Chapter 7

FORENSIC TRACKING AND MOBILITY PREDICTION IN VEHICULAR NETWORKS

Saif Al-Kuwari and Stephen Wolthusen

Abstract   Vehicular networks have attracted significant attention in the context of traffic control and management applications. However, vehicular networks also have important applications in crime prevention and criminal investigations. This paper presents a system for passively tracking a target vehicle whose driver is assumed to be a "person of interest." The tracking system relies on the dynamic recruitment of neighboring vehicles of the target as agents. A mobility prediction algorithm is used to probabilistically predict the target's future movement and to adjust the tracking process. Combining agent-based tracking and mobility prediction enables a target vehicle to be passively localized and tracked in an efficient manner.

Keywords: Vehicular networks, passive tracking, mobility prediction

1.   Introduction

Due to the heightened interest in driver safety and infotainment applications, vehicular networks along with other mature wireless technologies will be widely deployed in future automobiles. This has motivated the research community to investigate various aspects of vehicular networks.

The main motive behind the emergence of vehicular networks is driver safety. However, these networks have important applications in crime prevention and criminal investigations in which law enforcement agencies must track the movements of "persons of interest."

Vehicular network algorithms differ from those encountered in conventional mobile networks. Vehicular network nodes move more rapidly than the nodes in other mobile networks. However, vehicular nodes are characterized by somewhat limited freedom of motion since their move-

ments are constrained by roadways and traffic regulations. This introduces additional challenges for vehicular tracking algorithms because they must efficiently adapt to the peculiarities of vehicular movements; but this also improves their mobility predictions because nodes move along predefined paths.

This paper presents a passive vehicular tracking system that relies on the dynamic recruitment of neighboring vehicles as agents. A mobility prediction algorithm is used to probabilistically predict the target's future movement and to adjust the tracking process. Combining agent-based tracking and mobility prediction enables a target vehicle to be localized and tracked both efficiently and clandestinely.

2.   Related Work

Boukerche, *et al.* [3] discuss localization techniques that can be used in vehicular networks along with their practical implications. The authors demonstrate that most localization techniques suffer from inherent inaccuracies that may not be acceptable for vehicular-based applications that require precise location information. In such situations, the best solution is to use data fusion where the results of multiple localization techniques are combined to increase accuracy.

While localization techniques for vehicular networks are usually GPS-based, not all vehicles are equipped with GPS devices. Also, GPS-based techniques are useless when GPS signals are not available (e.g., inside tunnels). Benslimane [2] addresses this problem in an extension to the ODAM messaging dissemination protocol, enabling vehicles that are not GPS-capable to be localized.

A popular tracking technique that is well suited to vehicular scenarios involves map-matching tracking, which attempts to match a node's actual location (raw) data to maps. Barakatsoulas, *et al.* [4] present several such algorithms that exploit vehicular trajectory information. However, this approach is not suitable for applications that require real-time tracking. Other tracking techniques such as installing tracking tags on a target vehicle [9] also exist, but these solutions are not ad hoc and require additional preparation and overhead.

3.   Vehicular Networks

Generally, vehicular networks are based on ad hoc infrastructures and are, therefore, referred to as vehicular ad hoc networks (VANET), a subclass of mobile ad hoc networks (MANETs). The U.S. Federal Communications Commission (FCC) has allocated a 75 MHz spectrum in the 5.9 GHz band (5.850 GHz to 5.925 GHz) for vehicular communications,

which can either be vehicle to vehicle (V2V) or vehicle to infrastructure (V2I) communications. V2V depends on an ad hoc infrastructure where vehicles directly exchange information like accident and congestion warnings. V2I, on the other hand, assumes the presence of pre-installed road components that vehicles communicate with in order to retrieve information. However, the cost of installing these components often limits V2I applications.

Communication in vehicular networks is based on the conventional IEEE 802.11 wireless standard (WiFi) or 3G via CDMA (code division multiple access) technology. WiFi is simpler and less expensive to deploy, but has lower reliability because it was not designed to operate in environments with rapid movements. In contrast, 3G is more relevant to vehicular communications, but is more expensive and difficult to deploy because of its centralized architecture [8]. In this paper, we assume that vehicles communicate using WiFi, which is a more widely deployed solution. Note that WiFi has some efficiency limitations, especially because it operates in the half-duplex mode (i.e., two nodes cannot communicate with each other simultaneously).

4. Mobility in Vehicular Networks

Mobility models represent the motion behavior of vehicles. Mobility models designed to simulate the movement of nodes in MANETs can be used to simulate vehicular movements in VANETs and most vehicular simulation approaches employ these models. However, MANET models have proved unreliable and unrealistic, which has motivated the design and development of mobility models targeted for vehicular networks.

Vehicular mobility models are categorized as microscopic, mesoscopic and macroscopic models. Microscopic models describe the dynamics of the individual movement of each vehicle and its interaction with other vehicles (this may require excessive computational resources for large-scale simulations). Macroscopic models simulate the characteristics of the roadways (motion constraints) and the flow of vehicles, but do not consider individual vehicle movement, which impacts their degree of realism. Interested readers are referred to [1] for a discussion of macrosopic models versus microscopic models. Mesoscopic models belong to a new "hybrid" class of vehicular mobility models that combine features of microscopic and macroscopic models. In this paper, we adopt the microscopic intelligent driver motion (IDM) model [10] along with two extensions, IDM-IM (IDM with intersection management) and IDM-LC (IDM with lane changes).

5.   Vehicle Localization

Conventional localization techniques used in cellular and sensor networks can also be applied to vehicular networks, albeit with slightly degraded accuracy. These techniques involve measuring the received signal strength (RSS), time of arrival (TOA) or time difference of arrival (TDOA). Since we adopt a passive tracking approach, only RSS is relevant because, unlike TOA and TDOA, it does not require a fully synchronized network and active communication with the target.

The RSS technique involves measuring the strength of the signals emitted from the target vehicle in order to estimate the distance to the target. RSS measurements are nonlinear due to radio propagation effects; nevertheless, our algorithm adopts a free space radio propagation model, which assumes a direct line of sight path between vehicles. This assumption does not severely degrade the accuracy of the localization process because it is most likely that the only obstacles would be the moving vehicles and their effects can be eliminated by averaging the measurements. Note that vehicular acceleration also affects RSS measurements, but we assume that this effect can be mitigated by adding noise in the computations.

The localization of a target vehicle C_s involves three of its neighboring vehicles. However, only two of these neighbors (tracking agents) track the target; the other (localizing agent) is retired immediately after the localization is completed. We denote the tracking agents as C_a (main) and C_b (backup) and the localizing agent as C_l. We assume that the three agents can localize themselves using GPS, which is fairly common in current vehicles.

The localization algorithm has four steps:

- **Step 1 (Recruitment):** The tracking agents are recruited by randomly requesting RSS readings from nodes around the target. The nodes with the highest readings are recruited. Details about the initial and subsequent recruitment process are provided in Section 7.

- **Step 2 (Distance Measurement):** The distances between the recruited agents and the target are estimated using the Friis equation [6]:

$$d = \sqrt{P_t A_r / 4\pi P_r}$$

  where P_r and P_t are the received and transmitted power, respectively; and A_r is the effective area of the receiving antenna.

- **Step 3 (Trilateration):** The location of the target is estimated using a geometric transformation to find the intersection point of

the three circles centered at the agents that have radii equal to the respective distances of the agents from the target as measured in Step 2.

- **Step 4 (Retirement):** The localizing agent C_l is retired.

The localization algorithm can also be used in conjunction with the prediction algorithm (Section 6) to determine when the next recruitment is required. This assumes that the agent can measure its own speed, which is possible because it is equipped with a GPS device. The average speed is estimated by dividing a particular distance by its corresponding traversal time.

6.   Mobility Prediction

Mobility prediction is an important aspect to consider when tracking vehicles. Generally, mobility prediction in vehicular networks is slightly simpler in VANETs than MANETs because vehicular mobility is restricted to roadways and the pace of movement is usually limited to the maximum allowable speed of the roadway. The motion of mobile nodes in MANETs, on the other hand, is often unconstrained, although it may be restricted by terrain characteristics and physical limitations. Note also that vehicular movements have higher accelerations.

This section presents a vehicular mobility prediction algorithm that probabilistically predicts the near-future movement of the target vehicle based on its current location and estimated speed, assuming that the target has already been localized. The algorithm incorporates: (i) time prediction, which estimates the time elapsed before the target reaches the next intersection; and (ii) direction prediction, which identifies the direction that the target will most likely take after passing the intersection.

6.1   Time Prediction

Predicting the time taken for a vehicle to reach the next intersection requires an estimate of the speed of the vehicle. The estimation procedure is illustrated in Figure 1. Vehicle C_a estimates the speed of C_s by making two RSS measurements at times t_1 and t_2. For reasons of simplicity, it is assumed that the distance between the horizontally-aligned vehicles in a roadway is I, which is easily obtained. However, it is very unlikely that vehicles C_a and C_s will be aligned perfectly. Consequently, when the first RSS measurement (RSS_1) is taken, the vertical distance ahead or behind C_a is calculated to be perfectly adjacent to C_s. The
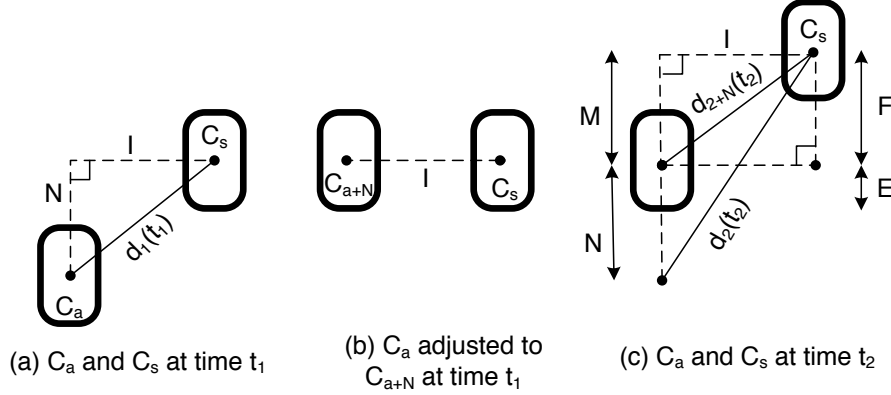
(a) C_a and C_s at time t_1 (b) C_a adjusted to C_{a+N} at time t_1 (c) C_a and C_s at time t_2

Figure 1. Estimating the speed of vehicle C_s.

position of C_a is then adjusted so that it is aligned with C_s using:

$$N = \sqrt{(d_1(t_1))^2 - I^2}$$

as shown in Figure 1(a). Thus, the position C_a becomes C_{a+N} as shown in Figure 1(b).

After the position of C_a is adjusted, the second RSS measurement (RSS_2) is taken at time t_2. However, the distance obtained from RSS_2 must be adjusted by recalculating it as if it was measured by C_{a+N}, which accounts for the extra distance N.

Figure 1(c) illustrates this process. First, the distance M is calculated as:

$$M = \sqrt{(d_2(t_2))^2 - I^2} - N$$

and thus,

$$d_{2+N}(t_2) = \sqrt{M^2 + I^2}.$$

Next, the speed of C_s is computed as:

$$v_s = \frac{F + E}{t_2 - t_1} \tag{1}$$

where $F = \sqrt{(d_{2+N}t_2))^2 - I^2}$ and $E = v_a(t_2 - t_1)$ assuming that the speed v_s of C_a is known (v_s is estimated easily).

After the speed is estimated, the time taken for C_s to reach the next intersection is calculated using the traditional distance equation:

$$t_z = n - d_x + \epsilon/v_s$$

where d_x is the distance the vehicle has covered on a roadway of length n and can be calculated relative to the location of C_a; ϵ is a uniformly distributed random variable introduced to compensate for the acceleration/deceleration error margin; and v_s is the estimated speed of vehicle C_s according to Equation 1.

It is very unlikely that vehicles C_a and C_s have perfect horizontal alignment; thus, we assume that one vehicle is leading or lagging the other. In Figure 1, vehicle C_s is leading C_a and is moving faster than C_a, so it will continue to lead vehicle C_a. Note that the algorithm accommodates other scenarios where C_a is the leading vehicle. This is because the absolute velocity of C_s is measured without considering its relation to the location and speed of C_a.

Since localization is performed before the prediction algorithm is executed, the lane that the target occupies is known. The discussion above assumes that C_a and C_s are not in the same lane, but the probability of them being in the same lane is equally likely. If this is the case and if C_a makes the first RSS measurement (RSS_1) and later makes the second RSS measurement (RSS_2), then the distance C_s has moved during the period between these two measurements is simply the difference between RSS_2 and RSS_1. Moreover, if the vehicles are not in the same lane, it does not matter which vehicle is in which lane. In Figure 1, it happens that C_a is in the left lane and C_s is in the right lane, but the algorithm still works if the lane positions are reversed. However, to simplify the design of the algorithm, we assume that the leading entity will continue to lead during the period between the RSS_1 and RSS_2 measurements. This is a reasonable assumption because the time period is usually short enough to preserve the leading status.

6.2   Direction Prediction

Predicting the direction that a target vehicle will take through the next intersection is more complex and bears a probabilistic distribution. Also, it presupposes that the target vehicle has been accurately localized in order to identify the lane it occupies. We assume that all drivers adhere to the following basic traffic rules as illustrated in Figure 2:

- **Rule 1:** For a vehicle to turn right at an intersection, it must be in the right lane of the roadway leading to the intersection.

- **Rule 2:** For a vehicle to turn left at an intersection, it must be in the left lane of the roadway leading to the intersection.
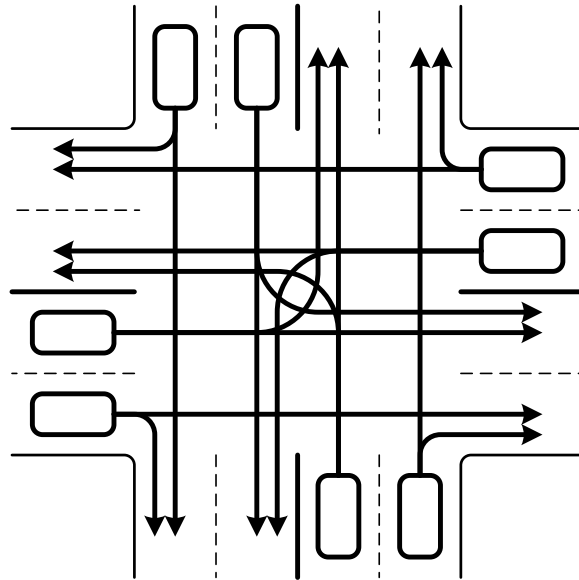
Figure 2.   Traffic rules at an intersection.

■ **Rule 3:** For a vehicle to go straight ahead at an intersection, it can be in the right lane or in the left lane of the roadway leading to the intersection.

■ **Rule 4:** A vehicle must position itself in the appropriate lane based on its intended direction and Rules 1–3 around 300 meters before the next intersection or halfway through the roadway leading to the next intersection, whichever comes later. A lane change beyond this point for the purpose of accelerating the pace (e.g., to overtake a vehicle) is not permitted.

It is reasonable to assume that the target vehicle will obey these traffic rules because the target (who is either a suspect or criminal) will most likely not want to attract attention. Obeying Rules 1–4 forces a vehicle to start the lane changing process when reaching approximately $n/4$ of a roadway of length n or 400 meters away from the intersection, whichever is later. This should allow sufficient time for the vehicle to complete the lane changing process without violating Rule 4. We call the region between $n/4$ and $n/2$ (or similarly, the 100 meters between 400 meters to 300 meters away from the intersection) a "critical area" and apply the following predictions:

$$IF \qquad C_i^L \xrightarrow{CriticalArea} C_i^R \qquad THEN \qquad Pr(C_i \to R) = 80\%$$

$$IF \qquad C_i^R \xrightarrow{CriticalArea} C_i^L \qquad THEN \qquad Pr(C_i \to L) = 70\%$$

Based on these predictions, if a particular vehicle C_i occupied the left lane (C_i^L) and shifted to the right lane as it passed through the critical area, the algorithm infers that the shift was imposed by traffic regulations. This means that the vehicle is very likely going to turn right (R) at the next intersection because: (i) the probability of turning left (L) from a right lane (where the vehicle has shifted) is 0; and (ii) if the vehicle intended to go straight ahead (A), it would most likely have stayed in its original lane without taking the overhead of a lane change as it is permitted to go straight ahead from either lane.

A similar argument applies to a vehicle that shifted from the right lane to the left lane as it passed through the critical area. The difference is that it is possible that the vehicle shifted from the right lane to the left lane to increase its pace of the movement while actually intending to go straight ahead. This decreases the probability of turning left for this particular course of action.

On the other hand, if the vehicle did not change its lane as it went through the critical area, it has a probability of 50% of taking either of the two permitted directions depending on its current lane. Also, it has a 0% probability of taking the banned direction (i.e., left for the right lane and right for the left lane).

An interesting extension to this direction prediction technique (which is not currently implemented in the algorithm) is to integrate points of interest [5] corresponding to frequently-visited locations such as grocery stores, banks, restaurants and offices. Such locations can significantly influence the probability distribution of vehicular mobility. For example, we know that the probability of a vehicle in the right lane turning right is (currently) the same as the probability of the vehicle going straight ahead if it had not shifted to the right lane as it passed through the critical area. However, that fact that a point of interest is located on the right could increase the probability of the vehicle turning right instead of going straight ahead. Note that the specific time of day can significantly influence the effect of a point of interest on the prediction probabilities (e.g., an office building will most likely be a popular point of interest only during the daytime).

7.   Tracking

In forensic and law enforcement applications, vehicle tracking is usually required to be passive so that the driver of the target vehicle is not aware of the tracking process. This passivity requirement potentially eliminates more active tracking systems such as those involving vehicle telematics.

In our tracking scenario, a group of trackers $C_{t(i)}$ $(i = 1, 2, ..., n)$ consisting of n privately-connected vehicles (e.g., police patrols) recruit a main tracking agent C_a and a backup tracking agent C_b from the public to track a target vehicle C_s. We assume that the recruiting $C_{t(i)}$ is initially located in range of C_s, which enables it to recruit suitable C_a and C_b possibly by visual estimation. During this initial stage, the $C_{t(i)}$ also recruits an additional localization agent C_l which, along with the tracking agents, localize the target as discussed in Section 5. The recruiting $C_{t(i)}$ supplies an address list of the other valid $C_{t(i)}$ to all the recruited agents.

The localizing agent is retired after the target is localized, leaving C_a to be responsible for the rest of the tracking. C_a is backed up by C_b, whose responsibility is to take over the tracking process if C_a suddenly fails. C_a is also responsible for sending location updates of C_s to a $C_{t(i)}$ whenever localization is triggered. C_a must ensure that its backup agent C_b is within range of C_a and the target at all times by regularly monitoring its RSS and probing it for the target's RSS; otherwise it has to recruit another C_b.

When tracking is initiated, the $C_{t(i)}$ creates a tracking table $TrackTB$, which stores records of the target's movements as received from C_a. This table is synchronized with all other $C_{t(i)}$ as soon as an update is received by a $C_{t(i)}$. When a localization update is available, C_a searches its range for a $C_{t(i)}$ to update it; if multiple $C_{t(i)}$ are found, C_a randomly chooses one of them; this potentially minimizes a traffic analysis attack by a third observer who observes the traffic between the agents and the trackers. If no $C_{t(i)}$ is available, C_a creates a temporary tracking table $TempTB$ and accumulates information while regularly probing for a valid $C_{t(i)}$. When a $C_{t(i)}$ is found, C_a transfers its $TempTB$ to the C_{ti} which, in turn, merges it with its copy of $TrackTB$ and synchronizes it with the other $C_{t(i)}$.

After running the prediction algorithm, C_a estimates how long it will be able to track the target (called the "alive period") and schedules the next localization for near the expiration of this period. During the alive period, C_a keeps observing the RSS measurements from C_s. If a specific lower threshold of RSS is reached or the predicted alive time is near expi-

Algorithm 1 Vehicular Tracking Algorithm

1: $C_{t(i)} \Leftarrow$ Trackers {identify n trackers}
2: $C_s \Leftarrow$ Target {identify the target}
3: $Recruit(C_a, C_b)$ {recruit tracking agents}
4: $Recruit(C_l)$ {recruit localization agent}
5: **repeat**
6:    $Localize(C_s)$ {C_a, C_b, C_l Localize C_s}
7:    $Retire(C_l)$
8:    $Alive \leftarrow Predict(C_s)$ {find alive period based on the prediction algorithm}
9:    **while** $currentTime \leq Alive - \omega$ **do**
10:      **if** $C_a.RSS(C_s) \leq$ threshold **then**
11:        break {if RSS from $C_s \leq$ threshold, break}
12:      **end if**
13:    **end while**
14:    C_a.Probe {search for agents}
15:    **if** $C_a.RSS(C_s) \leq$ threshold **then**
16:      **if** $C_b.RSS(C_s) \leq$ threshold **then**
17:        $Recruit(C_a, C_b,)$ {recruit new C_a and C_b}
18:      **end if**
19:      $Recruit(C_a)$ {recruit new C_a}
20:    **else**
21:      $Recruit(C_l)$ {recruit C_l}
22:    **end if**
23: **until** Tracking Expires

---

ration, C_a initiates a "probe process," which involves sending requests to the neighboring vehicles, supplying the C_s address and asking for their RSS measurements if they can receive transmissions from C_s. Note that the vehicle with the strongest RSS is recruited as the localization agent C_l. Furthermore, a $C_{t(i)}$ may at any time request C_a to localize C_s, in which case C_a executes the probe process to recruit C_l and works with C_b to localize C_s.

If C_a had to maintain a $TempTB$, but attempted to recruit another C_a (due to the expiry of the alive period or weak RSS observations from C_s) and there is still no $C_{t(i)}$ in range, the old C_a recruits a new C_a and hands off the tracking process to it along with the $TempTB$. The $TempTB$ is maintained by the new C_a until a valid $C_{t(i)}$ is found upon which time the old C_a retires itself. Details of the tracking process are presented in Algorithm 1.
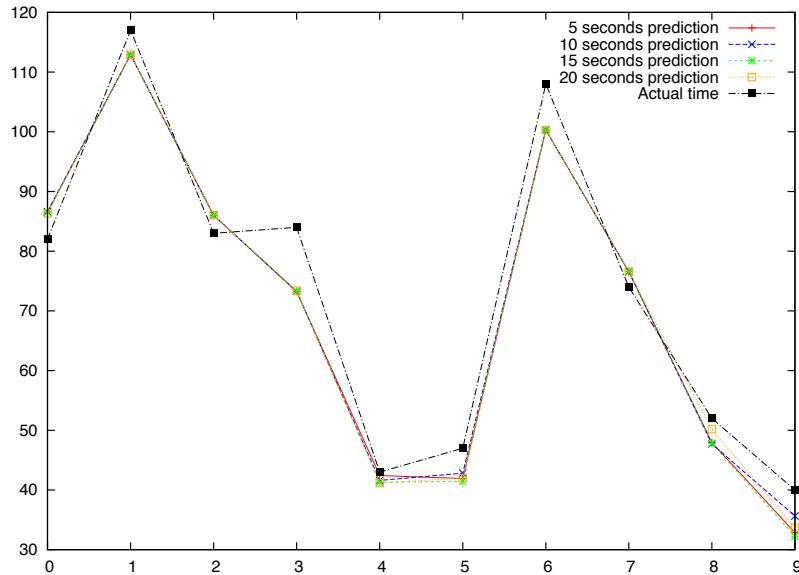
Figure 3.   Simulation results (10 node scenario).

8.   Simulation Results

A realistic vehicular environment was created using the VanetMo-biSim simulator [7]; vehicular mobility traces were generated based on IDM-LC model. The ns-2 simulator was used to run the tracking scenarios.

The prediction algorithm was evaluated by executing it over a number of different scenarios and estimating the time taken for vehicles to reach the next intersection. These times were compared with the actual times taken to reach the intersection as reported by the network simulator. The simulation scenarios adopted a set of interconnected roadways with different lengths and speed limits. Additionally, mobility traces with different node densities were generated in order to analyze the effect of node density on the accuracy of the prediction algorithm.

Figures 3 and 4 present the results of the prediction algorithm for node densities of 10 and 100 nodes with the simulations running for 1,000 seconds. Clearly, there is a variation in performance. The main factors influencing the precision of the prediction algorithm are:

- **Node Density:** As seen in Figures 3 and 4, the accuracy of the algorithm is moderately affected by increased node density. This is expected in a real-world environment – the more vehicles present
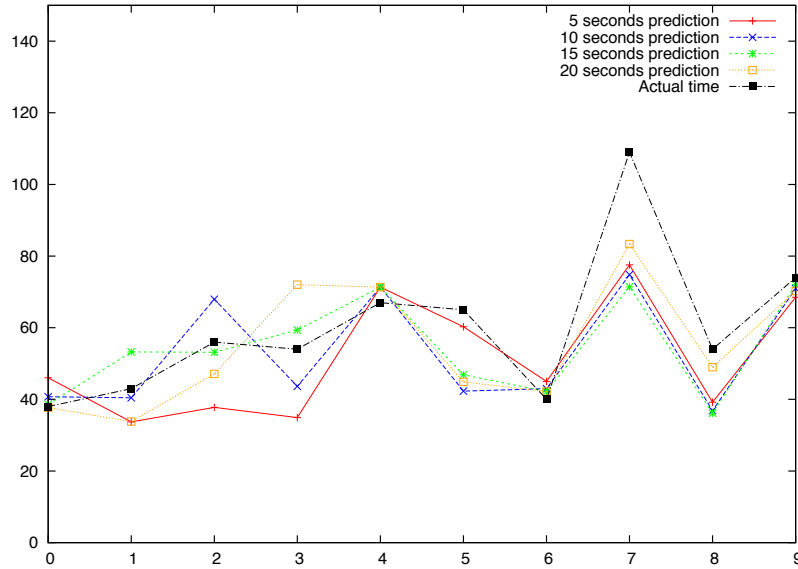
Figure 4.   Simulation results (100 node scenario).

on a roadway, the harder it is to accurately predict their movement due to their irregular acceleration/deceleration behaviors.

- **Roadway Length:** The length of the roadway affects the prediction algorithm because longer roadways allow for more fluctuations in acceleration, which can be difficult to model. Highways have very different characteristics than city roadways in terms of traffic flow.

- **Speed Limit:** Roadways with higher speed limits also allow for increased acceleration fluctuations, which affect the accuracy of the algorithm.

- **Duration of the Prediction:** Other than the macroscopic factors above, the duration that the prediction process is applied to the target is an important factor. The prediction algorithm measures the time taken by the target to pass a specific distance and then adds noise to the measurement to compensate for future acceleration/deceleration before calculating the predicted time. However, the accuracy of the algorithm is improved when the target is observed for a longer interval. This is illustrated in Figures 3 and 4, where a longer prediction interval of 20 seconds (i.e., the target's movement is observed for 20 seconds) yields a better prediction than a shorter interval. Another way to improve prediction accu-

racy is to take multiple observations at different times and compute the average.

- **Location of the Prediction:** Despite the fact that the prediction process can take place at any point on the roadway, the point where the prediction process takes place can be important, especially for highways that typically have higher speed limits than city roadways. For example, it is not advisable to execute the prediction process at the beginning of a highway. Since vehicles just start accelerating at this point, their accelerations are not representative of the rest of the journey.

Based on the above factors, it is clear that the tracking environment has an unavoidable influence on the tracking process. Carefully modeling the macroscopic features of the environment is, therefore, critical to improving tracking accuracy. Information about the general structure and layout of roadways can be obtained from online databases such as TIGER [11]. Other important road characteristics (e.g., density) can be modeled and estimated based on empirical observations.

9.   Conclusions

Vehicular tracking has important applications in crime prevention and criminal investigations. The algorithms presented in this paper implement the passive localization, tracking and prediction of the movement of a target vehicle. While there has been limited research related to passive vehicular tracking, we argue that the passivity requirement is very important for forensic and law enforcement purposes. Note, however, that the agent-based approach may face barriers in some jurisdictions where randomly recruiting tracking agents may be restricted by law.

Finally, we note that the RF measurements used in the algorithms include an unavoidable error margin. However, estimating the measurement parameters can provide an acceptable error margin. This is because in vehicular tracking it is only important to detect the presence of the target and partially track it to estimate its most likely trace in the corresponding roadway. Inferences can then be made if the target halted en route (e.g., to commit a crime) by observing the estimated time delay along each roadway.

References

[1] R. Baumann, S. Heimlicher and M. May, Towards realistic mobility models for vehicular ad hoc networks, *Proceedings of the Twenty-Sixth IEEE Conference on Computer Communications*, 2007.

[2] A. Benslimane, Localization in vehicular ad hoc networks, *Proceedings of the Systems Communications Conference*, pp. 19–25, 2005.

[3] A. Boukerche, H. Oliveira, E. Nakamura and A. Loureiro, Vehicular ad hoc networks: A new challenge for localization-based systems, *Computer Communications*, vol. 31(12), pp. 2838–2849, 2008.

[4] S. Brakatsoulas, D. Pfoser, R. Salas and C. Wenk, On map-matching vehicle tracking data, *Proceedings of the Thirty-First International Conference on Very Large Data Bases*, pp. 853–864, 2005.

[5] D. Engelhart, A. Sivasubramaniam, C. Barrett, M. Marathe, J. Smith and M. Morin, A spatial analysis of mobility models: Application to wireless ad hoc network simulation, *Proceedings of the Thirty-Seventh Annual Symposium on Simulation*, p. 34, 2004.

[6] H. Friis, A note on a simple transmission formula, *Proceedings of the IRE*, vol. 34(5), pp. 254–256, 2006.

[7] J. Harri, F. Filali, C. Binnet and M. Fiore, VanetMobiSim: Generating realistic mobility patterns for VANETs, *Proceedings of the Third International Workshop on Vehicular Ad Hoc Networks*, pp. 96–97, 2006.

[8] J. Luo and J. Hubaux, A Survey of Inter-Vehicle Communication, Technical Report IC/2004/24, School of Computer and Communication Sciences, Ecole Polytechnique Federale de Lausanne, Lausanne, Switzerland, 2004.

[9] J. Mansell and W. Riley, Vehicle Tracking and Security System, United States Patent 5,233,844, 1993.

[10] M. Treiber, A. Hennecke and D. Helbing, Congested traffic states in empirical observations and microscopic simulations, *Physical Review E*, vol. 62(2), pp. 1805–1824, 2000.

[11] U.S. Census Bureau, Topologically Integrated Geographic Encoding and Referencing, Washington, DC (www.census.gov/geo/www /tiger).