Cross-layer Location Information Security in Vehicular Networks☆

Gongjun Yan[a,1,*], Weiming Yang[b], jingli Lin[c], Danda B. Rawat[d]

[a] *Indiana University, Kokomo, IN 46902, USA*
[b] *Chongqing Technology & Business University, Chongqing 400067, China*
[c] *Xihua University, Chengdu 610041, China*
[d] *Old Dominion University, Norfolk, VA 23529, USA*

Abstract

Location, fundamental information, plays a critical role in many applications and network routings in Vehicular Adhoc NETworks (VANETs). Therefore, it is of importance to validate the location information. We propose a cross-layer design to achieve location validation. We assume vehicles are installed with radar, GPS and transceiver. On physical layer, radar detection can validate GPS coordinates. On network layer, an agreement of a location can be achieved. On application layer, location information as a set of measurements can be filtered and refined by using a proposed data fusion method. We also present the analytic results and evaluate the location validation methods.

Keywords: Data fusion, information integrity, vehicular adhoc network

1. Introduction

In the past few years, Vehicular Adhoc NETworks (VANETs), known as Vehicle-to-Vehicle and Vehicle-to-Roadside wireless communications, have received a huge amount of well-deserved attention in the literature. Indeed, because of their unmistakable societal impact that promises to revolutionize the way we drive, various car manufacturers, government agencies and

☆This work is an extension of a conference paper [1].

*Corresponding author

Email addresses: goyan@iuk.edu (Gongjun Yan), ywmls@hotmail.com (Weiming Yang), jinglilin77@gmail.com (jingli Lin), drawa001@odu.edu (Danda B. Rawat)

[1]This work is supported in part from IUK grant KO-2263160.

standardization bodies have spawned national and international consortia devoted exclusively to VANET. Examples include the Car-2-Car Communication Consortium [2], the Vehicle Safety Communications Consortium [3], and Honda's Advanced Safety Vehicle Program [4], among others.

The original impetus for the interest in VANET was provided by the need to inform fellow drivers of actual or imminent road conditions, delays, congestion, hazardous driving conditions and other similar concerns. Therefore, most concerns of applications on VANET focus on safety applications. Applications, such as traffic status reports, collision avoidance, emergency alerts, cooperative driving, are examples of safety applications. It is fare to say that most, if not all, applications are associated with location information.

Therefore it is of importance to ensure location security. This work was motivated by the need to provide confirmed topology information in VANETs and to build a network for applications, such as a congestion alert system or traffic view application. In this paper, we design cross-layer models which ensure the location security in physical layer, network layer, and application layer.

2. Related Work

Cryptography-based methods ensure the location information is unchanged by encryption. Many of the published work on position security focuses on using Public Key Infrastructure (PKI) [5, 6] and digital signatures [7, 8]. While these solutions provide security, they add significant overhead to the system. The algorithms involved in encrypting and decrypting the messages along with the issue of distributing public keys and their certificates make the system complex. In this paper, we take a different approach. We allow vehicles to send the information in plain-text and depend upon receivers to verify the information.

Leinmüller et al. [9] proposed a method to secure position information by using hard thresholds to detect false locations. Vehicles monitor data to verify the reported position. Although the authors do not use any other devices or hardware, the accuracy and efficiency are difficult to guarantee. In addition, this method is not flexible because of high mobility of vehicles in VANETs. Radio signal-based methods [10, 11] determine false claimed positions based on the received signal power. The basic idea of this method is that the distance between two nodes can be computed from the received signal power. If there is a vehicle at a position which does not match the distance computed from the received radio power, the position of the vehicle is determined to be a fake. However, a malicious node can use the same method to compute the transmission signal power to fool other nodes. Also, radio may bounce off vehicles and other obstacles. Detection based on these bounced radio signals may not be accurate.

Position estimation is often made by using reference devices. For example, cellular location estimation [12, 13] and local positioning system [14, 15] use infrastructure (base station) or reference device (RFID) to locate a position. Greater location estimation accuracy can be achieved as these devices are applied, even when the new devices have a short detection range (hundreds of meters). But the newly added devices increase the cost of the system. In addition, none of these researches address the location estimation in vehicular ad hoc network which is featured as high mobile speed and frequent topology change.

Kalman filter [16] and particle filter (Monte Carlo filter) [15] are applied in the position estimation as well. To apply these filters, the input data must follow the same distribution (normal distribution). But the input data from malicious attackers does not follow a constant distribution. The compromised input from malicious attackers will cause big error in these filters.

3. An Overview of Cross-layer Design

3.1. System Model

Today, new vehicles can have network, computing, and storage devices. Specifically, vehicles represented in this proposal are assumed to be endowed with the following features:

- A GPS receiver. The GPS receiver can provide the location information. The accuracy for civilian GPS is about 5m.

- A radar detector. The radar detector can detect the vehicle locations but the line of sight can be blocked. The radar range is 200m with precision about 3 meters.

- A wireless transceiver, using Dedicated Short Range Communications (DSRC, i.e. 802.11p) for fast communications. The transmission range is 300m.

- A computer center, which will provide data processing, computing and storage.

- A unique ID, like electronic vehicle plates which is a infrared device and is issued by a registration authority annually. This device can periodically broadcast its ID to neighboring vehicles.

We only address location validation in this paper. The validation of content of a message is out of scope.

3.2. Cross-layer Overview

We use radar, transceiver and GPS on physical layer to achieve location integrity which means location information is original (from generator, not replay messages), correct (not modified) and not fabricated. We present validation mechanisms to provide location integrity. On physical layer, we can validate location in the detection range of radar. The underlying idea is the famous proverb: "*Seeing is believing*". We use on-board radar as a virtual "eye" of a vehicle and a wireless transceiver as a virtual "ear" of the vehicle. By comparing what is seen to what has been heard, a vehicle can validate the heard location of a vehicle by seeing the vehicle at the announced location. Since applications on VANETs often involve location information of remote vehicles or entities which are ranging to miles, location information is often propagated in multi-hope mode. On the network layer, location can be validated and achieved an agreement. When a vehicle receives a message which includes location information of remote vehicles, the vehicle can randomly challenge and confirm the position by using vehicles on both co-directional and oncoming traffic. On application layer, a vehicle can query and obtain location information from other vehicles and can collect a set of location information from other vehicles as location reports. Therefore, we can apply a proposed algorithm to filter out the false location report and refine the remaining location report. In this paper, we will address a location validation mechanism that validates location information on three layers: physical layer, network layer and application layer.

4. Physical Layer: Radar Validation

The idea in this layer is to compare the location obtained from GPS with location obtained from radar detection [17]. If the radar detection matches the GPS location, the GPS location is validated. Otherwise, it is distrusted.

4.1. GPS Location

In GPS, when satellite radio signals are transmitted, they are distorted by the troposphere and the ionosphere. Therefore GPS coordinates have some tolerance. GPS data normally varies in the range of $\Delta x = \pm 5$ meter; $\Delta y = \pm 5$ meter. We assume that Δx and Δy are always equal, marked as $\Delta x = \Delta y = \Delta\alpha$. A set of possible GPS location (x, y) can be represented as

$$(x - x_{gps})^2 + (y - y_{gps})^2 \leq (\Delta\alpha)^2 \tag{1}$$

4.2. Radar-Detected Location

Since radar has tolerance, we assume that the radar's tolerance includes two parts: angle tolerance and radius tolerance $\triangle\gamma$, marked as $(\triangle\theta, \triangle\gamma)$. In Figure 1, the shaded region bounded by *HGQFEP* is the set of possible positions of the detected vehicle. We use (x, y) to represent the real position of vehicle in the GPS system and mark the radar readings as (θ, γ). We can use (2) and (3) to describe the two circles: *circle D* and *circle C* in Figure 1.

$$(x - \gamma \times \cos(\theta - \Delta\theta))^2 + (y - \gamma \times \sin(\theta - \Delta\theta))^2 \leq (\Delta\gamma)^2 \tag{2}$$

$$(x - \gamma \times \cos(\theta + \Delta\theta))^2 + (y - \gamma \times \sin(\theta + \Delta\theta))^2 \leq (\Delta\gamma)^2 \tag{3}$$

Here, θ is the detected angle, starting from north 0 degree and γ is the detected radius in meters (distance between vehicle A and vehicle B).

We note that there are two small regions *HRG* and *EBF* where the vehicle could be located, but these regions are not described by (2) or (3). Therefore we use the following formula to describe the region *FCGHDE* in Figure 1:

$$\begin{cases} \gamma - \Delta\gamma \leq \sqrt{x^2 + y^2} \leq \gamma + \Delta\gamma \\ \theta - \Delta\theta \leq \arctan\frac{x}{y} \leq \theta + \Delta\theta \end{cases} \tag{4}$$

Although (4) includes some regions which are described by (2) and (3), for example the region *RGCFB*, this has no negative effect because we will find an intersection between the GPS position formula and radar position formula by using the technique to be described in the next section.

4.3. Validating GPS Location by Radar Detection

If any of the following combinations has a solution, we can draw a conclusion that the detected vehicle is honest: (1) and (2); (1) and (3); (1) and (4) [17]. Otherwise, the vehicle is determined to be compromised.

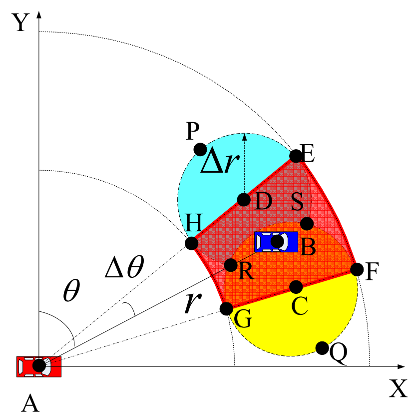

Figure 1: Message routing among the cells.

5. Network Layer: Location Agreement

Beyond a radar detection range, radar cannot validate location. Due to the DSRC transmission range limit, vehicles have to rely on each other to propagate location information. In such a network, it is hard to achieve an agreement of location information of a observed vehicle because some vehicles may change location information. We proposed a method to achieve location agreement by using the on-board radar in oncoming traffic. It is based on the fact that the vehicles in the same cell would see and hear almost the same traffic and road situation, so any modification done by malicious nodes can be detected by other honest vehicles. These honest vehicles then broadcast the correct record and isolate the malicious vehicle.

We use groups (or cells) as network propagation unit as well as location information validation unit on network layer shown in Figure 2. In a cell, secured location information needs to be propagated so that other vehicles approaching the cell can benefit from it. We have chosen a cell leader for each direction, which is responsible for forwarding this information along with the local traffic situation, to minimize collisions and bandwidth usage. Other vehicles in the cell have the responsibility of monitoring the responses of the cell leader. If either of them tries to change the records or inject wrong information, honest vehicles can notify other members about this compromised cell vehicle and initiate the process to determine new cell leader and broadcast the correct record. There could be a case where the presence of many compromised vehicles might isolate honest vehicles. We propose a challenging method on the basis of the following facts: (1) it is most likely that majority of the nodes are honest (one of our assumptions), (2) even if in one cell there are more compromised than honest nodes, it is difficult to maintain such a topology in a VANET, and (3) to confirm whether the cell router is compromised, vehicles in other cells who have received records from this cell router can run a simple verification test as described below.

The message challenging method is used to verify records which are disputed. Whenever vehicles in a cell do not agree with the broadcasted message, they express their concern by broadcasting their records. A vehicle getting different information for the same vehicle can verify the disputed record through this challenging method. Since we assume that traffic moves in both directions, a node can send the verification request to the cell router in the opposite direction. We show an example in Figure 2 whose traffic is moving in two directions. Suppose car I transmits a message which is propagated backwards and eventually received by car A. If car A would like to verify the position received from car I, it can send the verification request using cars moving in the opposite direction. The message is propagated on opposite direction traffic to car a through cars e and c. Since car I falls within the radar range of car a, its position can be verified using radar of car a. This verified information is then sent back to car A. This verification method works because of the low processing and propagation time. If the record that is being disputed is from a vehicle very far away, the vehicles drop the information instead of challenging it. Unless many records are being changed, or some other vehicle tells that vehicle in dispute is indeed close by, no request is sent.

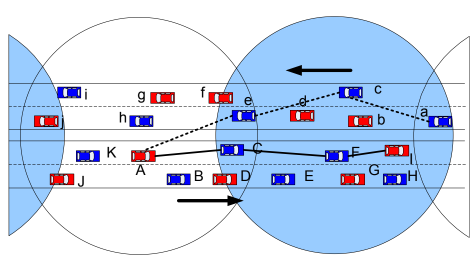

Figure 2: Message routing among the cells.

6. Application Layer: Information Fusion

6.1. Malicious Position and Error Position

On application layer, a vehicle Y can send a query message: can anybody tell me location of vehicle X? The neighboring vehicles will send location of X back to Y. The location information collected by Y includes two types of data. One is the data from honest vehicles. This type of data includes measurement errors which are assumed as normal distribution. Another type of data is the data from malicious attackers, called malicious data. Since attackers can modify the position into any arbitrary value, there is no universal distribution of the position from malicious attackers. Figure 3 shows the two types of data. In application layer, we will first filter out the malicious location and then refine location from the remaining locations.

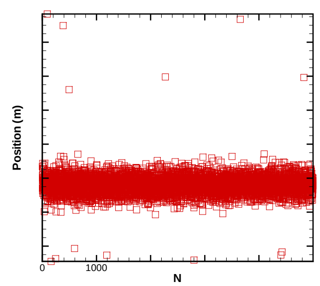

Figure 3: Malicious position with Gaussian error position. N is the number of positions. The outliers (malicious data) are far away from the center.

6.2. Filtering Malicious Data

The basic idea of filtering the malicious data from the collected data is by a method called box counting [18]. We assume the data is placed on a panel. The location data is the GPS coordinates in a two-dimensional mode. The panel is partitioned into grids. For each grid/multi-grid we count the number of positions. Since the majority vehicles are honest, we select the grid with highest number of vehicles. Inside this grid, we can partition it into smaller granularity grids and repeat the filtering process until the malicious positions are removed.

6.2.1. Gridding

The granularity of grid is a key issue. If the granularity is big, the malicious position will be included. If the granularity is small, the final position will deviate from the right position. The granularity of grid in this paper is $\sqrt{n}$ where n is the number of position data. The area of the panel is a square and length of the area edge equals to n. Therefore the grid is the unit of panel, i.e. $\sqrt{n}$. If the grid unit $\sqrt{n}$ is less than measurement error $2M_e$ where M_e is the mean of measurement error, i.e. $\pm M_e$, we partition the panel by $2M_e$. If $\sqrt{n} \gg 2M_e$, we will select the grid which includes highest number of vehicles. The selected grid will be used as new panel and we repeat the filtering and gridding process on the new panel until the grid unit is about $2M_e$.

6.2.2. Filtering

After gridding, the panel is partitioned into grids. We count the number of positions in each grid. The grid with the largest number of positions will be found. If there is more than one grid with the same largest number of positions, we will merge with other grids to find a bigger grid. If these grids are adjacent, these grids will be merged into a bigger grid. For example, shown in figure 6.2.2, grid(30,40) and grid(30,30) both have the same squares, 7 locations (shown as tiny squares). They are adjacent. We can totally create 4 bigger grids (2*2 grids) by merging 3 neighboring grids each time. The 2*2 grid with the largest number of square will be selected out of the 4 bigger grids, shown as the 2*2 grid(40,40) in figure 6.2.2. Once we selected the

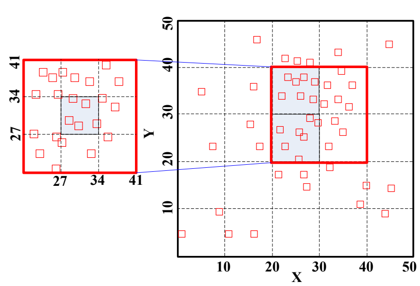

Figure 4: Refining grid and filtering [18].

grid with the largest number of squares, the rest of positions/squares are abandoned. If the grid is a merged grid, we will refine the grid and repeat the filtering process until the granularity of grid is about $2M_e$. Figure 6.2.2

shows how to refine the grid and reselect the grid with largest number of squares.

6.3. Estimating the Position

In the selected grid obtained and discussed in previous section, there are n input positions, marked as (x_1, y_1),(x_2, y_2),$\cdots$,(x_n, y_n). The task in this section is to search a point which has the shortest distance to all these known points in the grid selected previous section 6.2.2. This point is going to be our position estimation. The gradient method [18, 19] can find this position estimation. The basic idea of the gradient method is the following: randomly selecting a point in the grid, recursively walking through the area with a certain step length until finding the the position satisfying the acceptable precision. We define the objective function as following:

$$f(x, y) = \sum_{i=1}^{n} \sqrt{(x - x_i)^2 + (y - y_i)^2}$$
$$x \in (x_{min}, x_{max}), y \in (y_{min}, y_{max})$$

where $(x, y) \in \mathbf{R}$ are variables that represents optimal position estimation; x_{min} and x_{max} are the left-bound and right-bound points of the selected grid; similarly, y_{min} and y_{max} are the lower-bound and upper-bound of the selected bound. The purpose of objective function is to find a position (x,y) that has a shortest distance to all other position in a grid. To simplify discussion, we change notation of (x, y) as a vector $\mathbf{R}^{(k)} = \begin{pmatrix} x^{(k)} \\ y^{(k)} \end{pmatrix}$ where k is the steps. For initial step 0, i.e. $k = 0$, the initial search point is $\mathbf{R}^{(0)} = \begin{pmatrix} x^{(0)} \\ y^{(0)} \end{pmatrix}$ which is randomly selected (guessed) in the selected grid. The first order of gradient (differentiate) of function $f(\mathbf{R}^{(0)})$ writes

$$\nabla f(\mathbf{R}^{(0)}) = \begin{pmatrix} \sum_{i=1}^{n} \frac{x^0 - x_i}{\sqrt{(x^0-x_i)^2+(y^0-y_i)^2}} \\ \sum_{i=1}^{n} \frac{y^0 - y_i}{\sqrt{(x^0-x_i)^2+(y^0-y_i)^2}} \end{pmatrix}$$

where $\nabla f(.)$ represents the first order derivative of the objective function $f(.)$. If the mod of $\nabla f(\mathbf{R}^{(0)})$, i.e. $||\nabla f(\mathbf{R}^{(0)})||^2$ is no greater than the accepted position precision ϵ, the initial guessed point $\mathbf{R}^{(0)}$ satisfies the tolerance and is the optimal approximation of the vehicle position. Otherwise, the next point $\mathbf{R}^{(1)}$, marked as step 1 ($k = 1$), is calculated by the following recursion formula:

$$\mathbf{R}^{(k+1)} = \mathbf{R}^{(k)} - \lambda_k \nabla f(\mathbf{R}^{(k)}), k = 0, 1, 2, \cdots$$

where the searching length or the step length λ_k is determined as

$$\lambda_k = \frac{(\nabla f(\mathbf{R}^{(k)}))^T \nabla f(\mathbf{R}^{(k)})}{(\nabla f(\mathbf{R}^{(k)}))^T H(\mathbf{R}^{(k)}) \nabla f(\mathbf{R}^{(k)})}.$$

where $H(\mathbf{R}^{(k)})$ is the second order derivative of gradient of function $f(\mathbf{R})$. Based the recursion formula 5, a new guess point $\mathbf{R}^{(k+1)}$ will be obtained in next step until satisfying ϵ. Since $\epsilon > 0$ and $f(\mathbf{R})$ is a continues and infinite differentiable function, there always exists at least one point that $||\nabla f(\mathbf{R})||^2 \leq \epsilon$, which means that we can find a point that approximates the minimum point. If $f(\mathbf{R})$ has multiple peaks, we partition the selected grid into n sub-grid each of which contains a input position (point).

As a summary, the procedures of gradient method are the following:

1. Partition the selected grid into n sub-grid each of which contains one input position.
2. For each sub-grid, set the appropriate initial searching point, and set the tolerance $\epsilon > 0$. Compute $\nabla f(\mathbf{R}^{(k)})$. If $||\nabla f(\mathbf{R}^{(k)})||^2 < \epsilon$, then the $\mathbf{R}^{(k)}$ can be regarded as approximation of position in the sub-grid. The iteration stops in the sub-grid. Otherwise, the search process continues to the next step until ϵ is satisfied and the approximation position is found.

3. Compare the n approximation of points obtained from the previous step and find the position that cause the minimum value $f(\mathbf{R})$. This position is the position estimation in all input.

7. Cell Traffic Analysis

7.1. Big Cell Traffic Analysis

Consider a cell is so large that for all practical purposes it can be considered infinite to hold all vehicles passing the cell. The server of the queue is defined as passing the cell. If the server is busy, vehicles will wait until the server is empty. Denote by $X(t)$ the number of vehicles located in the cell at time t. Then, our physical intuition is not violated by assuming that

$\{X(t); t \geq 0\}$ is a birth and death process. Specifically, we are interested in the following problem. At time $t = 0$, the cell contains $X(0) \geq 0$ cars. After that, cars arrive at an arrival rate λ and depart at a departure rate μ. Of interest are:

- the probability that at a given time $t > 0$, there are exactly j cars in the cell;
- the expected number of the number cars in the cell at time $t > 0$;
- the limit as $t \to \infty$ of the expected number of cars in the cell (if such a limit exists and/or makes sense).

As we assumed, for all $t > 0, h > 0$ and $i = 0, 1, ...,$ the vehicles enters the cell at arrival rate λ.

$$P\{X(t+h) = i+1 | X(t) = i\} = \lambda h + o(h),$$
$$P\{X(t+h) = i-1 | X(t) = i\} = \mu_i h + o(h)$$

because the probability that a car arrival will occur in $(t, t+h)$ is independent of the number of vehicles in the cell at time t. It is obvious for the second equation as well.

Let us calculate $p_i(t)$ under the assumption that

$$\mu_i = i\mu$$

In this case, the system of difference-differential equations becomes

$$p_0^{'}(t) = -\lambda p_0(t) + \mu p_1(t),$$
$$p_j^{'}(t) = \lambda p_{j-1}(t) - (\lambda + j\mu)p_j(t) + \mu(j+1)p_{j+1}(t)$$

To solve this, we can use the generating functions. Set

$$g(t, u) = \sum_{j=0}^{\infty} p_j(t)\mu^j$$

then taking into account the previous two differential equations, we obtain,

$$
\begin{aligned}
\frac{dg(t,u)}{dt} &= \sum_{j=0}^{\infty} p_j^{'}(t)\mu^j \\
&= -\lambda p_0(t)(1-\mu) + \mu p_1(t)(1-\mu) - \lambda p_1(t)(1-\mu)\mu \\
&+ 2\mu p_2(t)\mu(1-\mu) - ... \\
&= -\lambda(1-\mu)\sum_{j=0}^{\infty} p_j(t)\mu^j + \mu(1-\mu)\sum_{j=1}^{\infty} jp_j(t)\mu^{j-1} \\
&= -\lambda(1-\mu)g(t,\mu) + \mu(1-\mu)\frac{dg(t,\mu)}{d\mu}
\end{aligned}
$$

Thus, the generating function $g(t,\mu)$ satisfies the linear partial differential equation

$$\frac{dg(t,\mu)}{d\mu} - \mu(1-\mu)\frac{dg(t,\mu)}{d\mu} = -\lambda(1-\mu)g(t,\mu)$$

Suppose that $X(0) = i_0$, then,

$$g(0,\mu) = u^{i_0} \tag{5}$$

Solving this equation by standard methods and using the initial condition (5), we obtain

$$g(t,\mu) = [1-(1-\mu)e^{-\mu t}]^{i_0} \exp\left[-\frac{\lambda}{\mu}(1-\mu)(1-e^{-\mu t})\right] \tag{6}$$

Lemma 1. *We now expand (6) to obtain $p_j(t)$ which represents the probability that the number of cell cars is j at time t.*

$$
\begin{aligned}
p_j(t) = \exp\left\{-\frac{\lambda}{\mu}(1-e^{-\mu t})\right\} &\sum_{k=0}^{min\{i_0,j\}} \tbinom{i_0}{k} \left(\frac{\lambda}{\mu}^{j-k}\right) \\
\times \frac{e^{-\mu t k}(1-e^{-\mu t})^{i_0+j-2k}}{(j-k)!}
\end{aligned} \tag{7}
$$

where $j = 0, 1, ...$

Proof. For the first part of (6), we can write

$$\begin{aligned} first &= [1-(1-\mu)e^{-\mu t}]^{i_0} \\ &= [1-e^{-\mu t}-\mu e^{-\mu t}]^{i_0} \\ &= \sum_{k=0}^{i_0} \tbinom{k}{i_0}(1-e^{-\mu t})^{i_0-k}(\mu e^{-\mu t})^k \\ &= \sum_{k=0}^{i_0} \tbinom{k}{i_0}(1-e^{-\mu t})^{i_0-k}(e^{-\mu t})^k \mu^k \end{aligned}$$

For the second part of (6), we can write

$$\begin{aligned} second &= \exp\left[-\frac{\lambda}{\mu}(1-\mu)(1-e^{-\mu t})\right] \\ &= \exp\left[-\frac{\lambda}{\mu}(1-e^{-\mu t})+\frac{\lambda}{\mu}\mu(1-e^{-\mu t})\right] \\ &= \exp\left\{-\frac{\lambda}{\mu}(1-e^{-\mu t})\right\}\exp\left\{\frac{\lambda}{\mu}\mu(1-e^{-\mu t})\right\} \\ &= \exp\left\{-\frac{\lambda}{\mu}(1-e^{-\mu t})\right\}\sum_{j=0}^{\infty}\frac{\frac{\lambda}{\mu}^j(1-e^{-\mu t})^j}{j!}\mu^j \\ &= \sum_{j=0}^{\infty}\exp\left\{-\frac{\lambda}{\mu}(1-e^{-\mu t})\right\}\frac{\frac{\lambda}{\mu}^j(1-e^{-\mu t})^j}{j!}\mu^j \end{aligned}$$

Since $g(t, \mu) = first \cdot second$, we can obtain

$$
\begin{aligned}
g(t,\mu) = & \left[\sum_{j=0}^{\infty} \exp\left\{ -\frac{\lambda}{\mu}(1-e^{-\mu t}) \right\} \frac{\frac{\lambda}{\mu}^{j}(1-e^{-\mu t})^{j}}{j!} \mu^{j} \right] \\
& \cdot \left[\sum_{k=0}^{i_0} \binom{k}{i_0} (1-e^{-\mu t})^{i_0-k} (e^{-\mu t})^{k} \mu^{k} \right] \\
= & \left[\sum_{j=k}^{\infty} \exp\left\{ -\frac{\lambda}{\mu}(1-e^{-\mu t}) \right\} \frac{\frac{\lambda}{\mu}^{j-k}(1-e^{-\mu t})^{j-k}}{j-k!} \mu^{j-k} \right] \\
& \cdot \left[\sum_{k=0}^{i_0} \binom{k}{i_0} (1-e^{-\mu t})^{i_0-k} (e^{-\mu t})^{k} \mu^{k} \right] \quad (change\ varibles) \\
= & \sum_{j=k}^{\infty} \exp\left\{ -\frac{\lambda}{\mu}(1-e^{-\mu t}) \right\} \sum_{k=0}^{min\{i_0,j\}} \binom{i_0}{k} \left(\frac{\lambda}{\mu}^{j-k} \right) \\
& \times \frac{e^{-\mu t k}(1-e^{-\mu t})^{i_0+j-2k}}{(j-k)!} \mu^{j} \\
= & \sum_{j=k}^{\infty} p_j(t) \mu^{j}
\end{aligned}
$$

Therefore, we can obtain $p_j(t)$, as claimed.

$$
\begin{aligned}
p_j(t) = & \exp\left\{ -\frac{\lambda}{\mu}(1-e^{-\mu t}) \right\} \sum_{k=0}^{min\{i_0,j\}} \binom{i_0}{k} \left(\frac{\lambda}{\mu}^{j-k} \right) \\
& \times \frac{e^{-\mu t k}(1-e^{-\mu t})^{i_0+j-2k}}{(j-k)!}
\end{aligned}
$$

Note that the first term on the right side of (6) is the probability generating function of the Binomial distribution with $p = \exp\{-\mu t\}$, whereas the second term is the probability generating function of the Poisson distribution with mean value

$$
\Lambda(t) = \frac{\lambda(1-e^{-\mu t})}{\mu}
$$

Therefore,

$$X(t) = X_0(t) + X_1(t) \tag{8}$$

where $X_0(t)$ is Binomial component, $X_1(t)$ is a Poisson component, and $X_0(t)$ is independent of $X_1(t)$ for all $t \geq 0$.

Lemma 2.

$$E\{X(t)\} = i_0 e^{-\mu t} + \frac{\lambda}{\mu}(1 - e^{-\mu t})$$

PROOF. From (8), we know that $X(t)$ is determined by $X_0(t)$ and $X_1(t)$. From (6) and Lemma (1), we know $X_0(t)$ is Binomial distribution:

$$X_0(t) \sim Binomial(i_0, e^{-\mu t})$$

. Therefore, the expectation of $X_0(t)$ is

$$E\{X_0(t)\} = i_0 e^{-\mu t}$$

Similarly, from Lemma (1), we know $X_1(t)$ is Poisson distribution:

$$X_1(t) \sim Poisson(\frac{\lambda}{\mu}(1 - e^{-\mu t}))$$

. Therefore, the expectation of $X_1(t)$ is

$$E\{X_1(t)\} = \frac{\lambda}{\mu}(1 - e^{-\mu t})$$

. From (8), we know

$$\begin{aligned} E\{X(t)\} &= E\{X_0(t)\} + E\{X_1(t)\} \\ &= i_0 e^{-\mu t} + \frac{\lambda}{\mu}(1 - e^{-\mu t}). \end{aligned}$$

Lemma 3.

$$\lim_{t \to \infty} E\{X(t)\} = \frac{\lambda}{\mu}$$

Based on Lemma 2, it is easy to obtain Lemma 3 and we skip the proof.

7.2. Small Cell Traffic Analysis

Consider a small cell where the number of vehicle capacity is finite and the total capacity of the cell is N. Denote by $X(t)$ the number of the cell in use at time t. Then, our physical intuition is not violated by assuming that $\{X(t); t > 0\}$ is a birth-death process. It also seems reasonable to assume that, the cars arrive at a rate of $\lambda(t)$, independent of the number of cars already in the cell, and if the cell contains k cars, then the departure rate is $k\mu(t)$, where $\mu(t)$ is a function of t.

The state space of this process is $S = \{0, 1, \ldots, N\}$. For every positive integer k, $1 \leq k \leq N$, the event $X(t) = k$ occurs if the cell contains k cars at time t. We let $P_k(t)$ denote the probability that the event $X(t) = k$ occurs, that is

$$P_k(t) = P\{X(t) = k\}$$

. To make the mathematical derivations more manageable, at this point we assume that $\lambda(t) = \lambda$ and $\mu(t) = \mu$. Thus, the transition rate matrix of this birth-death process is

$$Q = \begin{pmatrix} -\lambda & \lambda & 0 & \cdots & 0 & 0 \\ \mu & -(\lambda+\mu) & \lambda & \cdots & 0 & 0 \\ 0 & 2\mu & -(\lambda+2\mu) & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & -[\lambda+(N-1)\mu] & (N-1)\mu \\ 0 & 0 & 0 & \cdots & N\mu & -N\mu \end{pmatrix} \tag{9}$$

And the Fokker-Planck equation is as follows:

$$\begin{cases} \dfrac{dP_0(t)}{dt} = -\lambda P_0(t) + \mu P_1(t) \\[2ex] \dfrac{dP_j(t)}{dt} = \lambda P_{j-1}(t) - (\lambda + j\mu)P_j(t) + (j+1)\mu P_{j+1}(t) \\[2ex] \dfrac{dP_N(t)}{dt} = \lambda P_{N-1}(t) + N\mu P_N(t) \end{cases}$$

If the limit distribution $P = (P_0, P_1, \cdots, P_N)'$ exist, that is $P_j = \lim\limits_{t\to\infty} P_{ij}(t)$ and it is independent of i, then we have $dP_j(t)/dt = 0$ as $t \to \infty$. Hence, in the above equation let $t \to \infty$ we have

$$\begin{cases} -\lambda P_0 + \mu P_1 = 0 \\ \\ \lambda P_{j-1} - (\lambda + j\mu)P_j + (j+1)\mu P_{j+1} = 0 \\ \\ \lambda P_{N-1} - N\mu P_N = 0 \end{cases} \tag{10}$$

Let $g_j = \lambda P_{j-1} + j\mu P_j$, we have

$$\begin{aligned} & \lambda P_{j-1} - (\lambda + j\mu)P_j + (j+1)\mu P_{j+1} \\ = \; & [\lambda P_{j-1} - j\mu P_j] - [\lambda P_j - (j+1)\mu P_{j+1}] \\ = \; & g_j - g_{j+1} = 0 \end{aligned}$$

Hence $g_0 = 0$, $g_N = 0$ and $g_j = g_{j+1}$, for $1 \le j \le N-1$.

Now the equation system (11) can be written as

$$\begin{cases} P_1 = \dfrac{\lambda}{\mu} P_0 \\ \\ P_j = \dfrac{\lambda}{j\mu} P_{j-1} = \dfrac{1}{j!}\left(\dfrac{\lambda}{\mu}\right)^j P_0 \\ \\ P_N = \dfrac{1}{N!}\left(\dfrac{\lambda}{\mu}\right)^N P_0 \end{cases}$$

It is known that $\sum_{j=0}^{N} P_j = 1$, that is

$$\left[\frac{\lambda}{\mu} + \frac{1}{2!}\left(\frac{\lambda}{\mu}\right)^2 + \cdots + \frac{1}{N!}\left(\frac{\lambda}{\mu}\right)^N\right] P_0 = 1$$

Hence, solve the above equations, we have

$$P_j = \frac{\frac{1}{j!}\left(\frac{\lambda}{\mu}\right)^j}{\sum_{i=0}^{N}\frac{1}{i!}\left(\frac{\lambda}{\mu}\right)^i}$$

8. Conclusions and Future work

We described a cross-layer design for validating location of vehicles. On physical layer, we validate the vehicle locations by enlisted devices like radar on network layer, an agreement of location can be achieved by exchanging and challenging location information among vehicles. On application layer, we filter out the malicious location data (outliers) from a noise environment and compute the high precision location from low resolution location.

In the future, we will explore our algorithm by using the history of position estimates. New filters will be used, for example, Kalman filtering or particle filtering after filtering the malicious data. In addition, the position estimator can be incorporated into position security to ensure the position integrity.

- An active location integrity model which validates the vehicle locations by enlisted devices like radar.
- A passive location integrity model which builds a mobility history of neighboring vehicles and filters out the impossible locations.
- A general location integrity model which filters out the malicious location data (outliers) from a noise environment and compute the high precision location from low resolution location.

References

[1] G. Yan, S. Olariu, M. C. Weigle, Cross-layer location verification enhancement in vehicular networks, in: Proceedings of the IEEE Intelligent Vehicles Symposium, San Diego, USA, 2010, pp. 95–100.

[2] Car 2 Car Communication Consortium, `http://www.car-to-car.org/`.

[3] US Department of Transportation, National Highway Traffic Safety Administration, Vehicle safety communications consortium, `http://www-nrd.nhtsa.dot.gov/pdf/nrd-12/CAMP3/pages/VSCC.htm`.

[4] A. Takahashi, N. Asanuma, Introduction of Honda ASV-2 (Advanced Safety Vehicle-Phase 2), in: Proceedings of the IEEE Intelligent Vehicles Symposium, Detroit, USA, 2000, pp. 694–701.

[5] J.-P. Hubaux, S. Capkun, J. Luo, The security and privacy of smart vehicles, IEEE Security and Privacy Magazine 2 (3) (2004) 49–55.

[6] M. Raya, A. Aziz, J.-P. Hubaux, Efficient Secure Aggregation in VANETs, in: Proceedings of the ACM Workshop on Vehicular Ad Hoc Networks (VANET), Los Angeles, CA, 2006, pp. 67–75.

[7] F. Armknecht, A. Festag, D. Westhoff, K. Zeng, Cross-layer privacy enhancement and non-repudiation in vehicular communication, in: Proceedings of the Workshop on Mobile Ad-Hoc Networks (WMAN), Bern, Switzerland, 2007.

[8] J. Y. Choi, P. Golle, M. Jakobsson, Tamper-evident digital signatures: Protecting certification authorities against malware, in: Proceedings of the IEEE International Symposium on Dependable, Autonomic and Secure Computing (DASC), 2006, pp. 37–44.

[9] T. Leinmüller, E. Schoch, F. Kargl, C. Maihöfer, Improved security in geographic ad hoc routing through autonomous position verification, in: Proceedings of the ACM Workshop on Vehicular Ad Hoc Networks (VANET), Los Angeles, CA, 2006, pp. 57–66.

[10] T. Suen, A. Yasinsac, Ad hoc network security: Peer identification and au- thentication using signal properties, in: Systems, Man and Cybernetics (SMC) Information Assurance Workshop, 2005, pp. 432–433.

[11] B. Xiao, B. Yu, C. Gao, Detection and localization of Sybil nodes in VANETs, in: Proceedings of the Workshop on Dependability Issues in Wireless Ad Hoc Networks and Sensor Networks, Los Angeles, CA, USA, 2006, pp. 1–8.

[12] I. Papageorgiou, C. Charalambous, C. Panaviotou, An enhanced received signal level cellular location determination method via maximum

likelihood and kalman filtering, in: Wireless Communications and Networking Conference, 2005 IEEE, Vol. 4, 2005, pp. 2524–2529 Vol. 4.

[13] N. Khajehnouri, A. Sayed, A non-line-of-sight equalization scheme for wireless cellular location, in: Acoustics, Speech, and Signal Processing, 2003. Proceedings. (ICASSP '03). 2003 IEEE International Conference on, Vol. 6, 2003, pp. VI–549–52 vol.6.

[14] N. Patwari, I. Hero, A.O., M. Perkins, N. Correal, R. O'Dea, Relative location estimation in wireless sensor networks, Signal Processing, IEEE Transactions on 51 (8) (2003) 2137–2148.

[15] Widyawan, M. Klepal, D. Pesch, Influence of predicted and measured fingerprint on the accuracy of rssi-based indoor location systems, in: Positioning, Navigation and Communication, 2007. WPNC '07. 4th Workshop on, 2007, pp. 145–151.

[16] S. il Ko, J. suk Choi, B. hoon Kim, Performance enhancement of indoor mobile localization system using unscented kalman filter, in: SICE-ICASE, 2006. International Joint Conference, 2006, pp. 1355–1360.

[17] G. Yan, S. Olariu, M. C. Weigle, Providing VANET security through active position detection, Computer Communications: Special Issue on Mobility Protocols for ITS/VANET 31 (12) (2008) 2883C2897.

[18] G. Yan, X. Chen, S. Olariu, Providing vanet position integrity through filtering, in: Proceedings of the 12th International IEEE Conference on Intelligent Transportation Systems (ITSC2009), St. Louis, MO, USA, 2009, pp. 569–574.

[19] S. I. Gass, Recent developments in linear programming, in: F. Alt (Ed.), Advances in Computers 1, Academic Press, 1960, p. 314.