Common and effective variables used by the approaches discussed in Chapter 2 include vehicle movement and historical probability. However, the problem with these variables is that each only performs well in specific scenarios, which can result in an unreliable prediction. If the variables can be utilized to combine their benefits while also resolving each other's problems, a more reliable prediction can be achieved. This is illustrated and further explored in Appendix A.

Our proposed method uses a combinational approach to resolve problematic scenarios, achieved by probabilistically determining a vehicle's most likely future AP through observation of its current movement. In combining the probabilistic and temporal data, accuracy is maintained in a wider variety of situations, thus improving reliability. This is implemented through the use of an online hidden Markov model and a Kalman filter, both having low calculation costs to ensure that the prediction is efficient for the small time frame required by the advanced handover. The HMM is used to model the relational probability distributions between the APs based on observations of vehicle movements, and the Kalman filter is used to track the vehicle's location, velocity, and acceleration for movement projection.

HMMs have proven to be effective for learning system probabilities, but are particularly susceptible to noise and are not designed to handle multi-variable observations [76]. The Kalman filter's approach to smoothing out noisy input, in addition to tracking the multiple variables of vehicle movement, compliments the HMM restrictions, further improving learning and prediction performance. By first using the Kalman filter to refine and interpret the incoming measurements to projections, the HMM can then accurately determine the probability distributions. Attempting to use a standalone HMM otherwise will result in unreliable behavior. In addition, the Kalman filter's covariance estimation provides further refinement by filtering out unreliable inputs with high noise levels. Noisy observations could otherwise negatively alter the probability distributions from misleading data. Observation filtering is also used in the prediction process to prevent outlying data from causing an inaccurate prediction output.

The handover prediction process can be broken down into three components: projecting the vehicle's movement with the Kalman filter, calculating the AP probabilities using the HMM, and deriving the most likely AP using a probability threshold. Probability calculations are performed using a modified online expectation-maximization (EM) learning algorithm for adapting to environmental changes. Together, these components provide a low-cost prediction method that effectively considers probability and movement, producing an informed prediction for improving reliability in otherwise problematic situations.



# Chapter 4

## System Modeling

The prediction approach introduced in the previous chapter is mathematically modeled by defining the system variables and deriving the associated probabilities. These models are then used in Chapter 5 for the predictive handover. First, the hidden Markov model matrices in (4.1), (4.2), and (4.3) are derived, in which neighbor AP probabilities represent the hidden states, and vehicle projections represent the observations. This is followed by the derivation of the online learning method, defined by equations (4.13) to (4.15), and the initial HMM estimation method, defined by equations (4.17) to (4.20), which are based on the derived HMM and the mobile IP environment. Finally, the Kalman filter is modeled for vehicle movement projection.

### 4.1 Hidden Markov Model

We model the HMM matrices defined in equations (4.1), (4.2), and (4.3), based on the system shown in Figure 4.1a. These matrices represent the probabilistic relationships of the system variables, derived to be used for learning and AP prediction. In our system, we consider that there are a total of  $N$  neighboring APs and  $M$  possible observations. The definitions of the variables are provided below.

$$\phi(0) = \left( \pi_1 \quad \pi_2 \quad \pi_3 \quad \dots \quad \pi_N \right) \quad (4.1)$$

$$A = \begin{pmatrix} a_{11} & a_{12} & a_{13} & \dots & a_{1N} \\ a_{21} & a_{22} & a_{23} & \dots & a_{2N} \\ a_{31} & a_{32} & a_{33} & \dots & a_{3N} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ a_{N1} & a_{N2} & a_{N3} & \dots & a_{NN} \end{pmatrix} \quad (4.2)$$

$$B = \begin{pmatrix} b_{11} & b_{12} & b_{13} & \dots & b_{1M} \\ b_{21} & b_{22} & b_{23} & \dots & b_{2M} \\ b_{31} & b_{32} & b_{33} & \dots & b_{3M} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ b_{N1} & b_{N2} & b_{N3} & \dots & b_{NM} \end{pmatrix} \quad (4.3)$$

Let  $y_t$  be the movement observation of the vehicle at time  $t$  and  $x_j$  be the vehicle resulting in connection with the  $j^{th}$  neighboring AP. The relational probability between the observation  $y_t = k$ , and the vehicle being in state  $x_j$ , is denoted as  $b_{jk}$  where  $\sum_{k=1}^M b_{jk} = 1$ . Matrix  $B = \{b_{jk}\}$  represents all movement-to-AP relational probabilities, defined as

$$b_{jk} = P(y_t = k | x = j) \quad 1 \leq j \leq N; \quad 1 \leq k \leq M \quad (4.4)$$

where  $P(y = k | x = j)$  represents the probability of both the observation being  $k$  and the state being  $j$ . There also exists a relational probability between the vehicle's previous state,  $x_{t-1}$ , and its current state,  $x_t$ , that represents any change in the most likely neighbor. Let  $a_{ij}$  represent the probability of the occurrence of  $x_{t-1} = i$  and  $x_t = j$ , where  $\sum_{i=1}^N a_{ij} = 1$ . Note that there is also the probability that the state does not change from time  $t - 1$  to  $t$  and  $i = j$ , therefore requiring  $N$   $a_{ij}$  values instead of  $N - 1$ . The transition matrix  $A$  representing all  $a_{ij}$  is described as

$$a_{ij} = P(x_{t-1} = i | x_t = j) \quad 1 \leq i \leq N; \quad 1 \leq j \leq N \quad (4.5)$$

The matrices  $A$  and  $B$  are used to update the total probability of being in state  $i$  at time  $t$ , which is further explained in Section 4.2. The probability for being in state  $i$

at time  $t$  is represented by  $\gamma_i(t)$ , where  $\sum_{i=1}^N \gamma_i = 1$ . The set  $\phi = \{\gamma_i(t)\}$  represents the probabilities for all states where

$$\gamma_i(t) = P(x_t = i | y_t, \theta) \quad 1 \leq i \leq N \quad (4.6)$$

$\theta$  represents the HMM containing matrices  $A$ ,  $B$ , and  $\pi$ , where  $\pi$  is the initial distribution of  $\phi(0)$  used to determine  $\phi(1)$  with the first observation  $y_{t=1}$ . This distribution is defined as

$$\pi_i = P(x_1 = i) \quad 1 \leq i \leq N \quad (4.7)$$

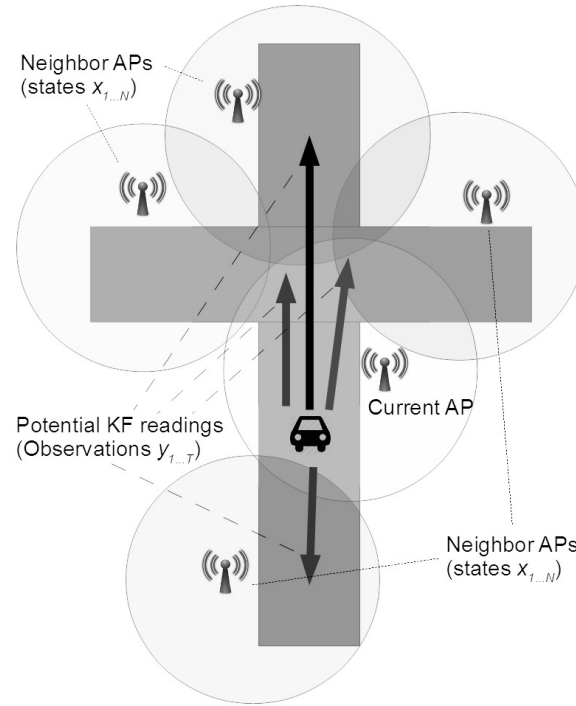
The probabilities of equations (4.4), (4.5), and (4.7) compose the HMM for predicting the most likely neighbor AP. The relationship between these probability values and the prediction variables, visualized in Figure 4.1b, can then be represented by the matrices in (4.1), (4.2), and (4.3).

## 4.2 HMM Learning

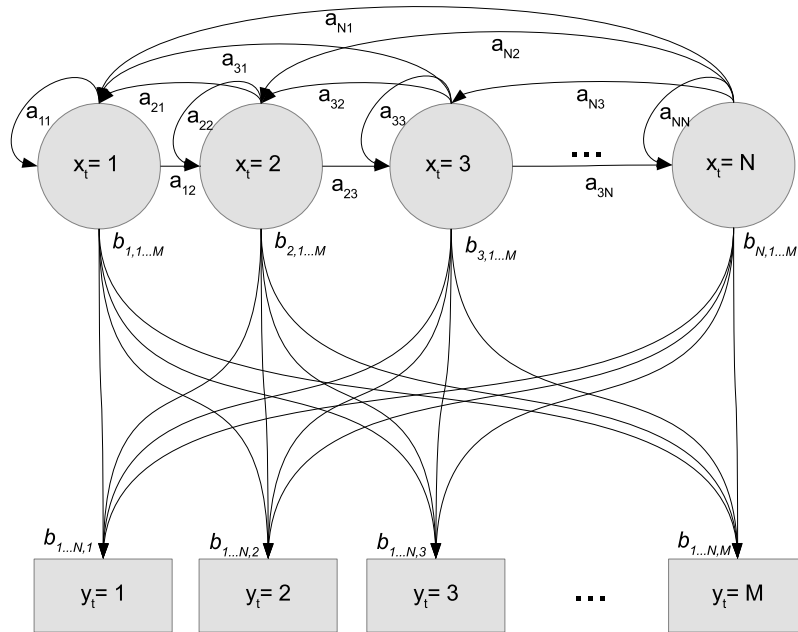
We derive a learning method to accurately determine the probability values based on system observations, while also functioning effectively within the mobile IP network. Our learning method has two main components: an online learning method and an initial matrix estimation. From our analysis, we derive that the AP probability at time  $t$  is determined by equation (4.8), and the AP's initial value is based on equation (4.9). The variables defining the online method are detailed in Section 4.2.1, and the variables describing the initial HMM method are explained in Section 4.2.2.

$$\gamma_j(t) = \frac{\gamma_j(t-1) \times a_{x_{t-1},j}(t-1) \times b_{j,y_t}(t-1)}{\sum_i^N a_{x_{t-1},i}(t-1) \times b_{i,y_t}(t-1) \times \gamma_i(t-1)} \quad (4.8)$$

$$\pi_i = \frac{w(i) \times \delta(i, x_a)}{\sum_{j=1}^N \pi_j} \quad (4.9)$$



(a) System variables considered in the prediction



(b) Relational HMM probabilities between  $N$  states and  $M$  observations, described by matrices  $A$  and  $B$

Figure 4.1: Deriving the hidden Markov model

An online symbol-wise expectation-maximization algorithm, based on the method proposed by Mongillo et al. [69], is chosen for the learning method. EM is chosen because it has been shown to provide more accurate state estimates than other online learning methods, has low memory requirements, and is easy to implement. B. Anderson et al. [75] discuss this further in their recent survey of online learning. Additionally, other methods, such as minimum divergence and minimum prediction error, utilize metrics that add complexity and distract from the observation-focused prediction of the proposed method. Between block-wise algorithms that update the HMM with a set of  $T$  observations, and symbol-wise algorithms that update every observation, a symbol-wise algorithm is chosen for its better fit memory and time complexity. Block-wise methods generally have a time complexity of  $O(N^2T)$  and a memory complexity of  $O(NT)$ , while symbol-wise methods have a time complexity of  $O(N^4)$  and a memory complexity of  $O(N)$  [70]. In the mobile IP network, the number of neighbors is relatively small with  $N \approx 8$ , while the number of observations is much larger, and will differ according to the vehicle's speed and movement. Therefore, the additional calculation of the symbol-wise method is minimal compared to the memory saved; it also provides updated  $\gamma$  values at each time step, which improves the prediction response.

The recursive algorithms that compose the chosen online learning method are adjusted in this thesis according to the mobile IP environment. In addition to online learning using observations, the method takes advantage of the mobile IP network to observe state information after the prediction is made, which facilitates the update of initial probability values over time. The manner in which this state information is obtained for initial variable calculation is explained in Section 6, as are the details of the handover protocol.

The offline EM algorithm is first derived and then converted to an online incremental system. The EM method is based on inferring a probability distribution through maximizing the likelihood of a series of observations. In the case of an HMM, this is achieved by using the Baum-Welch equations. With a total of  $T$  observations and initially estimated parameters  $\theta = (A, B, \pi)$ , the values of matrices  $A$  and  $B$  are determined with the following equations:

$$a_{ij} = \frac{\sum_{t=1}^T P(x_{t-1} = i, x_t = j | y_{1...T}, \theta)}{\sum_{t=1}^T P(x_{t-1} = i | y_{1...T}, \theta)} \quad (4.10)$$

$$b_{jk} = \frac{\sum_{t=1}^T P(x_t = j, y_t = k | y_{1...T}, \theta)}{\sum_{t=1}^T P(x_t = j | y_{1...T}, \theta)} \quad (4.11)$$

These values are then used to estimate the temporary state probabilities at time  $t$ , represented by  $\phi(t) = \{\gamma_i(t)\}$ , as calculated with equation (4.12) where  $a_i = \sum_{j=1}^N a_{ij}$  and  $b_i = \sum_{k=1}^M b_{ik}$ .

$$\gamma_i(t) = \frac{a_i(t) \times b_i(t)}{\sum_{j=1}^N a_j(t) \times b_j(t)} \quad (4.12)$$

### 4.2.1 Online Learning

The goal of our online method is to improve the learning convergence rate while minimizing performance costs. The offline method necessitates storing the observation sequence,  $\{y_1...y_T\}$ , and waiting for a full batch of information before beginning calculations. To remove these requirements in an online symbol-wise method, the equations are altered to be recursive, relying only on the previous time step to update each incoming observation. First, the original probabilities at  $t = 1$  are determined in order for the recursive functions to operate. The original state estimates  $\pi_{1...N}$ , defined by equation (4.7), are used to calculate  $\phi(1)$  as done with

$$\gamma_i(1) = \pi_i \times b_i(y_1) \times a_i(1) \quad (4.13)$$

where the values of  $\pi_i$ ,  $b_i(y_1)$ , and  $a_i(1)$  are determined using the method described below in Section 4.2.2. After calculating  $\phi(1)$ , equation (4.8) is used for each incoming  $y_t$ , where  $\gamma_i(t-1)$ ,  $a_{ij}(t-1)$ , and  $b_{jk}(t-1)$  are the estimated HMM parameters from the previous time step, and  $x_{t-1}$  is the estimated state at  $t-1$ . With  $\phi$  updated to the current time step, matrices  $A$  and  $B$  are then recalculated using  $x_t$ ,  $y_t$ , and  $\phi(t)$  values and the equations

$$a_{ij}(t) = \frac{a_{ij}(t-1) + \delta(x_{t-1}, i) \times \delta(x_t, j) \times \gamma_j(t)}{\sum_{n=1}^N a_{in}(t-1) \times \gamma_n(t)} \quad (4.14)$$

$$b_{jk}(t) = \frac{b_{jk}(t-1) + \delta(x_t, j) \times \delta(y_t, k) \times \gamma_j(t)}{\sum_{i=1}^N b_{ik}(t-1) \times \gamma_i(t)} \quad (4.15)$$

which are derived by using equation (4.12) to update equations (4.10) and (4.11) into incremental calculations of  $a$  and  $b$ , where  $1 < i < N$ ,  $1 < j < N$ ,  $1 < k < M$  and  $\delta(l, m)$  is the Kronecker delta, defined as

$$\delta(l, m) = \begin{cases} 1 & l = m \\ 0 & l \neq m \end{cases} \quad (4.16)$$

This equation ensures the learning process updates the correct values. The learning process, composed of equations (4.8), (4.14), and (4.15), is used to update HMM  $\theta$  for each received observation. These equations, and the resulting  $\phi(t)$  matrix, are used in the AP prediction process described in Chapter 5.

### 4.2.2 Initial HMM Estimation

An initial HMM estimation method is added to improve the accuracy and convergence rate of the learning process. Initial HMM matrix values in  $A$ ,  $B$ , and  $\pi$  are normally set according to a probability distribution that is loosely related to the real system. In our case, the mobile IP network is used to determine and update the initial matrix values, done by packet transmissions that include state, movement, and prediction information. The equations are derived from the online learning method to update the initial HMM values according to the information provided. Since the state is observed, a weight value is used instead of likelihood calculation. The weight value  $w(i)$  for the  $i^{th}$  packet is calculated by

$$w(i) = \begin{cases} 1 & i = 0 \\ \frac{w(i-1)}{w(i-1) + 1} & 1 \leq i < I \\ 1/I & i \geq I \end{cases} \quad (4.17)$$

where  $i$  is correlated to the sequential arrival number of that packet, and  $I$  is a maximum sequential number to prevent  $w(i) \rightarrow 0$ . Maintaining a minimum  $w(i)$  ensures continuous variable adjustment when different probability distributions may occur due to environmental changes. The matrices of the HMM are then updated similarly to the equations in (4.14) and (4.15), but replacing  $\gamma_i$  with  $w(i)$  to calculate the new values based on the observed state instead of the most likely state. This is done as according to

$$a_{ij} = \frac{a_{ij} + \delta(i, x_p) \times \delta(j, x_a) \times w(i)}{w(i) + \sum_{n=1}^N a_{in}} \quad (4.18)$$

$$b_{jk} = \frac{b_{jk} + \delta(j, x_a) \times \delta(k, k_f) \times w(i)}{w(i) + \sum_{i=1}^N b_{ik}} \quad (4.19)$$

where  $x_p$  is the predicted access point,  $x_a$  is the resulting AP, and  $k_f$  is the recorded Kalman filter output at the time of prediction. Note that the value of  $N$  is not constant during early iterations since it increases with the discovery of new neighbors, as discussed in Section 6.1.

Overlapping AP coverage, caused by certain AP deployment schemes, could cause a vehicle to have multiple candidate APs that are indistinguishable within the HMM. This scenario is assumed to have occurred when multiple vehicles report similar movements, but connect to different APs, and transition probabilities between these two APs approach 0.5. The value of 0.5 indicates that half of the predictions between the two APs resulted in the vehicle moving to the other AP. To differentiate between them, network metrics throughput and dwell time are added to the weight calculation. We consider a vehicle moving to  $AP_j$ , reporting a dwell time of  $t_d(j)$  and average throughput of  $th_{kbps}(j)$  while within  $AP_j$ . The new weight,  $w_j$ , is calculated with equation (4.20), where  $J$  is the total number of overlapping APs and  $w_{init}$  is the original weight calculated in equation (4.17).



$$w_j = J \times w_{init} \times \frac{th_{kbps}(j) \times t_d(j)}{\sum_{i=1}^J th_{kbps}(i) \times t_d(i)} \quad (4.20)$$

Thus, the weights are adjusted based on the performance ratio between the overlapping APs. This will result in the prediction method choosing the AP that is most likely to provide better network performance.

### 4.3 Movement Projection

Movement projection is orchestrated using a Kalman filter. The filter is modeled to track and project the vehicle's movement based on incoming GPS measurements, which is then used by the learning and prediction methods for probability calculations. The final vehicle state values and noise estimation are described by equations (4.21) and (4.22), respectively.

$$x_k = x_k^- + K_k(z_k - Hx_k^-) \quad (4.21)$$

$$P_k = (I - K_k H)P_k^- \quad (4.22)$$

These variables are obtained by smoothing incoming GPS measurements and providing an accurate estimation of the vehicle's location, velocity, and acceleration. Measurements are smoothed by first projecting both the vehicle's movement and signal noise, then comparing these projections to the GPS reading. The movement observation at time  $k$  is described as the vehicle's position ( $r_k$ ), velocity ( $v_k$ ), and acceleration ( $a_k$ ), which are maintained in state matrix  $x_k$ . These values are initially estimated from time  $k - 1$  with equation (4.24), where  $A$  is the transitional matrix derived from kinematic equation (4.23),  $\Delta t$  is the size of the time interval from  $k - 1$ , and  $x_k^-$  is the state estimation before GPS consideration. To simplify the explanation, our definition of these values show only one dimension of the two-dimensional tracking.

$$x_t = x_{t-1} + v * t + 0.5 * a * t^2 \quad (4.23)$$

$$x_k^- = Ax_{k-1} \quad x_k = \begin{pmatrix} r_k \\ v_k \\ a_k \end{pmatrix} \quad A = \begin{pmatrix} 1 & \Delta t & 0.5\Delta t^2 \\ 0 & 1 & \Delta t \\ 0 & 0 & 1 \end{pmatrix} \quad (4.24)$$

Before weighing the estimated state  $x_k^-$  against the GPS measurement, we estimated the noise level to predict the accuracy of the incoming GPS measurement. The error estimation,  $P_k^-$ , of actual error,  $P_k$ , is found with equation (4.25), where  $A$  is the matrix from equation (4.24) and  $Q$  is the initial covariance estimate at  $t = 0$ .

$$P_k^- = AP_{k-1} + Q \quad (4.25)$$

The estimated error  $P_k^-$  is then used to determine the weight between  $x_k$  and GPS measurement  $z_k$ . This weight is denoted as the Kalman gain,  $K_k$ . This is calculated with equation (4.26), where  $R$  is the covariance and  $H$  is the scaling matrix between  $x_k^-$  and  $z_k$ .

$$K_k = P_k^- H^T (H P_k^- H^T + R)^{-1} \quad (4.26)$$

The results from equations (4.24)-(4.26) are used in equation (4.21) to determine final state estimation  $x_k$  from the noisy measurement  $z_k$ . The estimated noise is then updated using equation (4.22) for the next time step  $k + 1$ , where  $I$  represents an identity matrix.

The modeled Kalman filter for the defined system is tested to ensure its precision for tracking a vehicle. A scenario was created to test performance in extreme conditions where an acceleration range of  $25m/sec^2$  is used with velocity values reaching  $150m/sec$  and noise values ranging from  $-20m$  to  $20m$ . The Kalman filter is looped at intervals of  $1.0sec$ . Functionality in the extreme scenario ensures the Kalman filter's reliability when applied to normal vehicle movement. The results are shown in Figure 4.2a which

illustrates the Kalman filter's output versus the actual location.

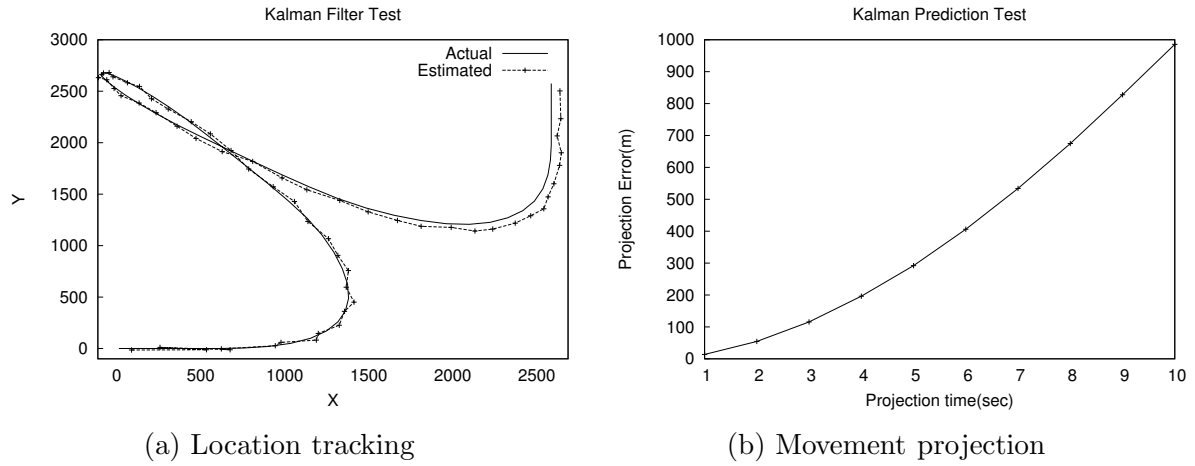


Figure 4.2: Modeled Kalman filter testing

The modeled KF produces accurate estimations of the vehicles location with max divergence of  $8m$  until acceleration of  $\approx 25m/sec^2$  is reached. Since a vehicle would not normally reach this acceleration rate, especially for an extended period of time, the modeled Kalman filter's performance proves to be accurate.

The Kalman filter is additionally tested for its accuracy in projecting future movement to determine how far in advance the algorithm should consider. The projection should look as far in advanced before a large distance error causes the projection to be indeterminate. Figure 4.2b displays the test results of the distance difference between the projection and the actual future location. A spike in the distance error occurs immediately after 3 thus a projection of 3 seconds is chosen as the best balance between time and error.

In addition, Figure 4.2b provides evidence for the statement in Section 3 that movement projection is unreliable for AP prediction. When looking in advance for next node determination the distance error is too large for reliable AP selection, which necessitates using the Kalman filter within the HMM. The projection then does not have to be accurate but instead must only distinguish the movement enough to clarify which AP the vehicle will move to. Prediction results shown in Chapter 7 support this statement.

# Chapter 5

## Prediction Method

In this chapter, we propose and analyze the prediction method using the mathematical tools derived in the previous chapter. First, we explain and discuss using the Kalman filter and hidden Markov model together, with details on their complimentary nature. This is followed by details of the proposed prediction algorithm, which makes the final decision of what AP is predicted for the handover. Finally, we analyze the prediction method in context of the mobile IP environment, and discuss adjustments made to the prediction method accordingly. An overview of the prediction method and its variable flow is illustrated in Figure 5.1. Note that many of the variables mentioned within the figure are detailed in the previous section.

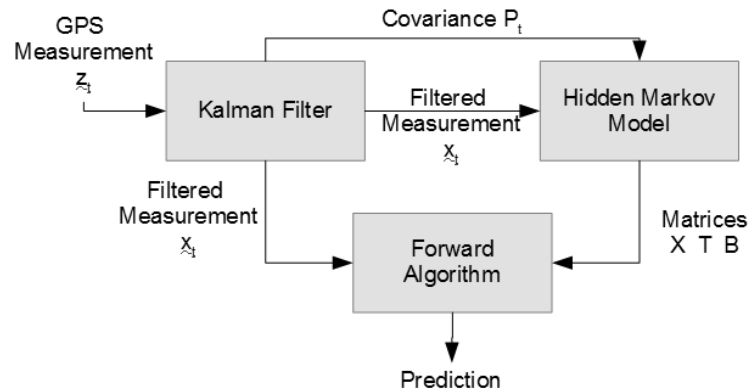


Figure 5.1: Prediction method overview

## 5.1 Combining the Kalman Filter and the HMM

We first detail the communication between the Kalman filter and hidden Markov model before the prediction algorithm is discussed. This includes discussion of the HMM learning algorithms interacting with the Kalman filter, and a proof-of-concept that demonstrates the benefit of using these two methods together.

The Kalman filter is first used to project the vehicle's movement based on incoming GPS signals, which is done according to the matrices and equations detailed in Section 4.3. The output of the Kalman filter's movement projection is then categorized into a discrete set of predefined movement ranges. The resulting movement range value is then sent as an observation to the HMM, where learning equations from Section 4.2.1 are then used to update the HMM matrices. The ranges allow the use of a discrete HMM instead of a continuous, which would have much higher calculation requirements.

The Kalman filter and HMM are chosen for their individually proven effectiveness [67, 77] and for their complementary nature to each other. First, the Kalman filter is able to handle multi-variable observations the HMM is not designed to maintain. For an HMM to handle the multiple observation variables of vehicle movement, there would be a large increase in its calculation and memory complexities due to the large number of additional probability relationships. Second, the Kalman filter solves the issue of a noisy signal causing incorrect observation detection by the HMM. Consider the scenario illustrated in Figure 5.2 with vehicle movement shown in Figure 5.2a and the noisy GPS measurements in Figure 5.2b. If the HMM interprets the GPS measurements without the Kalman filter, Figure 5.2c shows the result are first very different from the actual occurrence. When the observations are first interpreted by the filter, as done in Figure 5.2d, the original state of the vehicle is accurately observed. Third, the Kalman filter covariance estimation for noise reduction benefits the HMM by unreliable data removal. The Kalman filter estimates the covariance of each incoming GPS measurement and the actual location. If the Kalman filter estimates a covariance below a certain threshold, the HMM disregards that measurement in the learning process, removing any potentially misleading data. This is also utilized in the prediction method to ensure predictions based on inaccurate measurements are

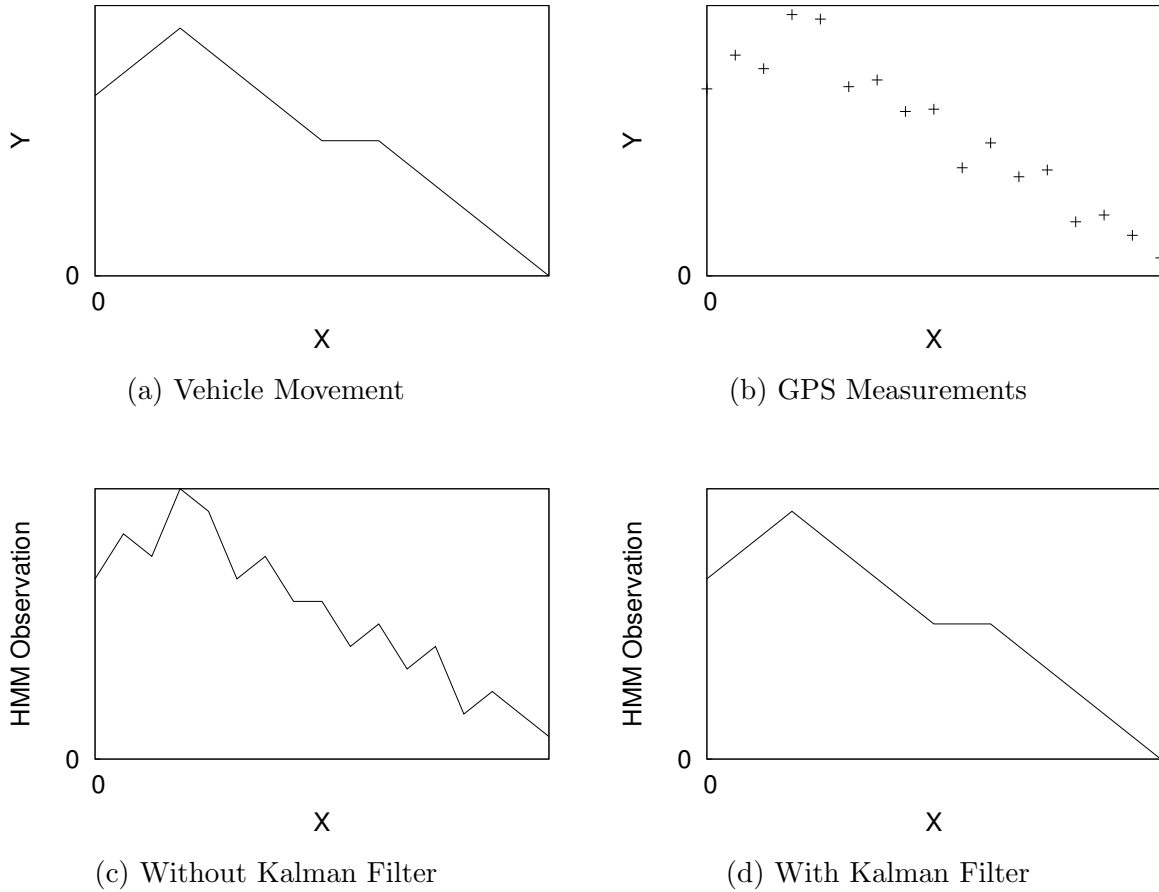


Figure 5.2: HMM observations with(d) and without(c) the Kalman filter

avoided. Utilizing the Kalman filter and HMM with these complimentary additions ensures prediction consistency even in less-than-ideal situations.

## 5.2 Prediction Algorithm

The prediction algorithm is derived using the learned HMM probabilities and the Kalman filter readings. This is achieved by using equations (4.8), (4.14) and (4.15), and the initial HMM values from Section 4.2.2. The most likely subsequent AP can then be determined upon observation of the vehicle's movement. The prediction process, shown in Algorithm 1, first updates  $\phi(t-1)$  using the incoming Kalman filter movement projection at time  $t$  and the online learning equations, denoted here as *LearningAlg()*. The updated  $\gamma(t)$  values

are then compared to the probability threshold,  $p_{th}$ . If a  $\gamma(t)$  meets the threshold, then the AP matching that value is used as the prediction. Otherwise, the algorithm resets a timer,  $t_o$ , to repeat the process at each time step until the prediction is made.

---

**Algorithm 1:** Prediction algorithm

---

```

input : HMM  $\theta$ , Kalman filter measurement ( $k_f(t)$ ), probability threshold ( $p_{th}$ ),
        timer timeout ( $t_o$ ), number of neighbors ( $N$ ), previous state
        probabilities ( $\phi(t-1)$ )
output: Predicted AP,  $x_p$ 

foreach  $t_o$  do
     $\phi(t) = \text{LearningAlg}(k_f(t), \phi(t-1), \theta)$ 
    for  $i = 0$  to  $N$  do
        if  $\gamma_i(t) \geq p_{th}$  then
             $p_{th} \leftarrow \gamma_i(t)$  // set new threshold to return highest value
             $x_p = i$ 
    if  $x_p \neq \text{null}$  then
         $\text{return } x_p$  // output prediction, if it exists
    else
         $\text{reset } t_o$  // no state probability meets threshold, reset timer

```

---

Using a probability threshold ensures enough learning has occurred before a prediction is decided, reducing the chance of a premature prediction.

# Chapter 6

## Predictive Handover Protocol

In this chapter, we adopt the proposed prediction method of Chapter 5 into the mobile IP system and propose a new predictive handover. The method for APs discovering their neighbors and exchanging information for online HMM learning is first discussed. The proposed predictive handover protocol is then discussed, including details on how to handle and detect prediction errors.

In our system, we choose to have APs manage the initial HMM values, and have vehicles use the Kalman filter to conduct the online learning and prediction procedures. We divided the tasks in this way for three central reasons. First, by having each AP maintain the HMM for its local area, system information retrieval is simplified and the required memory is much smaller. If vehicles were to maintain the HMMs for the surrounding network, large amounts of overhead and data storage would be required to transfer and process the information. Second, making prediction from within the vehicle reduces the frequency of packet exchange between the APs and vehicles. If a prediction was conducted by the AP, constant communication of the Kalman filter output would be required from the vehicle to the AP. Third, the vehicle will immediately know the predicted AP, removing the possibility of an AP failing to transmit the predicted AP information to the vehicle. This would otherwise be a costly situation, in which detection of and recovery from the failure would be difficult.

The rest of the chapter details the implementation and utilization of the prediction



method within the mobile IP network, including the retrieval of information for HMM learning, neighbor discovery, and prediction. This is followed by the changes made to the handover protocol to include the prediction and utilize performance benefits of a predicted handover.

## 6.1 Neighbor Discovery and HMM Updating

We first detail the neighbor discovery and HMM updating methods. The prediction process first requires APs to build their HMMs using information on neighboring APs and recorded vehicle movements. This is done by using information packets transmitted through the network between the vehicles and APs so that no manual setup is required. These packets are sent to a vehicle's original AP when the vehicle begins moving away from its new AP. The information packets contain the new AP's IP address, the vehicle's Kalman reading at the time of prediction, its originally predicted AP, its throughput at the new AP, and its dwell time.

For each of these packets received, the AP follows Algorithm 2 for discovering neighbors and updating the HMM. The AP first checks whether the prediction is correct and if there is a previously undetected neighbor. According to the result of these checks, the HMM is then updated according to equations (4.17)-(4.9), which are denoted as *UpdateInitialHMM()* within the algorithm. A neighbor list matrix  $L$  stores neighboring AP IP addresses used to distinguish between APs, where the  $n^{th}$  IP address within matrix  $L$  relates to the  $n^{th}$  state of the HMM. The neighbor list thus functioning as the connection between the network and the HMM.

## 6.2 Predictive Handover Protocol

The predictive protocol is derived through use of the learned HMM, the AP neighbor list, and the prediction algorithm from Section 5.2. The sequential diagram, shown in Figure 6.1, illustrates the overall procedure for the proposed predictive handover. When

---

**Algorithm 2:** Neighbor and HMM updating

---

**input** : predicted AP's IP ( $IP_P$ ), actual AP's IP ( $IP_A$ ), Kalman filter output ( $k_f$ ), number of neighbors ( $N$ ), neighbor list ( $L$ )

**output:** Updated HMM matrices  $\pi$ ,  $A$ ,  $B$ , neighbor list ( $L$ )

**foreach** *Inc. info packet* **do**

- if**  $IP_A == IP_P$  **then** vehicle committed to predicted AP
  - forall** the  $n < N$  **do**
    - if**  $L[n] == IP_A$  **then**
      - $x_a = n$  // find IP in list
      - UpdateInitialHMM**( $x_a, k_f, N$ )
- else** vehicle committed to different AP
  - if**  $L \notin IP_A$  **then** neighbor AP not in  $L$ , add new neighbor
    - $L[N] \leftarrow IP_A$
    - $N = N + 1$
  - forall** the  $n < N$  **do**
    - if**  $L[n] == IP_A$  **then**
      - $x_a = n$
    - if**  $L[n] == IP_P$  **then**
      - $x_p = n$  // also need predicted AP for update
    - UpdateInitialHMM**( $x_a, x_p, k_f, N$ )

---

the vehicle first connects with the current AP, the AP transmits the initial HMM matrices and AP neighbor list  $L$ . The vehicle then replaces its previous HMM and neighbor list, and begins conducting the learning process from Section 4.2.1 with the Kalman filter outputs. While connected with an AP, prediction algorithm 1 from Section 5.2 is conducted after each movement projection until a prediction is returned. Note that the Kalman filter is independent of the network, and therefore continuously tracks the vehicle regardless of the current protocol step.

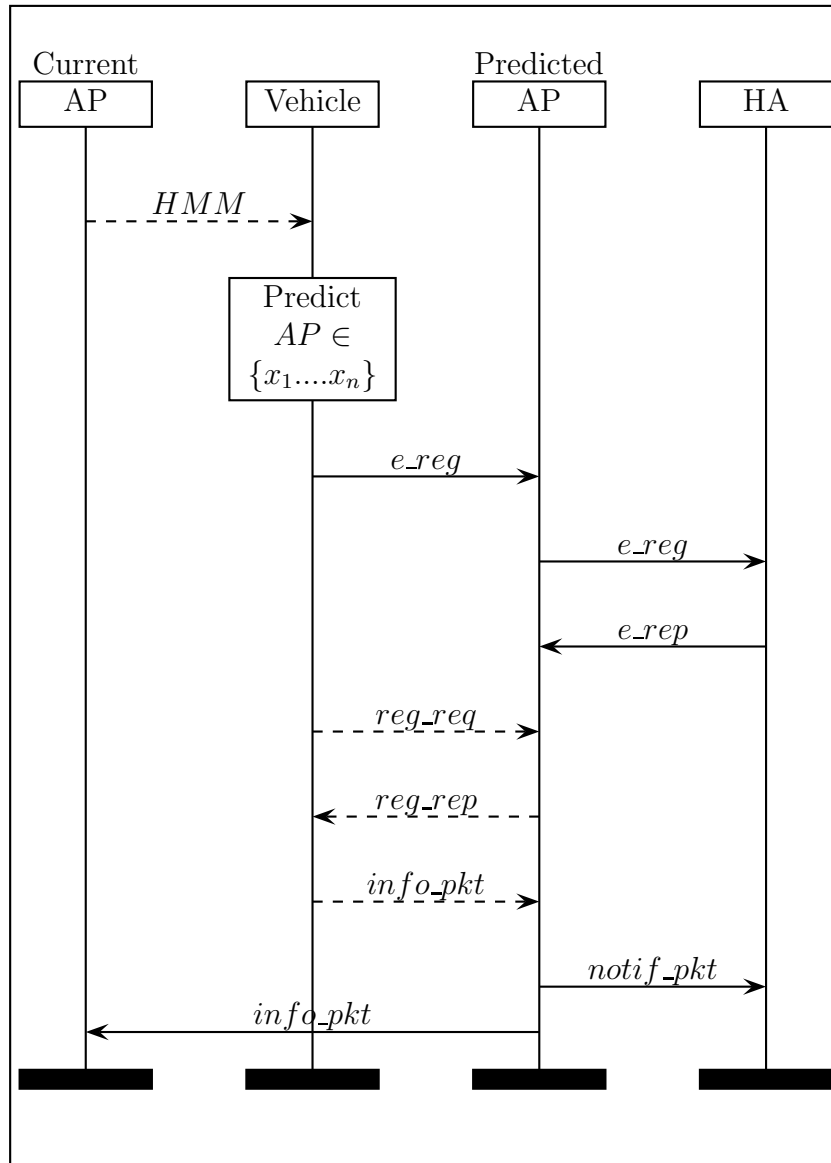


Figure 6.1: Predictive handover protocol

Once an AP's probability reaches the threshold value, the vehicle sends an early regis-

tration request (*e\_req*) packet to the predicted neighbor. This is done through the current AP and over the physical network. Upon receiving the *e\_req* packet, the predicted AP sends an early registration (*e\_reg*) to the HA. The HA then finishes registration with the predicted AP without disconnecting from the original and the predicted AP reserves a care-of address (CoA) for the incoming vehicle after a registration reply message (*reg\_rep*) from the HA is received. This is to prevent packet drops from an incorrect handover timing estimation. A timer is then set by the AP which will cancel the registration upon expiration to prevent endless waiting in the event of an error. Sending the *e\_req* directly to the HA from the vehicle was considered to reduce packet exchanges, but was not done to avoid delays for the predicted AP to be prepared for the vehicle. If the *e\_req* is lagged, the HA may not contact the AP until after the handover has been initiated. This would disrupt the predicted discovery phase. Additional load onto the wired network is worth avoiding this possibility.

When the vehicle begins to lose connection to its current AP, it begins to broadcast a registration request (*reg\_req*) to the predicted AP with an indication that a prediction to this AP was made. When it receives an ad from the predicted AP, the vehicle recognizes the address and sends a registration request (*reg\_req*) with a notification of early registration. The notification triggers the AP to check for a reserved CoA for the vehicle. If it exists, the AP replies to the vehicle with a registration completion message and concludes the handover procedure. After establishing a connection with the AP, the vehicle will then send an information packet (*info\_pkt*) to its previous AP, containing the variables described in Section 4.2. The AP also sends a notification packet (*notif\_pkt*) to the HA to stop sending packets to the previous AP, completing the handover. When successful, the process removes the costly discovery phase and minimizes registration costs. By having both the vehicle and AP contain knowledge of the other, searching for a new connection is not needed. The registration process with the HA is also already complete, so only one packet exchange occurs with immediate response. Therefore the handover overhead and latency are significantly reduced.

There exists the case of a prediction error and the vehicle does not enter the range of the

predicted AP after early registration is conducted. Prediction errors must also be properly handled to minimize their expensive performance costs. This requires early recognition of an error and a backup protocol to be followed upon recognition. Figure 6.2 shows the error protocol followed, where dashed lines represent wireless communication and solid lines represent communication across the network. The error cannot be detected until the vehicle begins searching for the predicted AP. Thus, the process is the same up until the vehicle begins broadcasting the *e\_reg* packets. Due to an error, the vehicle will not enter the predicted AP's range and will continue to wait, during which the vehicle starts building an ad list. A prediction error is assumed once the end-to-end delay of packets received from the vehicle's current AP reaches a set threshold, triggering the vehicle to check the next two highest  $\gamma$  values within its  $\phi$  matrix. If one of these two APs exist within the vehicle's ad list, the vehicle will send a registration request to that AP. If neither of the next two APs are within the ad list, the vehicle then conducts the original handover protocol.

A prediction error is noted within the registration request. When the HA receives the request, it sends a cancellation message, (*cncl\_pkt*), to the predicted AP and stops forwarding packets to it. The predicted AP can then remove the vehicle from its early registration list and free up space for other vehicles. While an error is costly, the reliable performance of the prediction method reduces the error frequency to where overall performance is minimally affected.

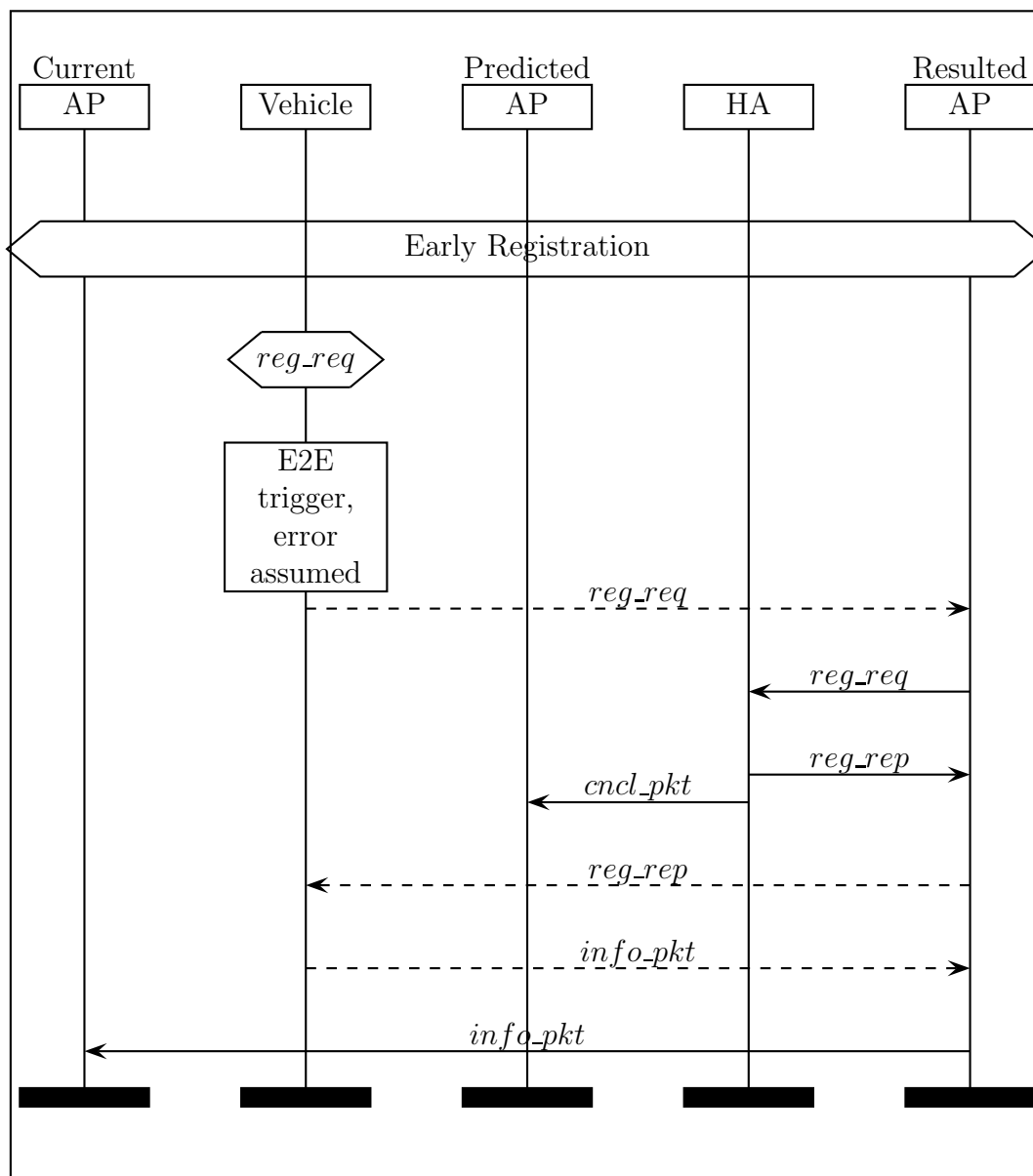


Figure 6.2: Prediction error protocol

# Chapter 7

## Performance Evaluation

In this chapter, we evaluate the proposed HMM-KF predictive handover by analysis of results generated through simulation. The vehicle prediction and mobile IP performance is analyzed and compared with methods of other recent literature. We first detail the network and road environments used for evaluation, then analyze the simulation results. Our analysis begins with parameter optimization and evaluation of online learning; the prediction accuracy is then evaluated, and finally the network performance is analyzed. This order ensures that an improved prediction method is used for analyzing the predictive handover. The common network metrics latency, throughput, and packet-drop rate are chosen as performance measures for analysis. These metrics are considered within the simulation of multiple road and network environments. Simulation of Urban Mobility (SUMO) [71] is used to generate vehicle traffic data within a road network, and Network Simulator (NS-2) [72] is used to simulate the wireless network. Table 7.1 summarizes the environment parameters for the simulation.

### 7.1 Environment Setup

The environment used for simulation is set up to represent a realistic vehicular network. The details for the environment are chosen to provide useful analyses of both prediction network performances. First, the road environment and vehicle movement is discussed,

followed by details of the mobile IP network and routing protocols. The compared methods from other literature are described in Sections 7.4 and 7.5, since different methods are used for prediction analyses and network analyses.

### 7.1.1 Road and Traffic

For the road and traffic environments, we use six separate road environments to observe the robustness of the prediction and handover methods. The first two simulated environments are a 1200m by 1200m urban road network taken from a section of downtown Toronto and Boston, as shown in Figures 7.1a and 7.1d. The urban network tests situations of high turn possibility and large road intersection variation. It also contains a high density of traffic lights which cause less predictable movement due to inconsistent speeds, an important aspect to test for a method that relies on movement. The second environment are sections of Highway 80 and Highway I-95, shown in Figures 7.1c and 7.1f, where vehicles reach much higher top speeds but maintain more consistent movement. This provides isolated observation of the effects of increased speed and more frequent handovers. The last environments are a combination of the previous two environments, where a highway overlaps an urban street network. These are taken from a section of the Bronx, NY that intersects with Highway I-95, as shown in Figure 7.1b, and a section of Highway I-66 going over northern Virginia, shown in Figure 7.1e. The added intricacy of overlapping roads and the dramatic variation in vehicle movement complicates the learning process, testing difficult-to-predict situations.

The road data for the three environments is downloaded from OpenStreetMap [73] and imported into SUMO to generate vehicle routes and traffic flow. The number of vehicles is changed for each simulation in increments of 50, ranging from 100 to 500 vehicles. For our simulations, we choose to categorize our vehicle density by the number of vehicles instead of by vehicles per  $km^2$ . Since our goal is to simulate realistic environments, the vehicles and APs do not maintain an even distribution across the map[74]. As a result, vehicle density varies greatly between APs, which would make the specification of vehicles per  $km^2$  misleading.

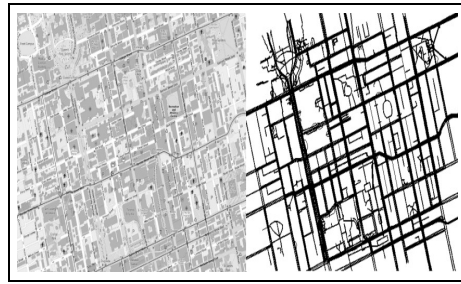


The specific vehicle movement and behavior is determined according to the Krauss car-following model. In this model, vehicles maintain safe following distances and acceleration while following speed limits. Vehicle speeds have a range of  $0 - 20m/s$  in urban environments and  $25 - 30m/s$  in highway environments.

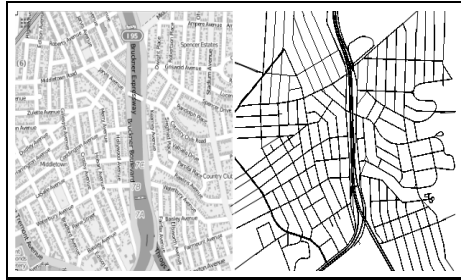
### 7.1.2 Network

To implement the network environment with our road and traffic environments, vehicle movement information is imported into NS-2 for simulation. 802.11p is selected as the protocol to be followed in the MAC layer, since its design improves vehicle connectivity. Vehicular networks utilize both vehicle-to-vehicle (V2V) and infrastructure-to-vehicle (I2V) communication due to common gaps of AP coverage. To evaluate compatibility with this combination, the routing protocols AODV [79], OLSR [80], and GPSR [81] are chosen for both prediction accuracy and network performance analysis. AODV, OLSR, and GPSR are popular routing protocols within mobile ad hoc networks, and represent three different methods to V2V communication. Recent research has been dedicated to testing and adjusting these protocols for vehicular networks [82-85]; therefore, testing these protocols will ensure our predictive handover will be compatible within vehicular networks. AODV, OLSR, and GPSR were modified in this thesis for compatibility with mobile IP, based on the method detailed in [86]. In this method, the V2V routing is used for local packet forwarding for vehicles within close range. If the packet is being sent further than the set range, the packet is sent through the AP using mobile IP.

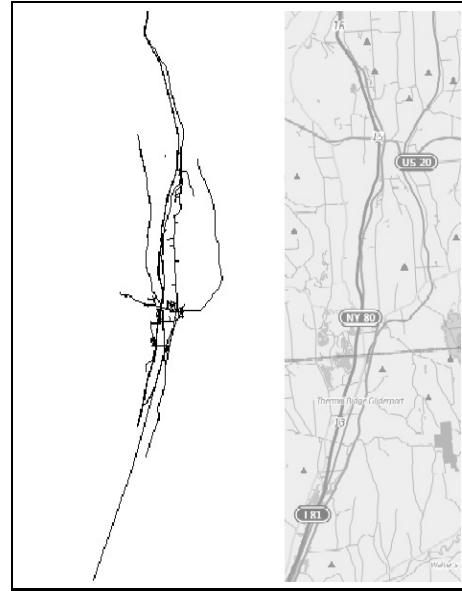
The Kalman filter, online HMM, and prediction algorithm are implemented within NS-2 as an extension to mobile IP. The NS-2 mobile IP implementation was adjusted according to the procedure detailed in Section 6. GPS measurements of the vehicles are provided by NS-2, with noise added artificially. The predefined HMM parameters are defined in Table 7.4.



(a) Toronto, ON



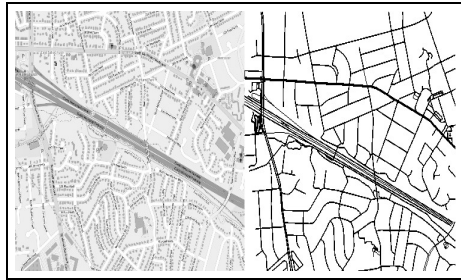
(b) Bronx, NY



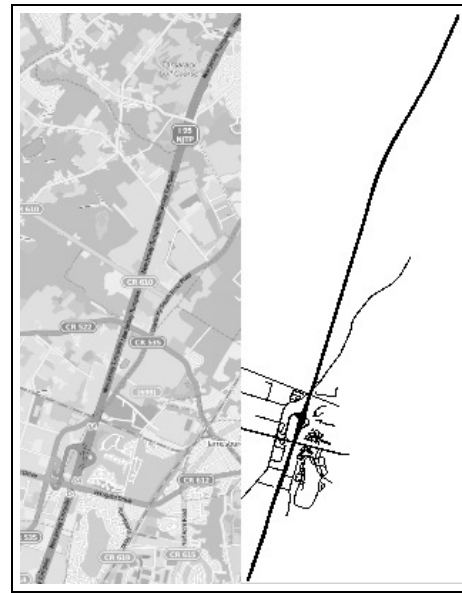
(c) Highway 80, NY



(d) Boston, MA



(e) I-66 going over northern Virginia



(f) Highway I-95, NY

Figure 7.1: Road networks imported into SUMO

Table 7.1: Simulation Parameters

Mobility Simulator	SUMO
Mobility Model	Krauss Car Following Model
Number of Vehicles	100 – 500
Vehicle Speeds	0 – 30 <i>m/s</i>
Road Network	Urban/Highway/Combined
Network Simulator	NS-2.35
PHY / MAC	IEEE 802.11p
Routing Protocol	AODV/OLSR/GPSR
Transport Protocol	UDP
Packet Size	160 <i>bytes/packet</i>
Packet Frequency	250 <i>packets/s</i>
AP Coverage	Spatial Coverage/200 <i>m</i> Range
Simulation Time	600 <i>s</i>
Repetitions	100 (10 × 10 vehicle increments)

## 7.2 Parameter Optimization

Before we evaluate the performance of our proposed method, we first use the simulation environment to optimize variable values that are otherwise considered NP-hard to derive. This is due to the complex nature of a vehicular network and the infinite number of possible values for these variables. The two variables we consider are the number of segments we use to divide the readings, and the threshold value that determines what prediction confidence results in the most accurate performance.

The number of neighbors,  $N$ , is static and decided by the AP distribution, but vehicle movement is continuous, thus allowing it to be divided into a flexible value of  $M$  ranges. The goal for determining  $M$  is to balance the probability dispersion of matrix  $B$ , ensuring that observations are useful and insightful. If the movement is divided into too few ranges, the observation will be unable to properly distinguish between the states; however, if the movement is divided into too many ranges, then  $b_{jk} \rightarrow 0$  will occur for all  $j$  and  $k$ , and observation will again provide little insight. Therefore, setting  $M$  must maximize  $b_{jk}$  values while maintaining variation in observation. A range of  $M$  values are tested against the algorithm's accuracy,  $P$ , using the urban environment. The results, shown in Table 7.2, reveal that the highest prediction accuracy is reached when  $M$  is of value 9.

The threshold also requires a similar balance to produce the most accurate results. The threshold prevents premature predictions from being made, thus improving accuracy. However, if the threshold is set too high, it may delay the prediction too much, preventing early registration from being completed in time. Therefore, the threshold  $p_{th}$  should, at a minimum, balance the probability for a delayed prediction  $p_d$  and a wrong prediction  $p_w$  for best performance. This threshold value is described by equation (7.1). The error probability  $P$  of 7.1 is found using equation (7.2), where  $e_d(t) = 1$  when the prediction is delayed for too long,  $e_w(t) = 1$  when the prediction is wrong, and  $T$  is the size of the sample prediction set.

$$p_{th} \leq \text{Min}(P(p_d|th) \cup P(p_w|th)) \quad (7.1)$$

Table 7.2: Observations  $M$  versus Prediction Accuracy  $P$ 

$M$	$P$	$M$	$P$
1	0.61	12	0.85
2	0.68	13	0.82
3	0.73	14	0.78
4	0.78	15	0.76
5	0.83	16	0.73
6	0.87	17	0.70
7	0.89	18	0.68
8	0.91	19	0.67
9	0.92	20	0.65
10	0.90	21	0.64
11	0.87	22	0.63

Table 7.3: Threshold Values

Threshold	$p_w$	$p_d$	$P(p_d \cup p_w)$
0.2	0.66	0	0.66
0.25	0.48	0	0.48
0.3	0.34	0.01	0.35
0.35	0.2	0.02	0.22
0.4	0.13	0.03	0.16
0.45	0.08	0.03	0.11
0.5	0.06	0.03	0.09
0.55	0.05	0.07	0.12
0.6	0.04	0.28	0.31
0.65	0.04	0.59	0.6
0.7	0.03	0.84	0.85

$$P(p_d|th) \cup P(p_w|th) = \frac{\sum_{t=1}^T e_d(t) + e_w(t)}{T} \quad (7.2)$$

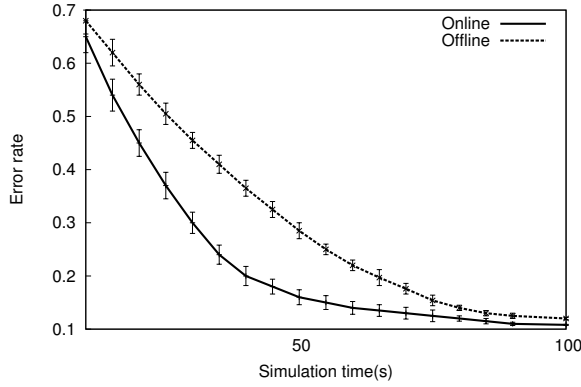
Solving (7.1) is done by a simulation based on the environment defined in Section 7.  $p_{th}$  values between 0.2 and 0.7 were tested, and the results are displayed in Table 7.3. It is found that  $p_{th} \approx 0.5$  results in the lowest prediction error, at a rate of about 0.09.

### 7.3 HMM Learning Analysis

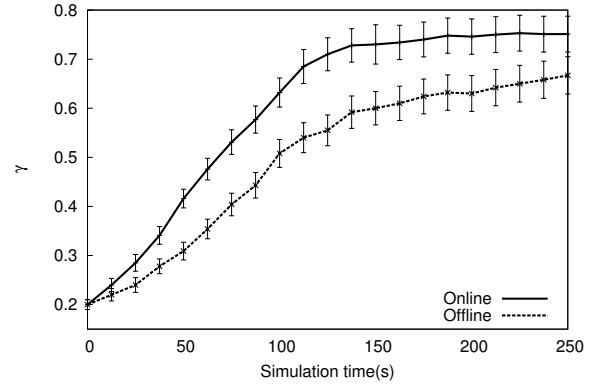
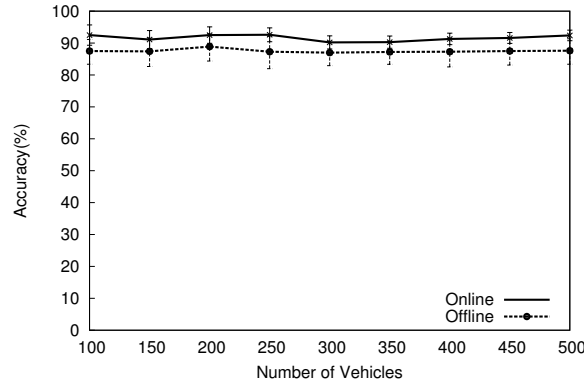
Online learning and initial HMM estimation methods are evaluated to ensure their functionality within the vehicular network. Since neighboring APs are discovered online, the number of states will change throughout the simulation, and early HMM values will not represent the final system. For this reason, final state probabilities and the prediction

Table 7.4: HMM-KF Parameters

Number of states(N)	$\approx 8$
Number of observations(M)	9
KF sampling rate	$\Delta t = 0.2s$
Threshold	$p = 0.5$



(a) Prediction error over time

(b) State convergence of  $\gamma = 0.2$  to  $\gamma = 0.7$ 

(c) Online versus offline prediction results

Figure 7.2: Testing the HMM learning in the mobile IP network

algorithm's accuracy are used as performance measurements instead of probability convergence rates. Initial HMM updating is analyzed through observing the speed and accuracy of converging to a state probability, whereas the overall online learning method is analyzed through observing the prediction accuracy convergence of the prediction algorithm. Both are compared to the equivalent offline learning version of the proposed method.

Figure 7.2 illustrates the average performances of the HMM initial value estimation and the online learning method, taken across the varying vehicle densities. Note that the performances are considered against simulation time, instead of HMM iterations, to consider the performance within the context of the mobile IP network. As shown in Figure 7.2a, the prediction error convergence rate of the online method is higher than the offline method, with the online method reaching below 0.2 error twice as fast as the offline method. The online method adjusts more quickly to the vehicle movement patterns because it handles new data as it arrives. The offline mode does, however, show less variance in its convergence rate, since batch learning is more likely to produce a balanced set of data. The incremental update of the online method depends on the accuracy with which the temporal data represents the system, thus varying more in performance.

Figure 7.2b shows the proposed method's  $\pi$  estimation from Section 4.2.2 compared to the standard offline  $\gamma$  learning described in Section 4.2. The starting value is set to 0.2 within a system where the value is 0.7. The average values of the initial estimation reach within 0.02 of the system probability at about 120s into the simulation, showing much faster convergence than the offline method, which does not reach similar accuracy until 240s, again showing a 50% improvement in convergence. This proves how the proposed method can adjust quickly to any potential changes within the network. One setback with this method is the large variance observed in the initial estimation results, which occurs when randomized vehicle movement fails to reflect the final probability at different moments within the simulation. However, this variance is expected, and is not reflected within the final prediction results. The initial HMM estimation is a precursor to the online learning method, which adjusts these values before a prediction is made.

The short-term prediction benefits of using our online method are illustrated in Figure 7.2c, which displays the overall performance accuracy of the prediction method using the offline and online learning methods. Results show that the online method increases the overall prediction accuracy to over 90%. This improvement over the offline method occurs in the absence of any changes to vehicle traffic patterns or network topology partway through the simulation. This demonstrates that while the online method is designed to



adjust to changes in the road and network, it still increases average accuracy in comparison to the offline method when no changes are made. This is a result of traffic simulations with sample learning data that is not representative of traffic patterns, where the online method can adapt its variables as the simulation continues, but the offline method makes predictions with inaccurate information.

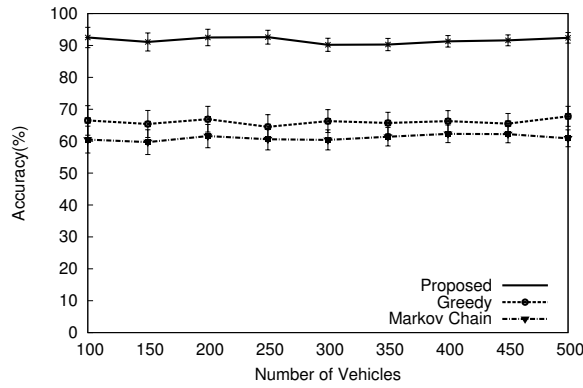
## 7.4 Prediction Analysis

We analyze the performance of the HMM-KF prediction method after initial HMM values have converged. Four recent vehicle prediction methods are implemented for comparison. One method uses multiple order Markov chains [60] to analyze a vehicle's AP history and determine its next most likely AP. The other is a greedy algorithm [61] that uses maximum likelihood to project a vehicle's long-term path with a greedy algorithm to predict the next road segment. These methods are outside mobile IP research, but are chosen so that the proposed method is compared with more recent vehicle prediction methods.

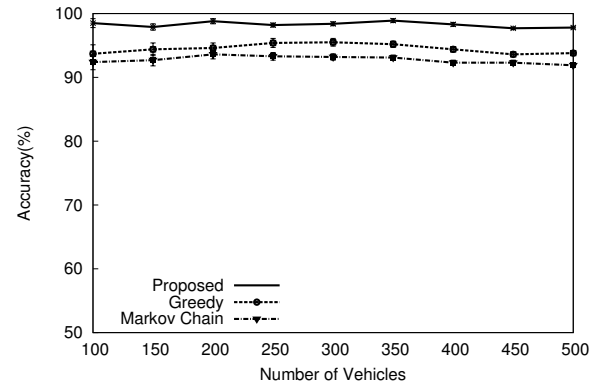
### 7.4.1 Urban with AODV

Figure 7.3a displays the result of the prediction performances using AODV within the urban environment. The accuracy of the proposed method for urban prediction outperforms both the greedy and Markov methods, with an overall success rate of 91%, reducing prediction error compared to the other methods by about two-thirds. The two other methods have similar results in accuracy to each other; the greedy method with an average accuracy of 66% and the Markov method with averages at 61%.

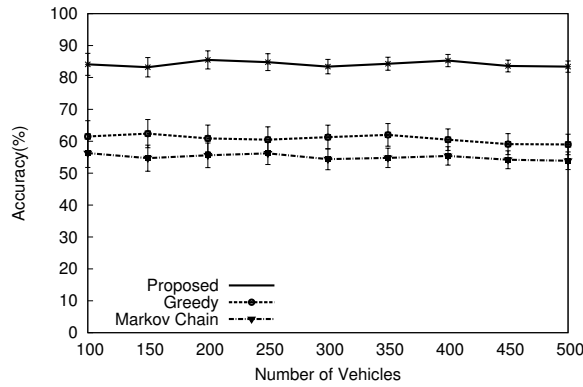
The proposed method's improvement is accredited to the inclusion of temporary movement data and the focus on local variables. The proposed method can ascertain a vehicle's intentions in probabilistically uncertain scenarios while maintaining reliability, since movement information provides more recent insight than history. Otherwise, options with similar probability values and vehicles with misleading history are problematic, issues for both the Markov and greedy methods. The greedy method attempts to resolve the issue



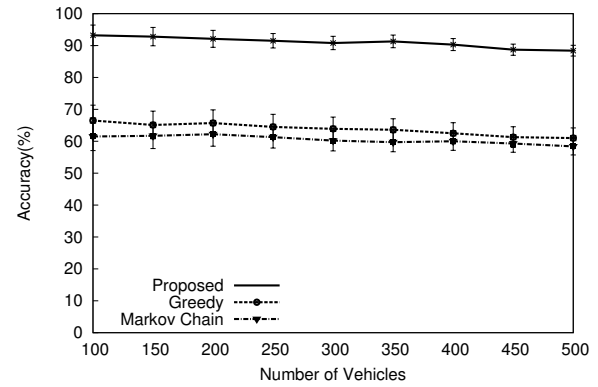
(a) Urban environment with AODV protocol



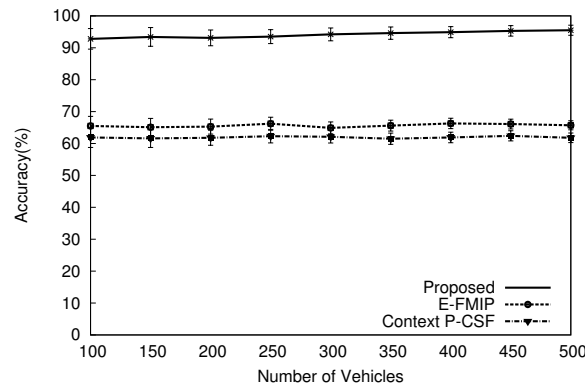
(b) Highway environment with AODV protocol



(c) Urban/Highway environment with AODV protocol



(d) Urban environment with OLSR protocol



(e) Urban environment with GPSR protocol

Figure 7.3: Prediction accuracies versus traffic density

of similar probabilities by using a vehicle's path history to determine its general projected direction, but this method is still affected by misleading vehicle history. One example of

this is when road networks force a vehicle to move in circuitous directions to reach its final destination, which is often the case with one-way streets. Another issue that arises from failing to consider temporal movement is the inability to attain more information when there is otherwise insufficient data to make a prediction. The greedy and Markov methods resort to an uninformed guess, while the proposed method can delay for more movement readings until enough data is collected for a well-informed decision. A problem delaying can cause is when a short AP dwell time, combined with too much prediction hesitation, results in a late prediction. However, the frequency of this occurrence is minimized by the tested threshold selection, as discussed in Section 4.

The effects of traffic density on vehicle movement minimally affect the accuracy of all three methods. This is expected for the greedy and Markov methods, since they do not depend on any variables changed by traffic; however, the proposed method does depend on such variables. The accuracy consistency of the proposed method is due to a balance maintained across different vehicle densities. In higher traffic density, the fluctuation of vehicle movement increases because of vehicles avoiding collision with each other, and at a similar rate the vehicle average speed decreases. This reduces the urgency to make a prediction when movement predictability is also reduced; therefore, more data can be collected to maintain consistent accuracy.

### 7.4.2 Highway and Urban/highway

Figure 7.3b displays the accuracy results from the highway scenario, where all three methods demonstrate much higher performances. In this scenario, the proposed method shows an accuracy of about 98% while the greedy and Markov methods sit at approximately 94% and 93%, respectively. Improved accuracy compared to the urban scenario is due to the easy predictability of a highway, where vehicles have very limited path options. The proposed method still reduces prediction error by about 65%. The slightly higher performance of the proposed method is a result of detecting when a vehicle exits the highway. While rare, the extreme change in speed gives a clear indication of a vehicle exiting, a change that the greedy and Markov methods can not detect.

Despite the performance increase in the isolated highway scenario, a large decrease in performance occurs when the urban roads and highway overlap. Figure 7.3c shows all three methods performing more poorly than in other environments. The proposed method has about an 85% accuracy, the greedy with about 60% accuracy, and the Markov chain showing a 55% accuracy.

The added complexity of having a highway pass over urban streets makes AP prediction much more difficult. The prediction algorithms only consider two dimensions, and, therefore, cannot directly observe on which street the vehicle is located since they are blind to the third dimension. Here, the proposed method demonstrates a 60% reduction to the prediction error compared to the greedy method, 10% less than the reduction in the urban environment. The proposed method suffers more degradation in accuracy due to the increased variety of movement the HMM must account for, since one AP may observe two different vehicular movement patterns of different speeds. This could potentially be of no concern when implemented in the real world, because the elevation may cause signal differences that cause vehicles to only connect with APs on their elevation.

### 7.4.3 OLSR and GPSR

Figure 7.3d shows the results in accuracy when using the OLSR routing protocol within the urban scenario. Accuracies within OLSR are similar to AODV with few vehicles, but performance reduces as traffic density increases. From 100 to 500 vehicles, this results in about a 15% increase in error for the proposed and greedy methods, and a 5% increase for the Markov method. This reduction is due to the increased occurrence of multi-hop routing, where the hierarchical packet forwarding may cause vehicles to move farther from the current AP without needing to conduct a handover. The vehicle may then commit to a different AP that would otherwise not be predicted. However, this event occurs infrequently, and the disruption to prediction accuracy is relatively small. The GPSR results shown in Figure 7.3e display overall higher accuracy compared to OLSR. GPSR's location-based packet forwarding is performed based on the closest node, causing vehicles to consistently commit to the nearest AP and increasing the handover predictability. The

greedy and Markov methods have similar accuracies between AODV and GPSR, but the proposed method shows a small improvement in accuracy. The GPSR location-based forwarding compliments the proposed method, and incorporates GPS information resulting in improved predictability of network behavior.

#### 7.4.4 Vehicle Location Predictions

The prediction method was also tested outside of the network for analysis of its general vehicle prediction accuracy. This includes predicting a vehicle's regional area and its exact location. Two other recent methods, deemed more appropriate for this analysis, are also implemented and tested for comparison and analysis to the proposed method. One is a cell-frequency method which utilizes the cell movement frequency of past vehicles to probabilistically determine the next cell [21], the other is a path comparison method that matches the vehicle's current path with other vehicle paths to predict the future movement [15].

The results from the described simulation are presented here for analysis. Figure 7.4a illustrates the accuracy results for each of the ten sets of traffic simulations as well as their respective standard deviations.

The results show the proposed method consistently performing better than the cell frequency and path comparison methods with about 20% increase in accuracy compared to the closest competitor. The KF-HMM's consideration of temporal movement data and probability relationships to increase the performance reliability largely contributes to this improvement. By taking the highest probability of what the vehicle is currently doing, instead of what it has previously done, the method produces better accuracy in situations where the vehicle is taking a less predictable path. Similarly, the temporal data resolves the issue of misleading historical information, of which other methods do not address. When using path comparison, the event of a vehicle starting a similar path but then veering away results in the method assuming an incorrect path and predicting incorrectly. When instead considering cell frequency, only vehicles taking the most common path will be predicted correctly.

The path comparison outperforms the cell frequency method due to the inclusion of

path information within its probability evaluation. This information is historically based on previous vehicles, but it still provides additional insight into the individual vehicle's future movement. The cell frequency method only observes neighboring region probability and so it has no method of distinguishing between individual vehicles.

Figure 7.4b shows the accuracy and standard deviation of the methods across different vehicle densities. The average prediction performance for all three methods remains consistent across the different traffic densities. However, there is a large difference in the standard deviation due to the path comparison and cell history methods relying on traffic consistency for performance. In lower traffic densities, their overall accuracy is much more affected by the chance of individual vehicles choosing less likely paths. Since vehicle consistency is random, the resulting accuracy varies widely. The proposed method does not depend on the regularity of a vehicle's route, and shows much smaller performance variance.

In addition to regional accuracy, the accuracy of predicting the vehicle's location is also considered. Figure 7.4c illustrates the distance error between the predicted location and the actual location versus the time in advance in which the method is predicting. The methods all perform similarly when predicting within the first two seconds because of the vehicle's physical restrictions. However, as predictions are made farther into the future, performance begins to differ in accordance with how accurately they predict the correct region. The proposed method maintains the lowest error since the consideration of movement improves the ability to estimate the vehicle's future location in addition to its high regional accuracy. The path comparison method performs better than the cell history for its ability to compare to previous vehicles; however, the error spikes if it picks the wrong path. This spike is why the differences in distance-error between cell history and path comparison are smaller than the differences in regional accuracy.

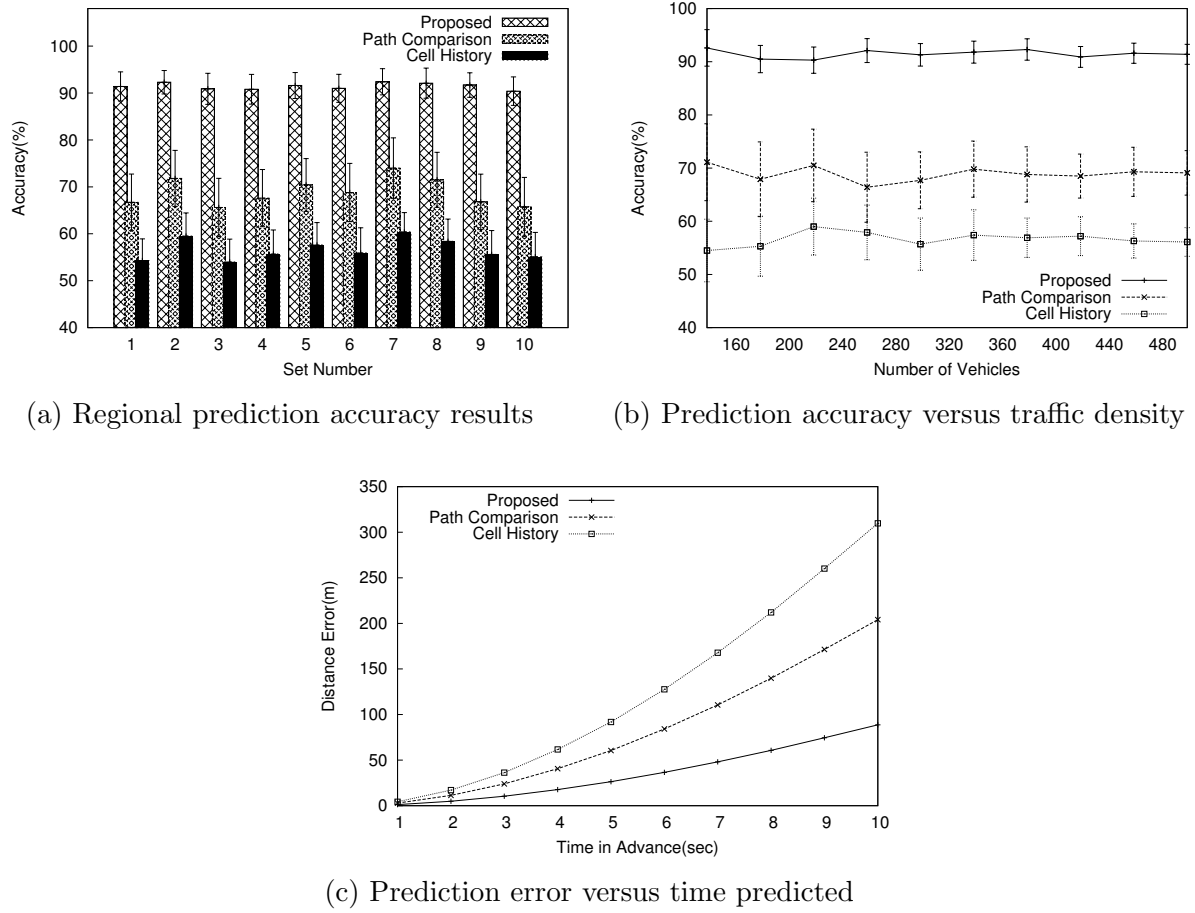


Figure 7.4: Prediction accuracies versus traffic density

## 7.5 Network Performance

In this section, we consider the predictive handover using the improved vehicle prediction method in terms of network performance. The recent handover methods E-FMIP [10], context P-CSF [8], FPMIP-PT [44], and C-HMIP [1] are implemented and tested for comparison. The E-FMIP handover extends the fast handover by adding additional FBACk retransmissions through a media independent handover, ensuring FBACk delivery to the mobile node and improving its consistency. The context P-CSF, comparable to the seamless handover, adds context messaging between IP multimedia subsystems (IMS), which act as equivalents to MAPs. The context messages sent between the IMS improve transitions to new IMS by the direct transfer of registration information to reduce HA communication. FPMIP-PT enhances the fast-proxy mobile IP handover by pre-establishing tunnels

between neighboring mobility access gates (MAG) before the handover begins. MAGs are similar to MAPs of the HMIP, but further track the vehicle to reduce the vehicle's involvement in the handover. By minimizing the tunneling time, the FPMIP-PT reduces registration latency when the vehicle switches between MAGs. The last method is the C-HMIP, which improves upon the hierarchical handover by adding context messages between the vehicles and APs to improve vehicle and AP network awareness. The context messages provide vehicles with knowledge of surrounding APs before the handover, allowing them to skip the discovery phase of the handover. FPMIP-PT and C-HMIP results are shown separately from P-CSF and E-FMIP results for clearer analysis.

### 7.5.1 Urban

First we consider the latency, throughput, and packet drop rate within the urban environment. Latency represents the amount of downtime a vehicle may experience before receiving IP packets at its new IP address. If it is too large, it will disrupt IP services. Figure 7.5 displays average latency values tested across different routing protocols, where the proposed method shows a 40% improvement compared to the closest other method. E-FMIP has the largest latency of the three, with an average of  $\approx 80ms$  across the different scenarios, as it provides no improvement to the costly tunneling process and it relies on a MAC layer trigger. The MAC trigger occurs when the vehicle has already begun transitioning to a new node without enough time in advance to avoid mobile IP latency, particularly when considering vehicle speeds. The context P-CSF performs better than the E-FMIP with an average latency of  $60ms$ , due to its improvement to both the discovery and registration processes. The reduction of registration latency by using IMS also provides the MAC trigger of the fast handover with a larger time frame for success, adding further improvement and consistency. However, unavoidable mobile IP handover processes, such as creating the tunneling route between the IMS and the current AP, still cause the results to be less than optimal. This latency is avoidable with a predicted handover, allowing the proposed method to reduce average latency to  $35ms$ , which is 40% less latency than P-CSF. An issue imposed by the predictive method is the inconsistency compared to the fast



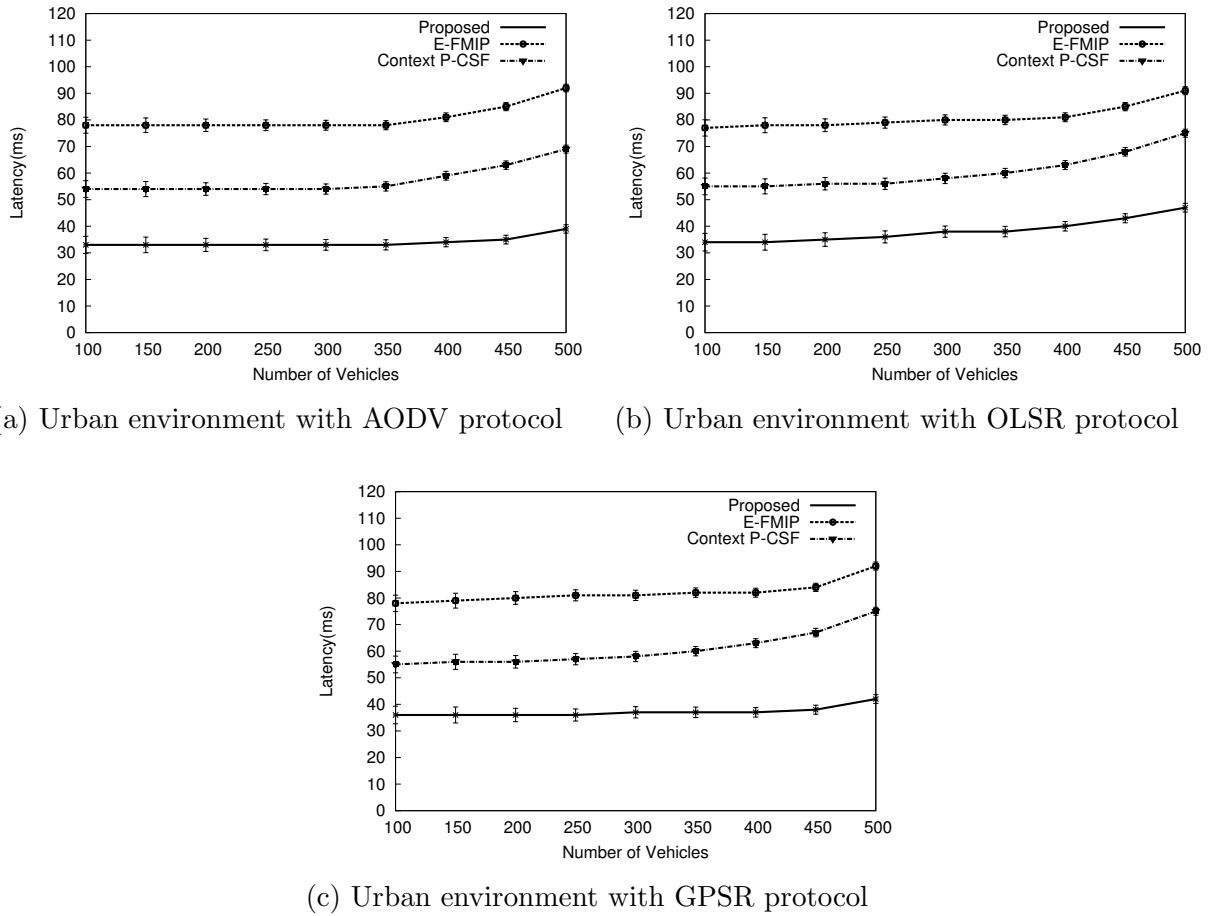


Figure 7.5: Latency results versus vehicle density

or seamless handover, since an incorrect prediction causes a latency as high as the original handover. However, this is minimized by the method's improved prediction accuracy.

Figure 7.5a shows the impact of traffic density on latency, using AODV in the urban environment. The latency values are consistent until the APs begin reaching full saturation at about 350 vehicles, when high collision rates and longer AP access times occur. Between 350 and 500 vehicles, we observe a  $5ms$  latency increase for the proposed method, a  $15ms$  increase for P-CSF, and a  $15ms$  increase for E-FMIP. Traffic density affects E-FMIP and P-CSF latency the most due to their added overhead to the handover, the E-FMIP adding packets for MAC layer exchanges and P-CSF adding packets for context messages to HA's. The predictive handover maintains the most consistency because it removes the ping-pong packet exchange that otherwise occurs during the handover, reducing the costs of

collision rates and AP access times. Figure 7.5b illustrates the results when OLSR is used instead of AODV. Here, the E-FMIP experiences the least latency increase at about  $15ms$ , compared to the proposed and P-CSF methods that experience a  $15ms$  and  $20ms$  increase, respectively. Despite maintaining the highest latency, E-FMIP is least affected by OLSR, since it is independent of network topology and is thus unaffected by the additional topological complexity and packet forwarding in OLSR. In contrast, Context P-CSF's latency performance is greatly affected because its adjusted IP topology in conjunction with OLSR requires higher overhead and information exchange for IMS transitions. The IMS must often track the vehicle through multiple nodes. The proposed method does not suffer from IP and topological complexity, but still experiences performance reduction due to the added prediction error from OLSR. This is opposite to the GPSR results in Figure 7.5c, where the results are closer to AODV in performance consistency with latencies lower than OLSR. Both AODV and GPSR are reactionary protocols, but GPSR's use of GPS measurements instead of sequence numbers functions results in slightly reduced performance. For the predictive method, a slower increase in latency is observed between 300 – 400 vehicles, where the benefits of the improved prediction accuracy within GPSR slows the effects of AP saturation.

Due to increased overhead required for proactive hierarchical packet forwarding, traffic performance costs are amplified when switching to OLSR, as shown in Figure 7.5b. Despite maintaining the highest latency, E-FMIP is least affected by OLSR, since it is independent of network topology and is thus unaffected by the additional topological complexity and packet forwarding in OLSR. In contrast, Context P-CSF's latency performance is greatly affected because its adjusted IP topology in conjunction with OLSR requires higher overhead and information exchange for IMS transitions. The IMS must often track the vehicle through multiple nodes. The proposed method does not suffer from IP and topological complexity, but still experiences performance reduction due to the added prediction error from OLSR. When then using GPSR, the results are closer to AODV in performance consistency with latencies lower than OLSR, as shown in Figure 7.5c. Both AODV and GPSR are reactionary protocols, but GPSR's use of GPS measurements instead of sequence numbers functions results in slightly reduced performance. For the predictive method, a

slower increase in latency is observed between 300-400 vehicles, where the benefits of the improved prediction accuracy within GPSR slows the effects of AP saturation.

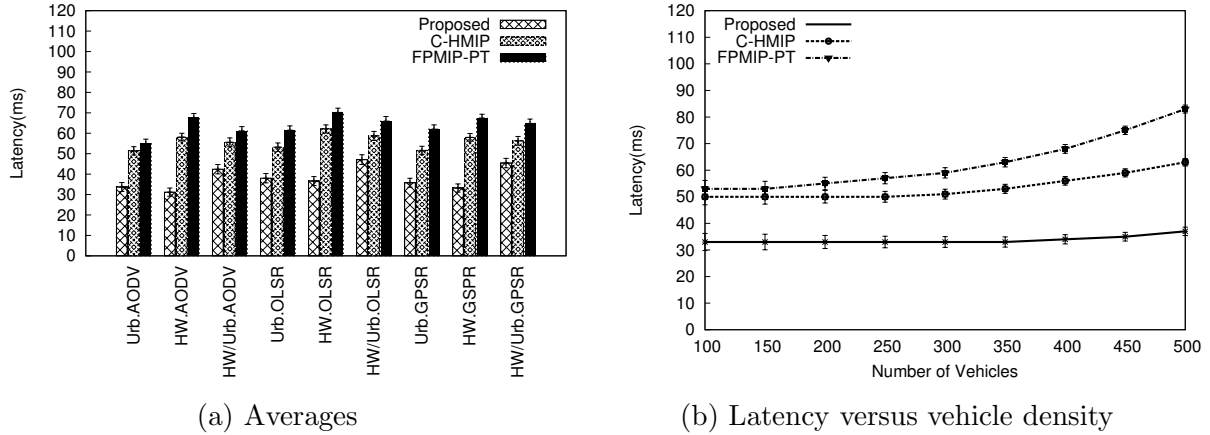


Figure 7.6: FPMIP-PT and C-HMIP compared latency results

Figure 7.6 presents the latency performances for FPMIP-PT and C-HMIP in comparison with the proposed method. The averages from the different routing protocols and road networks are illustrated in Figure 7.6a. Similar to the other comparison, the proposed method consistently has the lowest latency average in all scenarios. With the higher accuracy, the occurrence of a predicted handover is frequent enough to lower the average latency below the FPMIP-PT and C-HMIP results. This performance difference is amplified when on the highway, where latency is lowered further for the proposed method and increased for the other two methods. This is due to the predictability of the highway increasing the accuracy of handover prediction, improving the early registration consistency. On the other hand, the proposed method suffers the worst latency degradation in the highway/urban road network and the OLSR routing protocol where prediction becomes more difficult. The overlapping roads of the road network and the hierarchical routing of OLSR make it more difficult to predict the next handover.

The second set of average latencies versus traffic density are averaged and observed in Figure 7.6b. The proposed and C-HMIP methods are similarly affected by the density, unlike FPMIP-PT, which begins to suffer much larger latencies as density increases. The predictive handover and C-HMIP require minimal packet exchange during the actual han-

do over since the context messages of the C-HMIP and the early registration of the predictive handover allow removal of the discovery phase. Without discovery overhead, network loads caused by increased traffic minimally affect the average handover latency. The FPMIP-PT struggles due to the higher overhead it requires for vehicle tracking, making it difficult to handle multiple vehicles conducting handovers at similar times.

Figures 7.7a, 7.7b, and 7.7c show the effects of traffic density on throughput using AODV, OLSR, and GPSR, respectively. As expected, the throughput performance closely mirrors the latency results but with smaller differences between methods. The proposed method shows about a 10% gain in throughput over Context P-CSF. This smaller gain is because handovers only make up a small portion of the total time a vehicle is receiving packets. It is also observed that throughput performance begins deteriorating at a lower vehicle density than for latency. This results from the occurrence of packet collision before saturation is reached, which only has a noticeable effect on throughput. Further deterioration is observed in OLSR, as the additional overhead causes an increase in the collision rate. The proposed method in GPSR again shows a similar performance difference between 300 – 400 vehicles, as is explained in the latency discussion.

The other set of throughput results is displayed in Figure 7.8. Figure 7.8a shows the throughput versus traffic density, where similar performance patterns to the latency and packet drop results are observed. A larger performance gap between the proposed method and C-HMIP is observed because of the higher overhead required by C-HMIP between handovers. In C-HMIP, context messages are exchanged at a regular rate, compared to the proposed method, which only requires the exchange of the HMM values and the early registration packet. Figure 7.8b shows the average throughput performances across routing protocols and road scenarios. The one scenario where the proposed method has similar performance to the C-HMIP method is when the OLSR protocol is being used on the highway. The performance costs of the lower prediction accuracy caused by OLSR routing is amplified by the higher vehicle speeds on the highway because of the latency caused by a wrong prediction having a larger impact with the lower AP dwell times. C-HMIP's network awareness improves its handling of both speed and topological changes, allowing

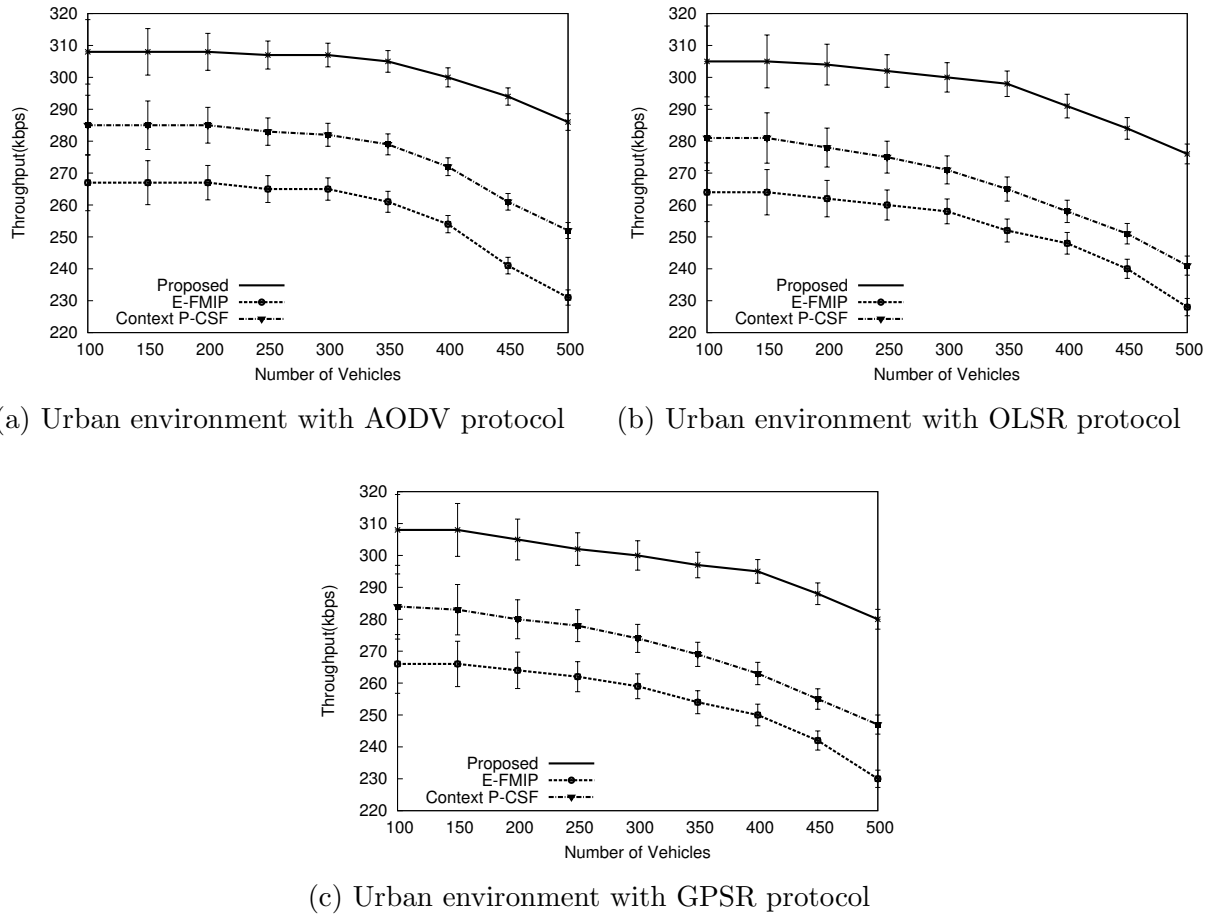


Figure 7.7: Throughput results versus vehicle density

it to maintain a closer performance.

The packet drop rates in Figure 7.9 show slightly greater variation in performance than the throughput. Overall, the results show our proposed method reduces the average packet drop rate by about 10% and reduces the packet drop increase caused by saturation by about 55%. Before AP saturation is reached, the proposed method performs more poorly than the E-FMIP and P-CSF, due to incorrect predictions producing greater chances of packet loss regardless of vehicle density. The other methods benefit from the E-FMIP establishing an early connection with the next AP to provide a smoother handover; however, packet loss begins to increase for the E-FMIP and P-CSF at a much quicker pace than the proposed method as saturation is reached. Once again, this is due to the higher handover overhead of the E-FMIP and P-CSF methods.

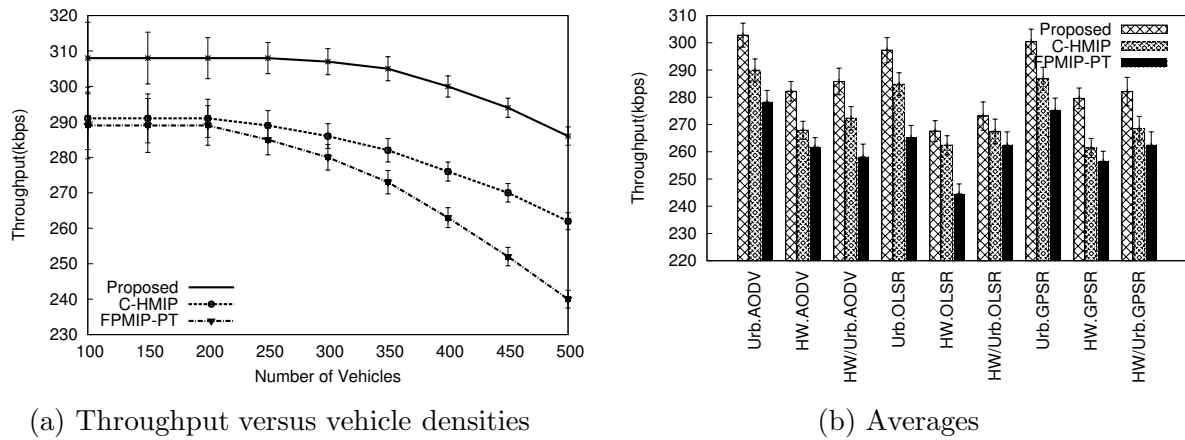


Figure 7.8: FPMIP-PT and C-HMIP compared throughput results

Figure 7.9 illustrates the results of the average packet drop rate for individual vehicles. The effect of traffic density on the average packet drop rate is displayed in Figure 7.9. At lower vehicle densities, the proposed method has a higher packet drop rate than the other two methods. This is due to the packets dropped from a wrong prediction, which occurs independently of the vehicle density. As no current solution exists for preventing packet drop due to a prediction error, packet drop rates will always be above zero. Both C-HMIP and FPMIP-PT have added security to ensure fewer packets are dropped, which brings packet drop rate to nearly zero at low densities. However, as traffic density increases, the C-HMIP and FPMIP-PT packet drop rates rise at a faster pace due to packet overhead. FPMIP-PT is the most affected due to its high overhead and a MAG's inability to handle a high rate of handovers. The C-HMIP conducting information exchanges before the handover helps to lower its drop rate, but it still has a larger overhead than the predictive method, causing it to have higher packet drop rates. Figure 7.11c compares the packet drop averages between the three methods, where less variation is seen compared to the latency results. The largest contributing factor to packet drop rates is the traffic density. While the road network and routing protocol has some varying effect on the drop rate, the performance pattern of Figure 7.9 is similar in all scenarios. Otherwise, the average packet drop rates closely follow the latency results.

Figure 7.10 illustrates the results of the average packet drop rate for the FPMIP-PT

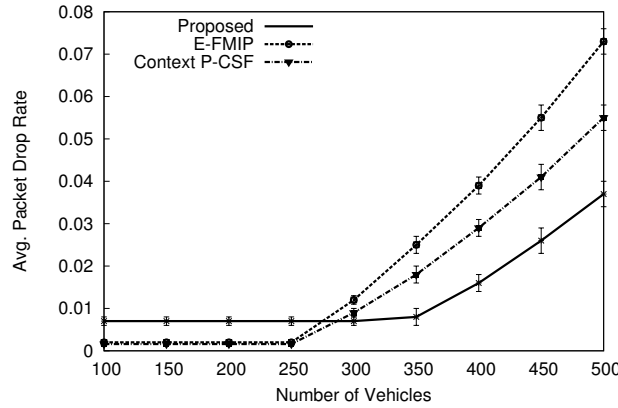
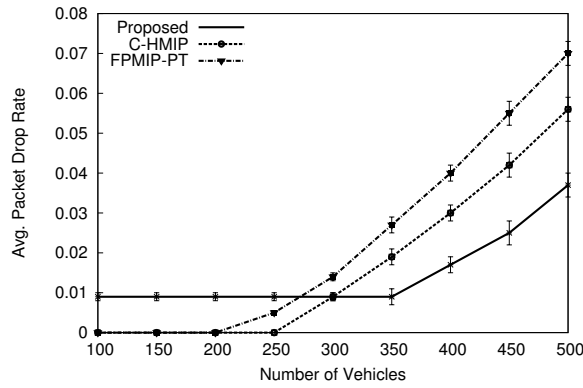


Figure 7.9: Packet drop rate versus vehicle density

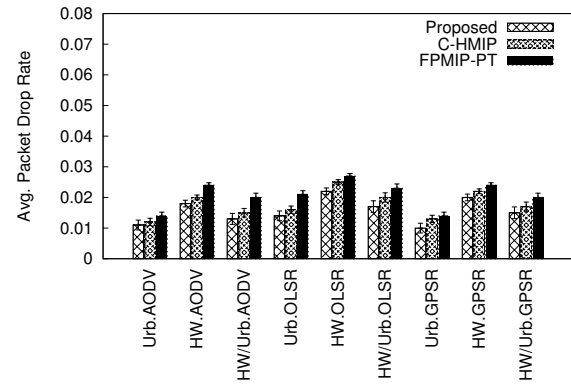
and C-HMIP methods. The effect of traffic density on the average packet drop rate is displayed in Figure 7.10a. Similar to E-FMIP and Context P-CSF, both C-HMIP and FPMIP-PT have added security to ensure fewer packets are dropped, which brings packet drop rate to nearly zero at low densities. As traffic density increases, the C-HMIP and FPMIP-PT packet drop rates rise at a faster pace due to packet overhead. FPMIP-PT is the most affected due to its high overhead and a MAG's inability to handle a high rate of handovers. The C-HMIP conducting information exchanges before the handover helps to lower its drop rate, but it still has a larger overhead than the predictive method, causing it to have higher packet drop rates. Figure 7.10b compares the packet drop averages between the three methods, where less variation is seen compared to the latency results. The largest contributing factor to packet drop rates is the traffic density. While the road network and routing protocol has some varying effect on the drop rate, the performance pattern of Figure 7.10a is similar in all scenarios. Otherwise, the average packet drop rates closely follow the latency results.

### 7.5.2 Highway and Urban/highway

The effects of traffic density is minimally altered between road environments, therefore we compare only averages between the urban and highway scenarios to avoid redundancy. Figures 7.11a, 7.11b, and 7.11c display average performances between the scenarios using

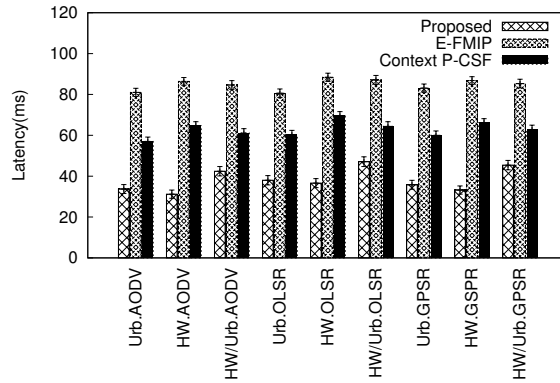


(a) Packet drop rate versus vehicle density

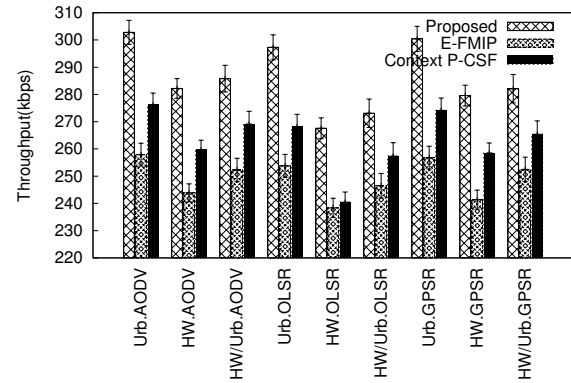


(b) Averages

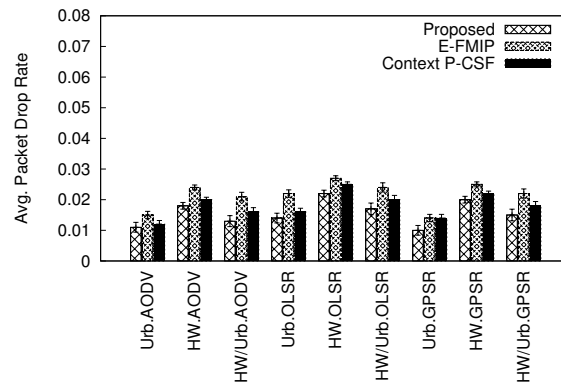
Figure 7.10: FPMIP-PT and C-HMIP compared packet drop results



(a) Latency averages



(b) Throughput averages



(c) Packet drop averages

Figure 7.11: Performance averages across all environments



latency, throughput, and packet drop rate, respectively. Within the highway environment, the proposed method shows an average latency reduction of 5% compared to its urban performance. This is due to the improved prediction accuracy reducing the handover latency. This is unlike the other two methods, which both suffer about a 10% increase in latency due to the increased speeds making it difficult to complete their processes in time. The higher frequency of handovers increases the additional costs of the added P-CSF and E-FMIP packets, thus amplifying the performance degradation within the highway scenario. This frequency cost is nullified for the proposed method due to the improved prediction accuracy. For throughput and packet drop rate, all three methods suffer similar performance differences from urban to highway, which is expected, as performance costs from an increased frequency of handovers is unavoidable. Bigger losses are seen for the proposed method and P-CSF than for E-FMIP when using OLSR in the highway environment, as the problems created by the hierarchical proactive routing are also amplified by the increased handover frequency. Performance is still reduced when using GPSR, but less so. Although OLSR has higher overhead issues, the reactionary routing of AODV and GPSR has a slower response time, which has a large performance cost on the highway. In addition, GPSR's movement calculations are affected by higher rates of movement, thus resulting in worse performance when using GPSR compared to AODV. The proposed method is affected slightly less with GPSR than with other methods, due to the small improvement in prediction accuracy with GPSR, further avoiding prediction error costs.

In consideration of the highway/urban scenario, the additional complexity has little effect on E-FMIP and P-CSF because of their independence from the road network. Both methods do not depend on knowing which direction the vehicle will move to next. As expected, their performance falls in between the results from the highway and urban scenarios due to the variation of vehicle speeds. The proposed method's performance reduction is larger due to reduced prediction accuracy, as discussed in Section 7.4. This reduced accuracy has a greater performance effect on the highway because wrong predictions have higher costs when the dwell time is lower. Overall, the change in road scenarios have larger performance effects on the predictive method. However, it still outperforms the other methods in all scenarios, demonstrating its improved compatibility.

# Chapter 8

## Conclusion and Future Work

### 8.1 Conclusion

The potential for vehicular networks calls for IP compatibility to provide additional services. However, the mobile IP handover is an expensive process, with performance costs too great for vehicular network implementation. Recent methods aimed to resolve the handover issues suffer from the minimum requirements required, which is recognized to be resolvable with a predictive handover. In this thesis, we proposed an HMM-KF predictive handover which utilizes both movement projection and probability to attain higher handover prediction accuracy and resolve the mobile IP issues. A Kalman filter and an online HMM were modeled according to the variables available within a vehicular network. A predictive handover protocol that uses the derived models was then introduced for conducting the advanced handover and reducing network costs of a prediction error. The proposed method was tested and analyzed in comparison with other recent methods using simulated routing protocols and road networks taken from map data. Results gathered from the simulations illustrate that our proposed KF-HMM method reduced prediction error by about 70%, and reduced handover latency by about 40%.

## 8.2 Future Work

This thesis has proposed a predictive mobile IP handover that shows significant improvement to predicting handovers and network performance. For our future work, we plan to extend upon and improve our predictive handover by exploring the following possible research directions.

**Prediction Timing** An area to explore would be to also predict the timing of the handover. This would prevent early registration from occurring too far in advance before the handover occurs. In addition, timing prediction could be used to improve prediction error detection by recognizing if the vehicle does not detect the predicted AP within the given time frame.

**Dynamic Threshold Calculation** Our current threshold calculation includes only considering the overall prediction accuracy of the entire simulation. However, individual AP environments vary greatly, and an encompassing threshold does not produce the best accuracy for each AP. Therefore, having each AP dynamically calculate its own threshold would improve overall accuracy.

**Caching at Next AP** Caching future IP packets at the vehicle's predicted AP could further improve performance. If the upcoming IP packets can be predicted, caching them at the next AP would produce near-zero delay in the IP handover. For example, if a vehicle is downloading a video, an algorithm could determine to have a section of it cached at the predicted AP.

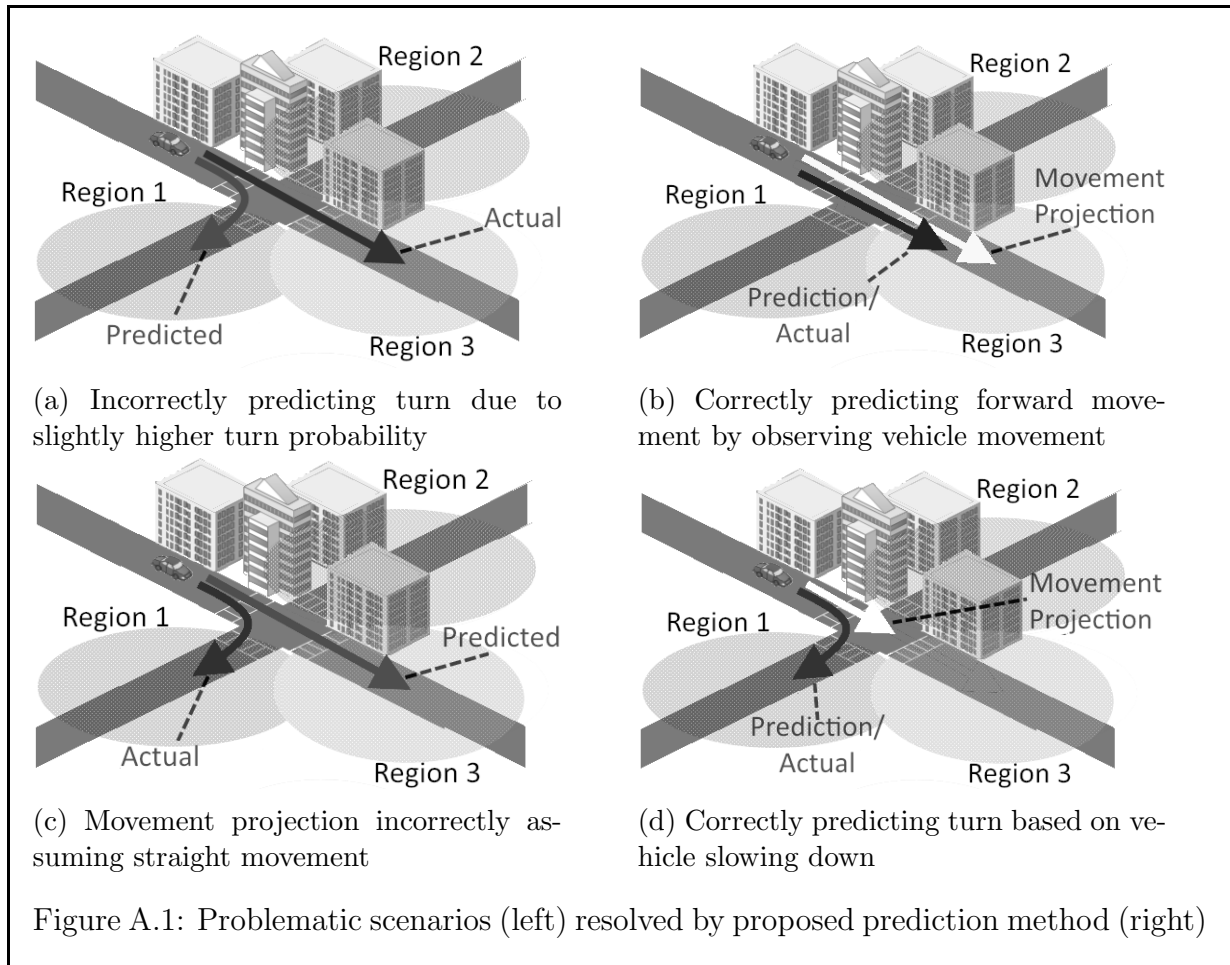
**Addressing Inconsistencies** Our method suffers from inconsistencies that reduce the performance reliability. Especially in comparison to other methods, the predictive handover suffers the largest inconsistencies due to the difficulty of accurately predicting the next AP in advance. Providing a reliable handover is particularly important for fast-moving vehicles, where the frequency of AP transitions largely increases the possibility of handover failures.

**Further Exploration of Hybrid Approaches** Hybrid methods have shown the potential to resolve current mobile IP issues, as different methods are often complimentary. However, there still are many possible combinations that have not been fully explored. For example, our predictive method being utilized with HMIP, FMIP, or PMIP methods. In methods that provide low latencies but high overhead, correctly predicting some of them in advance could nullify the overhead issues.

# APPENDICES

# Appendix A

## An Illustrative Example



The example scenarios in Figure A.1 are used to illustrate the proposed prediction's

solution to previous issues. Probability distributions commonly found within realistic road environments show where previous prediction approaches perform poorly, and how the proposed method is able to better predict in the given situation. By reducing the number of problematic scenarios, the reliability and prediction accuracy of the proposed solution is improved.

First, consider the example in Figure A.1a, where probability of moving to region 1,  $p_1$ , and moving to region 3,  $p_3$ , are approximately equal, but with  $p_1$  being slightly higher. If a probability-based algorithm is used, it will always predict region 1, and will be wrong about 50% of the time. Without temporal data to consider, the probability-based approach cannot further distinguish between region 1 and region 2, and thus its performance suffers. If we also include a probability for region 2,  $p_2$ , and assume  $p_1 \approx p_2 \approx p_3 \approx \frac{1}{3}$ , prediction accuracy is even worse. This problematic scenario is resolved with the proposed method's conjunction of temporal and probabilistic data, which increases the information used in the prediction. As shown in Figure A.1b, the additional consideration of the vehicle's movement allows the method to distinguish the otherwise similar options. Observation of the speed and acceleration of the vehicle will reveal when it intends to continue straight, and predict accordingly.

Second, consider the example of Figure A.1c, which illustrates a problematic scenario when only vehicle movement is the measure for determining the next region. If we assume that very few vehicles continue straight to region 3, but report a projection of forward movement, the movement-based approach will still predict region 3. This will result in a very low accuracy close to zero. The lack of probability consideration results in significant performance costs in situations with large turn probabilities and where the movement is misleading of the vehicle's intentions. However, as illustrated in Figure A.1d, our proposed approach is not misled by this movement projection. With the learned probability distribution revealing a small  $p_3$  and the movement projection reporting the vehicle slowing down, the proposed approach will accurately predict region 1. By example of these two scenarios, it is shown that the performance degradation caused by previously problematic scenarios is avoided.

# References

- [1] H. Teng, R.H. Hwang, and C.F. Tsai, "A Context-Aware Seamless Handover Mechanism for Mass Rapid Transit System," in *2011 UIC 8th International Conference*, September 2011, pp. 109-123.
- [2] P. Nath and C. Kumar, "User Movement-Based Hierarchical Mobility Management Scheme (UM-HMIP) for IP Networks," in *Third International Conference on Computer and Communication Technology*, November 2012, pp. 210-215.
- [3] K. Kawano, K. Kinoshita, and K. Murakami, "A Multilevel Hierarchical Distributed IP Mobility Management Scheme for Wide Area Networks," in *Eleventh International Conference on Computer Communications and Networks*, October 2002, pp. 480-484.
- [4] T. You, S. Pack, and Y. Choi "Robust Hierarchical Mobile IPv6(RH-MIPv6): an Enhancement for Survivability and Fault-Tolerance in Mobile IP Systems," in *58th Vehicular Technology Conference*, October 2003, Vol. 3, pp. 2014-2018.
- [5] M. Yi and C. Hwang "A Mobility-Based Binding Update Strategy in Hierarchical Mobile IPv6," <http://dpnm.postech.ac.kr/papers/DSOM/03/118-yunmnmayuy.pdf>.
- [6] E. Mirzamany, A. Lasebae, and O. Gemikonakli, "Using Aggregated RSVP in Nested HMIPv6," in *8th International Wireless Communications and Mobile Computing Conference*, August 2012, pp. 716-721.
- [7] J. Lee, J. Min, K. Park, and S. Kim "Paging Extension for Hierarchical Mobile IPv6: P-HMIPv6," in *11th IEEE International Conference on Networks*, October 2003, pp. 245-248.



- 
- [8] R. Farahbakhsh and N. Movahhedinia, "Seamless Handover for IMS over Mobile-IPv6 Using Context Transfer" in *China Communications*, Vol. 6 Issue 3, July 2009, pp. 122-133.
  - [9] L. Zhang, L. J. Zhang, and S. Pierre, "Performance Analysis of Seamless Handover in Mobile IPv6-based Cellular Networks" in *Cellular Networks - Positioning, Performance Analysis, Reliability*, April 2011.
  - [10] M. Boutabia and H. Afifi, "Maximizing Predictive Mode Probability in Fast Handovers for Mobile IP" in *2013 20th International Conference on Telecommunications*, May 2013, pp. 1-5.
  - [11] A.S. Sadiq, K.A. Bakar, K.Z. Ghafoor, J. Lloret, and R. Khokhar, "An Intelligent Vertical Handover Scheme for Audio and Video Streaming in Heterogeneous Vehicular Networks" in *Mobile Networks and Applications*, December 2013, Vol. 18, Issue 6, pp. 879-895.
  - [12] H. Huang and J. Wu, "A Pre-Binding Update Fast Handover Control Using IEEE 802.21 MIH over 802.16e Networks" in *WRI International Conference on Communications and Mobile Computing*, January 2009, pp. 417-421.
  - [13] H. Kim and Y. Kim, "An Early Binding Fast Handover for High-Speed Mobile Nodes on MIPv6 over Connectionless Packet Radio Link" in *7th International Conference on Software Engineering, Artificial Intelligence, Networking, and Parallel/Distributed Computing*, June 2006, pp. 237-342.
  - [14] H. Kim and K.G. Shin, "Predictive Routing of Contexts in an Overlay Network" *IFIP/IEEE International Symposium on Integrated Network Management*, June 2009, pp. 57-64.
  - [15] A. Bohlooli and K. Jamshidi, "A GPS-Free Method for Vehicle Future Movement Directions Prediction using SOM for VANET," in *Applied Intelligence*, April 2011, Vol.36, Issue 3, pp. 685-697.

- [16] N.V.D. Wijngaert and C. Blondia, "A Predictive Low Latency Handover Scheme for Mobile IP," in *2005 International Conference on Mobile Computing and Ubiquitous Networking*, Vol. 63 2005.
- [17] M. Al Masri, A. Sesay, A. Fapojuwo, "Session State Aware Handover Procedure for VoIP Sessions in Heterogeneous Wireless Networks," in *Wireless Communications and Networking Conference*, April 2014, pp. 3011-3016.
- [18] M. Kyriakakos, N. Frangiadakis, L. Merakos, and S. Hadjiefthymiades, "Enhanced path prediction for network resource management in wireless LANs" *IEEE Wireless Communications*, December 2003, Vol. 10 Issue 6, pp. 62-69.
- [19] A. Mishra, M. Shin, and W.A Arbaush, "Context Caching Using Neighbor Graphs for Fast Handoff in a Wireless Network," in *23rd Annual Joint Conference of the IEEE Computer and Communications Societies*, March 2004, pp. 7-11.
- [20] S. Hadjiefthymiades, S. Papayiannis, and L. Merakos, "Using Path Prediction to Improve TCP Performance in Wireless/Mobile Communications," in *IEEE Communications Magazine*, August 2002, Vol. 40 Issue 8, pp. 54-61.
- [21] Z. Becvar, "Efficiency of Handover Prediction Based on Handover History," in *Journal of Convergence Information Technology*, December 2009, Vol. 4 Issue 4, pp. 41-47.
- [22] I.F. Akyildiz and W. Wang, "The Predictive User Mobility Profile Framework for Wireless Multimedia Networks," in *IEEE/ACM Transactions on Networking*, December 2004, Vol. 12 Issue 6, pp. 1021-1035.
- [23] E. Hernandez and S. Helal, "Predictive Mobile IP for Rapid Mobility," in *29th Annual IEEE International Conference on Local Computer Networks*, November 2004, pp. 16-18.
- [24] G. Jeney, L. Bokor, and Z. Mihaly, "GPS Aided Predictive Handover Management for Multihomed NEMO Configurations," in *9th Intelligent Transport Systems Telecommunications*, October 2009, pp. 69-73.

- 
- [25] 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*, August 2002, Vol. 16 Issue 6, pp. 922-936.
- [26] F. Fang and D.S. Reeves, "Explicit Proactive Handoff with Motion Prediction for Mobile IP," in *Wireless Communications and Networking Conference*, March 2004, Vol. 2, pp. 855-860.
- [27] S. Bhaskar, G. Kumar, P. Reddy, "Design and Analysis of an Adaptive Handover Protocol for 4G Networks," in *Journal of Shanghai Jiaotong University (Science)*, April 2015, Vol. 20, Issue 2, pp. 209-217.
- [28] A.S. Sadiq, K.A. Bakar, K.Z. Ghafoor, J. Lloret, and S.A. Mirjalili, "A Smart Handover Prediction System Based on Curve Fitting Model for Fast Mobile IPv6 in Wireless Networks," in *International Journal of Communication Systems*, July 2014, Vol. 27, Issue 7, pp. 969-990.
- [29] M. Almulla, Y. Wang, A. Boukerche, and Z. Zhang, "Design of a Fast Location-Based Handoff Scheme for IEEE 802.11 Vehicular Networks," in *IEEE Transactions on Vehicular Technology*, October 2014, Vol. 53, Issue 8, pp. 3853-3866.
- [30] E. K. Paik and Y. Choi, "Prediction-based Fast Handoff for Mobile WLANs," in *10th International Conference on Telecommunications*, March 2003, Vol. 1, pp. 748-753.
- [31] S. Kim, D. Kim, J. Kim, D. Han, and B. Park, "A Novel Predictive Binding Update Scheme for Fast Handover Over IP-Based Wireless/Mobile Networks," in *Communications in Computer and Information Science*, September 2011, Vol. 195, pp. 322-330.
- [32] T. Schmidt and M. Wahlisch, "Predictive Versus Reactive—Analysis of Handover Performance and Its Implications on IPv6 and Multicast Mobility," in *Telecommunication Systems*, November 2005, Vol. 30 Issue 1-3, pp. pp 123-142.
- [33] G. Chellani and A. Kalla, "A Review: Study of Handover Performance in Mobile IP," in *International Journal of Computer Networks and Communications*, November 2013, Vol. 5 Issue 6, pp. pp 137-151.

- 
- [34] H. Soliman, C. Castelluccia, K. El-Malki, and L. Bellier, "Hierarchical MIPv6 Mobility Management (HMIPv6)," *Internet Draft, work in progress*, June 2003.
  - [35] G. Tsirtsis, A. Yegin, C. Perkins, G. Dommety, K. El-Malki, and M. Khalil, "Fast Handovers for Mobile IPv6," in *Internet Draft, work in progress*, March 2003.
  - [36] B.B. Dubey, "Effect of position of fixed infrastructure on data dissemination in VANETs," in *International Journal of Research and Reviews in Computer Science (IJRRCS)*, March 2003, Vol. 2, Issue 2.
  - [37] W. Su, S. Lee, and M. Gerla, "Mobility Prediction and Routing in Ad Hoc Wireless Networks," in *International Journal of Network Management*, 2001, pp. 3-30.
  - [38] R. Gunasekaran, A. Rajakumar, K. Raja, J. Ramkumar, S. Kumar, and S. Umapathy, "An Efficient Handover Prediction and Initiation Algorithm for Vehicular Communication in 4G Wireless Networks," in *Lecture Notes on Information Theory*, 2014, Vol. 2, Issue 2, pp. 121-128.
  - [39] N.V. Hanh, S. Ro, and J. Ryu, "Simplified Fast Handover in Mobile IPv6 Networks," in *Computer Communications*, September 2008, Vol. 31, Issue 15, pp. 3594-3603.
  - [40] L. Zhuang, C. Wang, and W. Song, "Research of an Optimized Mobile IPv6 Real-time Seamless Handover Technology" in *Journal of Networks*, Vol. 6 Issue 9, September 2011, pp. 1373-1378.
  - [41] H. Yokota, K. Chowdhury, R. Koodli, B. Patil, and F. Xia, "Fast Handovers for Proxy Mobile IPv6" in *RFC 5949*, September 2010.
  - [42] A. Moravejosharieh and H. Modares, "A Proxy MIPv6 Handover Scheme for Vehicular Ad-hoc Networks" in *Wireless Personal Communications*, March 2014, Vol. 75, Issue 1, pp. 609-626.
  - [43] Y. Wang, Y. Feng, and L. Zhang, "Coordinating Fast Handover and Route Optimization in Proxy Mobile IPv6" in *5th International Conference on Wireless Communications, Networking and Mobile Computing*, September 2009, pp. 1-4.

- [44] S. Moon, M. Kim, S. Lee, D. Cypher, and N. Golmie, "Fast Handover with Lower Latency for Proxy MIPv6 in Vehicular Networks," in *International Conference on Ubiquitous Information management and Communication*, February 2011, Article No. 103.
- [45] C. Huang, M. Chiang, and P. Chau, "A Load-Considered Fast Media Independent Handover Control Scheme for Proxy Mobile IPv6 (LC-FMIH-PMIPv6) in the Multiple-destination Environment," in *IEEE International Black Sea Conference on Communications and Networking*, May 2015, pp. 171-175.
- [46] H. Park, Y. Kwon, K. Lee, Y. Choi, S. Lee, and Y. Cho, "Network Mobility Management Using Predictive Binding Update" *Distributed Computing-IWDC 2005*, December 2005, pp. 560-565.
- [47] G. Yavas, D. Katsaros, O. Ulusoy, and Y. Manolopoulos, "A Data Mining Approach for Location Prediction in Mobile Environments," in *Data & Knowledge Engineering*, August 2005, Vol.54, Issue 2, pp. 121-146.
- [48] P. Fazio, M. Tropea, and S. Marano, "A Distributed Hand-over Management and Pattern Prediction Algorithm for Wireless Networks with Mobile Hosts" *9th International Wireless Communications and Mobile Computing Conference*, July 2013, pp. 294-298.
- [49] W. Hu, X. Xiao, D. Xie, T. Tan, and S. Maybank, "Traffic Accident Prediction Using 3-D Model-Based Vehicle Tracking," in *IEEE Transactions on Vehicular Technology*, May 2004, Vol.54, No. 3, pp. 677-694.
- [50] S. Pack and Y. Choi, "Fast Handoff Scheme Based on Mobility Prediction in Public Wireless LAN Systems," in *IEEE Proceedings- Communications*, October 2004, Vol. 151, Issue 5, pp. 489-495.
- [51] P. Lai, X. Wang, N. Lu, and F. Liu, "A Reliable Broadcast Routing Scheme Based on Mobility Prediction for VANET," in *IEEE Intelligent Vehicles Symposium*, June 2009, pp. 1083-1087.
- [52] S. Gundavelli, K. Leung, V. Devarapalli, K. Chowdhury, and B. Patil, "Proxy Mobile IPv6," *IETF Internet Draft, work in progress*, August 2008.

- 
- [53] K. Lee, W. Seo, Y. Cho, J. Kim, J. Park, and B. Moon, "Inter-Domain Handover Scheme using an Intermediate Mobile Access Gateway for Seamless Service in Vehicular Networks," in *International Journal of Communication Systems*, October 2010, Vol. 23, Issue 9-10 pp. 1127-144.
- [54] S. Ro and V. Nguyen, "Inter-Domain Mobility Support in Proxy Mobile IPv6 Using Overlap Function of Mobile Access Gateway," in *Wireless Networks*, April 2015, Vol. 21, Issue 3, pp. 899-910.
- [55] N. Neumann, J. Lei, X. Fu, and G. Zhang, "I-PMIP: an Inter-Domain Mobility Extension for Proxy-Mobile IP," in *The International Wireless Communications and Mobile Computing Conference*, June 2009, pp. 994-999.
- [56] T. Chiba, H. Yokota, A. Dutta, D. Chee, and H. Schulzrinne, "Route Optimization for Proxy Mobile IPv6 in IMS Network," in *2nd International Conference on Signal Processing and Communication Systems*, December 2008, pp. 1-9.
- [57] H. Jung, M. Gohar, J. Kim, and S. Koh, "Distributed Mobility Control in Proxy Mobile IPv6 Networks," in *IEICE Transactions on Communications*, August 2011, Vol. 94, Issue 8, pp. 2216-2224.
- [58] S. Son, S. Jeong, J. Choi, and J. Jeong, "On Mobility-Aware Dual Pointer Forwarding Handoff Scheme in Cost-Optimized Proxy Mobile IPv6 Networks," in *Ubiquitous Information Technologies and Applications*, 2014, Vol. 280, pp. 649-657.
- [59] F. Lassabe, P. Canalda, P. Chatonnay, and F. Spies, "Predictive Mobility Models based on Kth Markov Models," in *IEEE International Conference on Pervasive Services*, June 2006, pp 303-306.
- [60] Y. Wu, Y. Zhu, and B. Li, "Trajectory Improves Data Delivery in Vehicular Networks," in *2011 Proceedings IEEE INFOCOM*, April 2011, pp 2183-2191.
- [61] H. Jeung, M.L. Yiu, X. Zhou, and C.S. Jensen, "Path Prediction and Predictive Range Querying in Road Network Databases," in *The VLDB Journal*, August 2010, Vol. 19, Issue 4, pp 585-602.

- 
- [62] R. Hsieh, Z.G. Zhou, and A. Seneviratne, "S-MIP: A Seamless Handoff Architecture for Mobile IP," in *2003 Proceedings IEEE INFOCOM*, April 2003, Vol. 3, pp 1774-1784.
- [63] Z. Zhang, R. Pazzi, and A. Boukerche, "A Mobility Management Scheme for Wireless Mesh Networks Based on a Hybrid Routing Protocol," in *Advances in Wireless and Mobile Networks*, March 2010, Vol. 54, Issue 4 pp. 558-572.
- [64] Z. Zhang, A. Boukerche, and H. Ramadan "Design and Evaluation of a Fast MAC Layer Handoff Management Scheme for WiFi-based Multichannel Vehicular Mesh Networks," in *Journal of Network and Computer Applications*, May 2013, Vol. 36, Issue 3, pp. 992-1000.
- [65] J. Haerri, F. Filali, and C. Bonnet, "Performance Comparison of AODV and OLSR in VANETs Urban Environments under Realistic Mobility Patterns," in *Proceedings of 5th IFIP Mediterranean Ad-Hoc Networking Workshop*, 2006.
- [66] R. E. Kalman, "A New Approach to Linear Filtering and Prediction Problems," in *Transaction of the ASME-Journal of Basic Engineering*, March 1960, pp 35-45.
- [67] L.E. Baum and T. Petrie, "Statistical Inference for Probabilistic Functions of Finite State Markov Chains," in *The Annals of Mathematical Statistics*, 1966, Vol. 37, Issue 6, pp 1554-1563.
- [68] O. Cappe, "Online EM Algorithm for Hidden Markov Models," in *J. Computer Graphic Statistics*, 2011, Vol. 20, Issue 3, pp 728-749.
- [69] G. Mongillo and S. Deneve, "Online Learning with Hidden Markov Models," in *Neural Computation*, 2008, Vol. 20, No. 7, pp 1706-1716.
- [70] W. Khreich, E. Granger, A. Miri, and R. Sabourin, "A Survey of Techniques for Incremental Learning of HMM Parameters," in *Information Sciences*, 2012, Vol. 197, pp 105-130.
- [71] D. Krajzewicz, G. Hertkorn, C. Rossel, and P. Wagner, "SUMO(Simulation of Urban Mobility), An Open-source Traffic Simulation," *4th Middle East Symposium on Simulation and Modeling*, 2002, pp 183-187.

- 
- [72] "The Network Simulator - ns-2," <http://www.isi.edu/nsnam/ns/>.
- [73] "OpenStreetMap," <https://www.openstreetmap.org/>.
- [74] W. Bao and B. Liang, "Handoff Rate Analysis in Heterogeneous Cellular Networks: a Stochastic Geometric Approach" in *17th ACM Conference on Modeling, Analysis, and Simulation of Wireless and Mobile Systems*, September 2014, pp 95-102.
- [75] B. D. O. Anderson, "From Wiener to Hidden Markov Models," in *IEEE Control Systems*, 1999, Vol. 19, Issue 3, pp 41-51.
- [76] W. Shuying and W. Shanpei, "Hidden Markov Model with Noise-Adaptive Codebook for Noisy Speech Recognition," in *TENCON*, 1993, Vol. 3, pp 325-327.
- [77] D. Dailey, Z. Wall, S. Maclean, and F. Cathey, "An Algorithm and Implementation to Predict the Arrival of Transit Vehicles," in *IEEE ITSC*, October 1990, pp 5-8.
- [78] E. Charniak, *Statistical Language Learning*. Cambridge, Massachusetts: MIT Press, 1993.
- [79] C. Perkins, E. Belding-Royer, and S. Das, "Ad hoc On-Demand Distance Vector(AODV) Routing," *RFC 3561*, IETF Network Working Group, July 2003.
- [80] T. Clausen and P. Jacquet, "Optimized Link State Routing Protocol(OLSR)," *RFC 3626*. IETF Network Working Group, October 2003.
- [81] B. Karp and T. Kung, "GPSR: Greedy Perimeter Stateless Routing for Wireless Networks," *MobiCom 2000*, August 2000.
- [82] M. Ayash, M. Mikki, and K. Yim, "Improved AODV Routing Protocol to Cope with High Overhead in High Mobility MANETs" in *Conference on Innovative Mobile and Internet Services in Ubiquitous Computing*, July 2012, pp 244-251.
- [83] J. Toutouh, J. Garcia-Nieto, and E. Alba, "Intelligent OLSR Routing Protocol Optimization for VANETs," in *IEEE Transactions on Vehicular Technology*, May 2012, pp 1884-1894.



- 
- [84] S. Singh and S. Agrawal, "VANET Routing Protocols: Issues and Challenges," in *Recent Advances in Engineering and Computational Sciences*, March 2014, pp 1-5.
- [85] R. Bala and C.R. Krishna, "Scenario Based Performance Analysis of AODV and GPSR Routing Protocols in a VANET," in *Conference on Computational Intelligence and Communication Technology*, February 2015, pp 432-437.
- [86] U. Jonsson, F. Alriksson, T. Larsson, P. Johansson, and G. Maguire, "MIPMANET: Mobile IP for Mobile Ad Hoc Networks," in *Mobile and Ad Hoc Networking and Computing*, 2000, pp 75-85.
- [87] A. Boukerche and M. Notare, "Behavior-based Intrusion Detection in Mobile Phone Systems," in *Journal of Parallel and Distributed Computing*, 2002, Vol. 62, No. 9, pp 1476-1490.
- [88] A. Boukerche, R. Pazzi, and R. Araujo, "Fault-tolerant Wireless Sensor Network Routing Protocols for the Supervision of Context-aware Physical Environments," in *Journal of Parallel and Distributed Computing*, 2006, Vol. 66, No. 4, pp 586-599. 1476-1490.
- [89] A. Boukerche and X. Li, "An Agent-based Trust and Reputation Management Scheme for Wireless Sensor Networks," in *IEEE Global Telecommunications Conference*, 2005.
- [90] S. Samarah, M. Al-Hajri, and A. Boukerche, "A Predictive Energy-efficient Technique to Support Object-tracking Sensor Networks," in *IEEE Transactions on Vehicular Technology*, 2011, Vol. 60, No. 2, pp 656-663.
- [91] A. Boukerche, X. Fei, and R. Araujo, "An Optimal Coverage-preserving Scheme for Wireless Sensor Networks Based on Local Information Exchange," in *Computer Communications*, 2007, Vol. 30, No. 14, pp 2708-2720.
- [92] A. Boukerche, K. Juca, J. Sobral, and M. Notare "An Artificial Immune Based Intrusion Detection Model for Computer and Telecommunication Systems," in *Parallel Computing*, 2004, Vol. 30, No. 5, pp 629-646.

- 
- [93] A. Boukerche, S. Hong, and T. Jacob, "A Distributed Algorithm for Dynamic Channel Allocation," in *Mobile Networks and Applications*, 2002, Vol. 7, No. 2, pp 115-126.
  - [94] A. Boukerche, S. Hong, and T. Jacob, "An Efficient Synchronization Scheme of Multimedia Streams in Wireless and Mobile Systems," in *IEEE Transactions on Parallel and Distributed Systems*, Vol. 13, No. 9, pp 911-923.
  - [95] A. Boukerche and L. Bononi, "Simulation and Modeling of Wireless, Mobile, and ad hoc Networks," in *Mobile ad hoc Networks*, 2004, pp 373-409.
  - [96] A. Boukerche, R. Pazzi, and R. Araujo, "HPEQ, A Hierarchical Periodic Event-driven and Query-based Wireless Sensor Network Protocol," in *IEEE Conference on Local Computer Networks*, 2005, pp 560-567.
  - [97] R. Batista, A. Boukerche, and A. Melo, "A Parallel Strategy for Biological Sequence Alignment in Restricted Memory Space," in *Journal of Parallel and Distributed Computing*, 2008, Vol. 68, No. 4, pp 548-561.
  - [98] A. Boukerche and C. Tropper, "A Distributed Graph Algorithm for the Detection of Local Cycles and Knots," in *IEEE Transactions on Parallel and Distributed Systems*, 1998, Vol. 9, No. 8, pp 748-757.
  - [99] A. Boukerche, S. Das, A. Fabbri, and O. Yildiz, "Exploiting Model Independence for Parallel PCS Network Simulation," in *13th Workshop on Parallel and Distributed Simulation*, 1999, pp 166-173.
  - [100] T. Antoniou, I. Chatzigiannakis, G. Mylonas, S. Nikoletseas, and A. Boukerche, "A New Energy Efficient and Fault-tolerant Protocol for Data Propagation in Smart Dust Networks Using Varying Transmission Range," in *37th Annual Symposium on Simulation*, 2004.
  - [101] A. Boukerche and X. Fei, "A Veroni Approach for Coverage Protocols in Wireless Sensor Networks," in *IEEE Global Telecommunications Conference*, 2007, pp 5190-5194.

- 
- [102] Y. Ren and A. Boukerche, "Modeling and Managing the Trust for Wireless and Mobile ad hoc Networks," in *IEEE International Conference on Communications*, 2008, pp 2129-2133.
- [103] A. Boukerche and S. Nikolettseas, "Protocols for Data Propagation in Wireless Sensor Networks," in *Wireless Communications Systems and Networks*, 2004, pp 23-51.
- [104] A. Boukerche and X. Fei, "A Coverage-preserving Scheme for Wireless Sensor Network with Irregular Sensing Range," in *Elsevier ad hoc Networks*, 2007, Vol. 5, No. 8, pp 1303-1316.
- [105] A. Bamis, A. Boukerche, I. Chatzigiannakis, and S. Nikolettseas, "A Mobility Aware Protocol Synthesis for Efficient Routing in ad hoc Mobile Networks," in *Computer Networks*, 2008, Vol. 52, No. 1, pp 130-154.
- [106] O. Abumansoor and A. Boukerche, "A Secure Cooperative Approach for Nonline-of-sight Location Verification in VANET," in *IEEE Transactions on Vehicular Technology*, 2012, Vol. 61, No. 1, pp 275-285.
- [107] A. Boukerche, S. Das, and A. Fabbri, "SWiMNet: A Scalable Parallel Simulation Testbed for Wireless and Mobile Networks," in *Wireless Networks*, 2001, Vol. 7, No. 5, pp 467-486.
- [108] A. Boukerche and S. Samarah, "A Novel Algorithm for Mining Association Rules in Wireless ad hoc Sensor Networks," in *IEEE Transactions on Parallel and Distributed Systems*, 2008, Vol. 19, No. 7, pp 865-877.
- [109] A. Boukerche, X. Cheng, and J. Linus, "A Performance Evaluation of a Novel Energy-aware Data-centric Routing Algorithm in Wireless Sensor Networks," in *Wireless Networks*, 2005, Vol. 11, No. 5, pp 1476-1490.
- [110] A. Boukerche, S. Das, and A. Fabbri, "Analysis of a Randomized Congestion Control Scheme with DSDV Routing in ad hoc Wireless Networks," in *Journal of Parallel and Distributed Computing*, 2001, Vol. 61, No. 7, pp 967-995.

- [111] A. Boukerche and Y. Ren, "A Trust-based Security System for Ubiquitous and Pervasive Computing Environments," in *Computer Communications*, 2008, Vol. 31, No. 18, pp 4343-4351.
- [112] Y. Ren, R. Pazzi, and A. Boukerche, "Monitoring Patients via a Secure and Mobile Healthcare System," in *IEEE Wireless Communications*, 2010, Vol. 17, No. 1, pp 59-65.
- [113] L. Villas, A. Boukerche, H. Ramos, H. Oliveira, R. Araujo, and A. Loureiro, "DRINA: A Lightweight and Reliable Routing Approach for In-network Aggregation in Wireless Sensor Networks," in *IEEE Transactions on Computers*, 2013, Vol. 62, No. 4, pp 676-689.
- [114] A. Boukerche, "Handbook of Algorithms for Wireless Networking and Mobile Computing," in *CRC Press*, 2005.
- [115] A. Boukerche, "Algorithms and Protocols for Wireless, Mobile ad hoc Networks" in *John Wiley & Sons*, 2008.
- [116] A. Boukerche, H. Oliveira, E. Nakamura, and A. Loureiro, "Localization Systems for Wireless Sensor Networks," in *IEEE Wireless Communications*, 2007, Vol. 14, No. 6, pp 6-12.