

Research Article

An Efficient Methodology for Calibrating Traffic Flow Models Based on Bisection Analysis

Enzo C. Jia,¹ Jianqiang Wang,² and Daiheng Ni¹

¹ Department of Civil Engineering, University of Massachusetts, Amherst, MA 01003, USA

² State Key Laboratory of Automotive Safety and Energy, Tsinghua University, Beijing 100084, China

Correspondence should be addressed to Jianqiang Wang; wjqlws@tsinghua.edu.cn

Received 19 July 2013; Accepted 22 December 2013; Published 18 February 2014

Academic Editor: Bo Yu

Copyright © 2014 Enzo C. Jia et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

As urban planning becomes more sophisticated, the accurate detection and counting of pedestrians and cyclists become more important. Accurate counts can be used to determine the need for additional pedestrian walkways and intersection reorganization, among other planning initiatives. In this project, a camera-based approach is implemented to create a real-time pedestrian and cyclist counting system which is regularly accurate to 85% and often achieves higher accuracy. The approach retasks a state-of-the-art traffic camera, the Autoscope Solo Terra, for pedestrian and bicyclist counting. Object detection regions are sized to identify multiple pedestrians moving in either direction on an urban sidewalk and bicyclists in an adjacent bicycle lane. Collected results are processed in real time, eliminating the need for video storage and postprocessing. In this paper, results are presented for a pedestrian walkway for pedestrian flow up to 108 persons/min and the limitations of the implemented system are enumerated. Both pedestrian and cyclist counting accuracy of over 90% is achieved.

1. Introduction

1.1. Introduction to a Severe Highway Safety Problem. Every year around 1.2 million people are killed in road accidents all over the world, and another 50 million get injured [1]. Among these tragedies, many are caused by aggressive driving in severe conditions that require extra carefulness, like bad weather, road sections near work zone or crash site, and so forth. Usually people have their consistent driving styles, which can be described by a traffic flow model, and under a special situation it is not guaranteed a driver adjusts his/her style properly in correspondence to the change of situations which happened on road. For example, a proportion of drivers failed to drive more carefully near work zones, and this is a remarkable source of accidents [2]. And during conditions of light precipitation, friction between tyres and pavement deteriorates significantly, but drivers not experienced enough might be not sure how careful they should be. Under such a circumstance an unexperienced driver has a high probability to drive in an overly aggressive style. As a result, an investigation conducted by FHWA [3] suggests that an average annual rate for weather-related crashes from 1995

to 2008 is approximately 1.5 million; this number accounts for 24% of all vehicle crashes annually. And during the same period more than 7 thousand people were killed on road every year, which accounts for 17% of all crash fatalities. These statistics can be found in Table 1.

1.2. Existing Solutions. Currently there are several methods deployed by the highway administration to improve traffic safety under severe weather conditions:

- (1) reshaping dangerous sections of road network to improve on-road safety condition. By analysing crash history, administration departments can identify road sections with relatively higher accident/fatality rates, and it is helpful to redesign and reconstruct such sections;
- (2) deploying more roadside information board, including LED display board, so the highway administration can edit information displayed;
- (3) delivering postcrash care including temporary information board and guidance by police cars.

TABLE 1: Weather-related crash statistics (annual averages 1995–2008) [3].

Road weather conditions	Annual rates (approximately)	Weather-related crash statistics	
		Percentages	
Wet pavement	1,128,000 crashes	18% of vehicle crashes	75% of weather-related crashes
	507,900 persons injured	17% of crash injuries	81% of weather-related crash injuries
	5,500 persons killed	13% of crash fatalities	77% of weather-related crash fatalities
Rain	707,000 crashes	11% of vehicle crashes	47% of weather-related crashes
	330,200 persons injured	11% of crash injuries	52% of weather-related crash injuries
	3,300 persons killed	8% of crash fatalities	46% of weather-related crash fatalities
Snow/sleet	225,000 crashes	4% of vehicle crashes	15% of weather-related crashes
	70,900 persons injured	2% of crash injuries	11% of weather-related crash injuries
	870 persons killed	2% of crash fatalities	12% of weather-related crash fatalities
Icy pavement	190,100 crashes	3% of vehicle crashes	13% of weather-related crashes
	62,700 persons injured	2% of crash injuries	10% of weather-related crash injuries
	680 persons killed	2% of crash fatalities	10% of weather-related crash fatalities
Snow/slushy pavement	168,300 crashes	3% of vehicle crashes	11% of weather-related crashes
	47,700 persons injured	2% of crash injuries	8% of weather-related crash injuries
	620 persons killed	1% of crash fatalities	9% of weather-related crash fatalities
Fog	38,000 crashes	1% of vehicle crashes	3% of weather-related crashes
	15,600 persons injured	1% of crash injuries	2% of weather-related crash injuries
	600 persons killed	1% of crash fatalities	8% of weather-related crash fatalities
Weather-related	1,511,200 crashes	24% of vehicle crashes	
	629,300 persons injured	21% of crash injuries	
	7,130 persons killed	17% of crash fatalities	

Obviously methods mentioned above are helping in improving road safety significantly; however, as we are discussing safety issues under severe conditions, there is more we can do with specially designed technologies for further help. Among methods mentioned above, solution 1 requires years of crash record, which might not be available for new road sections. And usually highway segments are designed and constructed by considering specific shapes of landscape, so it is not easy to make significant modifications. What is more, highway construction requires huge budget and amount of time, and this makes it even more difficult to reshape a segment. Regarding solutions 2 and 3, they work well in hazardous situations, but the problem we are facing is, these applications distract travellers' attention from normal driving, and this can be a remarkable cause of accidents like tailgating. When drivers are careful enough for on-going incidents on road, say, drizzle or slight accidents, distracting methods like LED displays and large-scale police car manipulation would contribute to troubling consequences. However, when severe situations like snowstorm or heavy accidents happen, it is much more possible that drivers' average aggression is too high to keep traffic safe. So here we need a supporting system to point out whether drivers are driving too aggressively to keep them safe on road or a system tells degree of drivers' aggression at the moment of observation. By deploying such a system it is possible to apply administrating strategies like solutions 2 and 3 accordingly to help modify drivers' aggression.

2. An Efficient Methodology

2.1. Drivers' Aggression. Thus, we raised the idea of developing an aggression surveillance system; intuitively the following question is how to rate a driver's aggression. The most efficient way would be finding a parameter or constant which denotes drivers' aggression in a traffic flow model, and by calibrating this model we get a quantified value of average aggression. By comparing this value with the normal aggression value under similar situations, it is helpful in making decisions of whether to intervene with on-road traffic.

2.2. Introduction to Traffic Flow Models and Calibration Methods

2.2.1. Previous Works in Traffic Flow Modelling and Calibrating. Pursuing of a precise mathematical model to describe traffic flow behaviours has lasted for more than half a century, and there are models successfully developed and applied. Greenshields' model [4] applies a simple linear relationship to traffic density and average speed, which is not precise enough but worked as the very beginning of research on modelling traffic flow characters. Newell's model [5, 6] has a much more precise description of traffic behaviours. General Motors model [7, 8] was developed as a car-following model but can be used as a tool for research on macroscopic traffic flow theories. Similarly Gipps' model [9] is a successful

development on drawing how traffic's flow, density, and average speed depend on each other. In Kosonen's Ph.D. dissertation [10], a rule-based model was applied in simulating traffic behaviours in urban environment, and this was based on previous research on utilizing fuzzy decision rules to simulate traffic flow [11]. As the latest achievement on modelling traffic flows, Daiheng Ni et al. [12–14] utilized field theory to simulate behaviours of vehicles on road and got the longitudinal control model (LCM), which we will introduce in detail.

On the calibration part, there are also previous works trying to locate sufficient and precise methods for finding values of parameters in traffic flow models. This field of research was initiated by Chandler et al. [7], Gazis et al. [8], and Herman et al. [15] in the late 1950s and early 1960s. GM model was opted for by Chandler's team and field data was studied for obtaining parameter values in the model. With development of computational technology, simulation tools are utilized in studying traffic phenomena and calibrating models, as revealed by Brackstone and McDonald [16]. In the past decade there are several research projects improving calibration methods. Using field data obtained on IH-10 in Houston, Texas, Schultz and Rilett [17] evaluated the parameter values they calculated with microscopic traffic simulation tools. In 2008, Kesting and Treiber [18] tested a method that minimized the difference between field data and model prediction. They resulted with errors lower than 30%. Also some ideas from computer science helped getting a higher accuracy. S. P. Hoogendoorn and R. Hoogendoorn [19] implemented genetic algorithm for calculating parameter values in traffic flow models in 2010.

2.2.2. Traffic Flow Models for Evaluating Drivers' Aggression.

A suitable model for finding average aggression is longitudinal control model (LCM) [12–14]. In this macroscopic model we define relationship between traffic density and speed as follows:

$$k = \frac{1}{(\gamma v^2 + \tau v + l) \left[1 - \ln \left(1 - \left(v/v_f \right) \right) \right]}, \quad (1)$$

where k is traffic density, v is space-mean speed, γ denotes degree of aggression that characterizes the driving population, τ is average response time that characterizes the driving population, l is average effective vehicle length, and v_f is free flow speed. So if we get the value of γ against field data under a specific situation, current degree of aggression on road is found.

Also we use Newell's model as an enhancement for modelling drivers' behaviours in this research.

So if we set a live camera at roadside to collect real-time traffic data and use these data to calibrate LCM model and Newell's model, it is possible to get current value of average aggression or other related parameters. However, calibration tasks lead to optimization problems, which usually costs huge amounts of time finding solutions. In order to clear this challenge, an efficient algorithm developed for calibrating traffic flow models against empirical data is demonstrated in the following sections.

2.3. *Essential Points of This Methodology.* To achieve an approach for such a system, three requirements are pointed out:

- (1) this approach should be able to precisely reflect traffic conditions at the specific moment on a chosen segment of highway;
- (2) computation should not be a significant burden so the system can react to changing situations immediately;
- (3) algorithm adopted by this system must have a rigorous mathematical derivation of evaluating criterion.

Item 1 is intuitive since precision is fundamental for a research on calibration. In item 2, by emphasizing efficiency of the algorithm this system adopts, it is necessary to get it to react quickly, and this fact further strengthens to the system's ability to reflect real situation on assigned location and time. Also we design the third item because such a system is supposed to work in various environments, regardless of different traffic flow, density, speed, or such rates affected by abnormal situations.

To achieve these objectives, the following ideas are raised:

- (1) aggregating raw observation data to reduce the algorithm's running time and applying coefficients on aggregated observation points to offset loss on accuracy because of data aggregation;
- (2) using bisection method to reach a solution of optimized parameters rapidly.

3. Principle of Methodology

3.1. *Background.* Generally speaking, calibration problems lead to optimization problems. However, under most circumstances people solve optimization problems by applying numeration methods, considering that it is not easy to find a universal method for a kind of problems. So even successful commercial optimization packages, like IBM ILOG CPLEX Optimizer, work in an extremely time-consuming way. Meanwhile, most root-finding problems have corresponding well-defined solving methods, either analytical or numerical or both. And among these methods a lot are time efficient, such as bisection method and Newton's method for solving linear equations and Crank-Nicolson method for partial differential equations.

Based on facts mentioned above, we determine principle of this research as converting an optimization problem into a root-finding problem. However, the challenge here is that an optimization problem requires a solution which achieves the minimized value of its objective function, whilst what a root-finding problem needs is a solution to get the value of a function equal to zero. So regarding this particular calibration problem, we establish a function (12). And when value of this function goes towards zero, a traffic flow model gets better accuracy. In this way we convert an optimization problem into a root-finding problem, and then it is possible to apply numerical methods for solutions. Method of establishing this variable is introduced in Section 4.2.

3.2. *An Introduction to Bisection Method.* In engineering applications, there are usually complex computing tasks that require high amount of resources on both time and hardware. This leads to an obvious conflict between efficiency and cost. As a compromised solution, scientists in applied mathematics developed a series of algorithms to approximate solutions of specified computing problems. In this way an algorithm, which we call a numerical method, is designed to deliver an approximated solution to a computing problem with a high enough and thus adjustable accuracy.

Among the most well-known numerical algorithms, bisection method, also known as binary search method, is widely used because of its simplicity and robustness. As most other numerical algorithms, this method is applied to solve root-finding problems $f(x) = 0$, where the continuous function f is defined on interval $[a, b]$ as sign of $f(a)$ is opposite to $f(b)$'s. Thus, such a problem can be described as

$$\begin{aligned} f(x) = 0, \quad x \in [a, b] \quad a < b \\ f(a) \times f(b) \leq 0. \end{aligned} \tag{2}$$

A loop is designed to solve the problem described in (2) where, in every iteration of the loop, we divide the interval $[a, b]$ into two halves $[a, (a + b)/2]$ and $[(a + b)/2, b]$. By computing value of function f at midpoint $(a + b)/2$, we can determine which half of the interval a root of $f(x) = 0$ lies in. If $f((a + b)/2)$ has the same sign as $f(a)$, then it's certain that there is a root of $f(x) = 0$ in interval $[(a + b)/2, b]$ or if $f((a + b)/2)$ has the same sign as $f(b)$, then this root can be found in $[a, (a + b)/2]$. Also it is theoretically possible, but very unlikely, that $f((a + b)/2) = 0$ which leads to an exact solution $(a + b)/2$. As this loop going interval $[a, b]$ keeps shrinking and this method makes sure that there is a root which lies in this interval. So when length of $[a, b]$ is smaller than expected error ϵ , either a or b can be final result of this looping as approximated solution to the problem expressed in (2). This means we can take $c = (a + b)/2$ as final result when length of $[a, b]$ is smaller than 2ϵ , since in this way one less loop is computed to get a better performance on running time. So determination of the result from this method can be described in (3), as s denotes approximated solution for above root-finding problem and ϵ is the maximum error allowed:

$$s = \frac{a + b}{2} \quad \text{when } |a - b| < \epsilon. \tag{3}$$

Though there are many methods providing faster convergence, we still adopt bisection method instead of other options. And the reason is, our problem is finding root for a function defined in (12), and this function is not a definite mathematical function, so simplicity of a root-finding method is essential to guarantee robustness under any possible situation of input data provided. For example, Newton's method is more advanced compared to bisection method but requires derivation operations, and in this case derivation leads to unpredictable results. After testing repeatedly, we see that bisection method delivers both time efficiency and robustness we need.

And Figure 1 shows principle of how this method works.

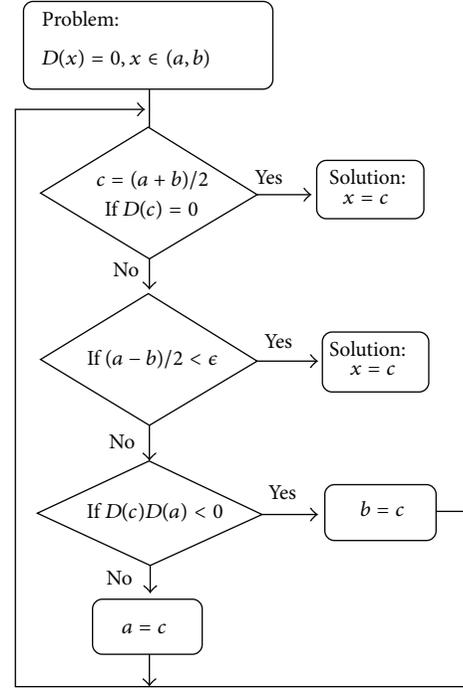


FIGURE 1: Flow chart for bisection method.

4. Structure of Algorithm

Structure of our methodology is a nested two-level looping. The inner looping is a collection of evaluating results for a particular set of parameters of a traffic flow model, and the outer looping is an implementation of bisection method for search of a set of optimized parameters. And before the nested looping we aggregated data first.

4.1. *Data Aggregation.* To reduce running time of this algorithm, aggregation of empirical data should be the first step. We aggregate raw field observation data by averaging 3D data values along an axis. For example, a set of speed-density-flow data are aggregated by averaging speed, density, and flow values of all observations which fall into a section $k^{(i)} < k < k^{(i+1)}$, where $k^{(i)}$ and $k^{(i+1)}$ are lower and upper bounds of i th density section. And this procession can be expressed as

$$\begin{aligned} \bar{v}_i &= \frac{1}{n} \sum_{j=1}^{n_i} (v_{ij}), \\ \bar{k}_i &= \frac{1}{n} \sum_{j=1}^{n_i} (k_{ij}), \\ \bar{q}_i &= \frac{1}{n} \sum_{j=1}^{n_i} (q_{ij}), \end{aligned} \tag{4}$$

where footnote i means this calculation is for i th density slice, n_i is number of raw observations in this slice, $\bar{v}_i, \bar{k}_i,$ and \bar{q}_i are averaged values of $v, k,$ and q for i th slice, respectively, and $v_{ij}, k_{ij},$ and q_{ij} are coordinates for j th raw observation in i th slice of v - k - q coordinate system.

4.2. Inner Looping: Evaluation of Parameters

4.2.1. *Derivation of Objective Function.* A calibration criterion has been raised by Rakha and Arafeh [20] and is formulated as

$$\text{Min } E = \sum_i \left\{ \left(\frac{v_i - \hat{v}_i}{\bar{v}} \right)^2 + \left(\frac{q_i - \hat{q}_i}{\bar{q}} \right)^2 + \left(\frac{k_i - \hat{k}_i}{\bar{k}} \right)^2 \right\}, \quad (5)$$

where q_i , k_i , and v_i are flow, density, and average speed data from field tests, respectively. \hat{q} , \hat{k} , and \hat{v} are flow, density, and average speed derived from a model, as \bar{q} , \bar{k} , and \bar{v} are maximum value of flow, density, and speed observed in filed. This is a very good estimation for models simulating traffic flows; however, two concerns are still worth more research for solutions:

- (1) there is still room for improvement on time and resource efficiency. It is acceptable to compromise on accuracy of an algorithm in order to get it faster;
- (2) all aggregated data dots receive the same treatment, while in fact some of the dots are aggregated from more raw field observations than others, as there are many more vehicles travelling around 60 mph than those opt 30 mph on the same section of highway.

Regarding issue 1, since $q_i = k_i \times v_i$, also approximately $\hat{q}_i \approx \hat{k}_i \times \hat{v}_i$ and $\bar{q}_i \approx \bar{k}_i \times \bar{v}_i$, thus we consider flow q in this evaluation criterion redundant, which means we can get the same result by switching to criterion 6 instead of 5:

$$\text{Min } E = \sum_i \left\{ \left(\frac{v_i - \hat{v}_i}{\bar{v}} \right)^2 + \left(\frac{k_i - \hat{k}_i}{\bar{k}} \right)^2 \right\}. \quad (6)$$

As shown in Section 4.1, we make narrow slices on k - v coordinate system to get groups of data for aggregation, and here we presume it is k -axis we slice to group data, so each observation which falls into i th group satisfies $k^{(i)} < k \leq k^{(i+1)}$, where $k^{(i)}$ and $k^{(i+1)}$ are lower and upper bounds of i th density section. According to (4), we get aggregated dot's coordinate on k -axis by averaging all raw observations' corresponding coordinates; thus, if we opt for middle point between $k^{(i+1)}$ and $k^{(i)}$, instead of \bar{k}_i , to approach k_i the corresponding error would be

$$\begin{aligned} \text{err} &\leq \frac{\epsilon}{2}, \\ \epsilon &= k^{(i+1)} - k^{(i)} \\ \text{when } k_i &= \frac{k^{(i+1)} + k^{(i)}}{2}, \end{aligned} \quad (7)$$

where err is error derived from approximation for k_i by middling i th slice and ϵ is width of i th slice. So obviously if we set ϵ to be a small enough number it is possible to opt for (7) as approximation for k_i without losing too much accuracy. In our simulation we choose $\epsilon = 0.5$ vehicle/km. In this way

we avoid calculating average value of a set of observations as shown in (4).

By narrowing slices on k -axis, another obvious strategy is intuitive that, in calculation for (6), we can ignore the second term which is square of relative difference for density k . This is because in a slice relative difference between densities of a raw observation and the aggregated dot is very small, especially compared to that between speeds of the same dots. This can be proved by computing curve of the following function (8), which is the relationship between two terms in (6):

$$f = \frac{\sum_i ((v_i - \hat{v}_i) / \bar{v})^2}{\sum_i ((k_i - \hat{k}_i) / \bar{k})^2}. \quad (8)$$

This curve is shown in Figure 2. Also this figure perfectly explains drivers' different behaviour patterns corresponding to different density rates. When density is very low (lower than 7 vehicles per km), drivers opt for different speeds freely according to their own preferences, so speeds under such situations are distributed widely because of big variance on people's personal preferences. When there are some more vehicles on road but still not too many (8 to 24 vehicles per km), traffic flows travel fast, so under such circumstances drivers who prefer driving slowly are encouraged to drive faster to follow the flow. And as density grows higher to around 40 vehicles per km, average speed decreases remarkably. Actually average speed at this stage is so low that a significant proportion of drivers opt for fast lanes for higher speed and overtaking, and this causes a peak at 40 vehicles per km. As density goes up, there is less room for drivers' maneuvers including lane changing, accelerating, and braking; thus, there are less people driving aggressively, and this makes the curve go down gradually at density higher than 40 vehicles per km.

Thus, we can delete second term in criterion 6 without losing too much on precision of a model. In this way we improve it

$$\text{Min } E = \sum_i \left(\frac{v_i - \hat{v}_i}{\bar{v}} \right)^2. \quad (9)$$

Actually here we do not need the denominator \bar{v} , as for an optimization problem, dividing its objective function by a constant does not make any difference to the result. Similarly, replacing square operation by calculating absolute value will not affect optimization result since minimizing an equation's square leads to the same point of that by minimizing its absolute value. Thus, (10) reflects exactly the same criterion as (9):

$$\text{Min } E = \sum_i |v_i - \hat{v}_i|. \quad (10)$$

Here v_i is speed value for an aggregated field observation and \hat{v}_i is the estimated speed which corresponds to density value of the same aggregated observation. This estimation of speed is done by applying macroscopic traffic flow simulation models.

By opting for (10) as objective function, it is convenient to calculate its value without any other looping, whilst Rakda's

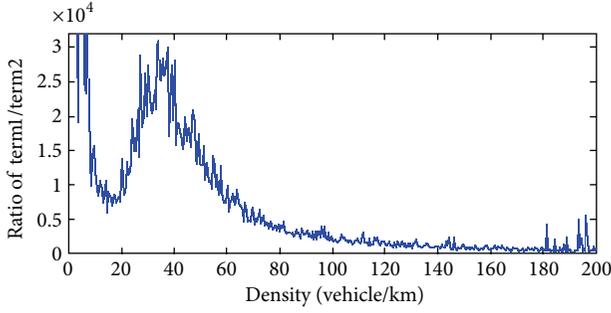


FIGURE 2: Curve computed from function (8).

method with objective function (5) requires another level of optimization to find a point (q'_i, k'_i, v'_i) which is the nearest one on a curve to an aggregated observation (q_i, k_i, v_i) . Usually this kind of optimization uses a looping to try a series of points on the curve; after calculating distance from every one of this series of points to the observation (q_i, k_i, v_i) , the point with the least distance is considered as the nearest point from the curve to (q_i, k_i, v_i) . In our algorithm this level of optimization can be avoided; thus, a considerable amount of time is saved from this step.

4.2.2. Conversion to a Root-Finding Problem. In order to implement bisection method in later Section 4.3, it's necessary to convert this optimization problem into a root-finding problem. Key to this conversion is finding a function which has a root in a domain of its parameters, and such a root should be very near, if not the same, to solution of the optimization problem solved with objective function (10). Thus, we assign positive and negative signs to results of inner looping. In the case discussed above, if an aggregated observation point locates at left side of corresponding speed-density curve we multiply the result of its objective function by -1 , and if it sits at right side then we keep it as positive. Actually this strategy can be applied simply by removing absolute value calculation in objective function (10). So the root-finding problem is as follows:

$$F = \sum_i v_i - \hat{v}_i. \quad (11)$$

In this function every aggregated observation is considered equally important; however, in reality we do not agree with this assumption, since during most time vehicles travel in traffic with density less than 50 vehicles per km. Our observation data reveals that more than 60% traffic happens with density lower than 50 vehicles per km, and this can be found in following Figure 3.

So it is appropriate to multiply every aggregated observation with a term of weight which is number of raw observations it is aggregated from. So we have the final version of the root-finding problem as

$$F = \sum_i n_i (v_i - \hat{v}_i), \quad (12)$$

where n_i is the number of raw observations which fall into i th slice of k - v plane, and these raw observations are used to

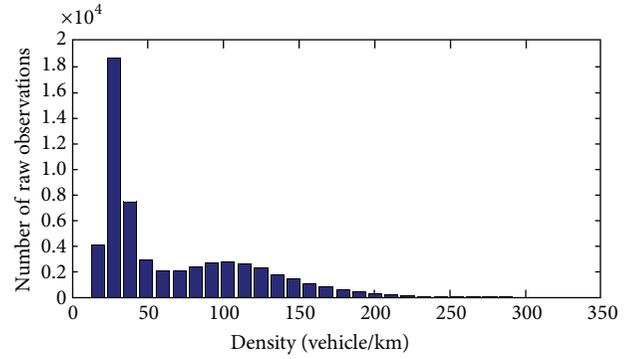


FIGURE 3: Number of raw observations which happened at different density levels.

produce i th aggregated observation by applying (4). Here we name result of this function degree of satisfaction for a set of parameters, and purpose of inner looping is to find degree of satisfaction for a specific set of parameters, whilst outer looping explained in Section 4.3 is to find a set of parameters which has its degree of satisfaction the nearest one to zero.

4.3. Outer Looping: Bisection Method. As we get degree of satisfaction of every set of parameters in inner looping, different parameter sets get evaluated by running outer looping. Goal of this evaluation is to find a set of model parameters which let right side of (12) equal 0. So the problem here is to find a root for this function, and intuitively numerical analysis methods can help because it is not a simple linear function. Here we adopt bisection method because of its simplicity and robustness. Using LCM model as an example to calibrate,

$$k = \frac{1}{(\gamma v^2 + \tau v + l) [1 - \ln(1 - (v/v_f))]}, \quad (13)$$

where k is traffic density, v is space-mean speed, γ denotes aggressiveness that characterizes the driving population, τ is average response time that characterizes the driving population, l is average effective vehicle length, and v_f is free flow speed. So in this model we need to allocate four parameters γ , τ , l , and v_f . As an example, steps of this bisection algorithm to optimize γ include the following.

- (i) Step 1: if any of τ , l , and v_f has already been optimized before γ , set these parameters as their calibrated values. And for those that have not been calibrated, set default parameter values. Default values are best guesses based on common sense and earlier experience. For γ we set lower bound γ_l and upper bound γ_u . Set acceptable largest error of gamma ϵ .
- (ii) Step 2: if $\gamma_u - \gamma_l < \epsilon$, then return $\gamma = (\gamma_l + \gamma_u)/2$ as result of this algorithm and quit algorithm. Otherwise, run inner looping with $\gamma = (\gamma_l + \gamma_u)/2$ and other parameter values as set in step 1 for LCM model; get degree of satisfaction for this set of parameters with new γ value.
- (iii) Step 3: if degree of satisfaction value got in step 2 is greater than 0, it means we underestimated drivers'

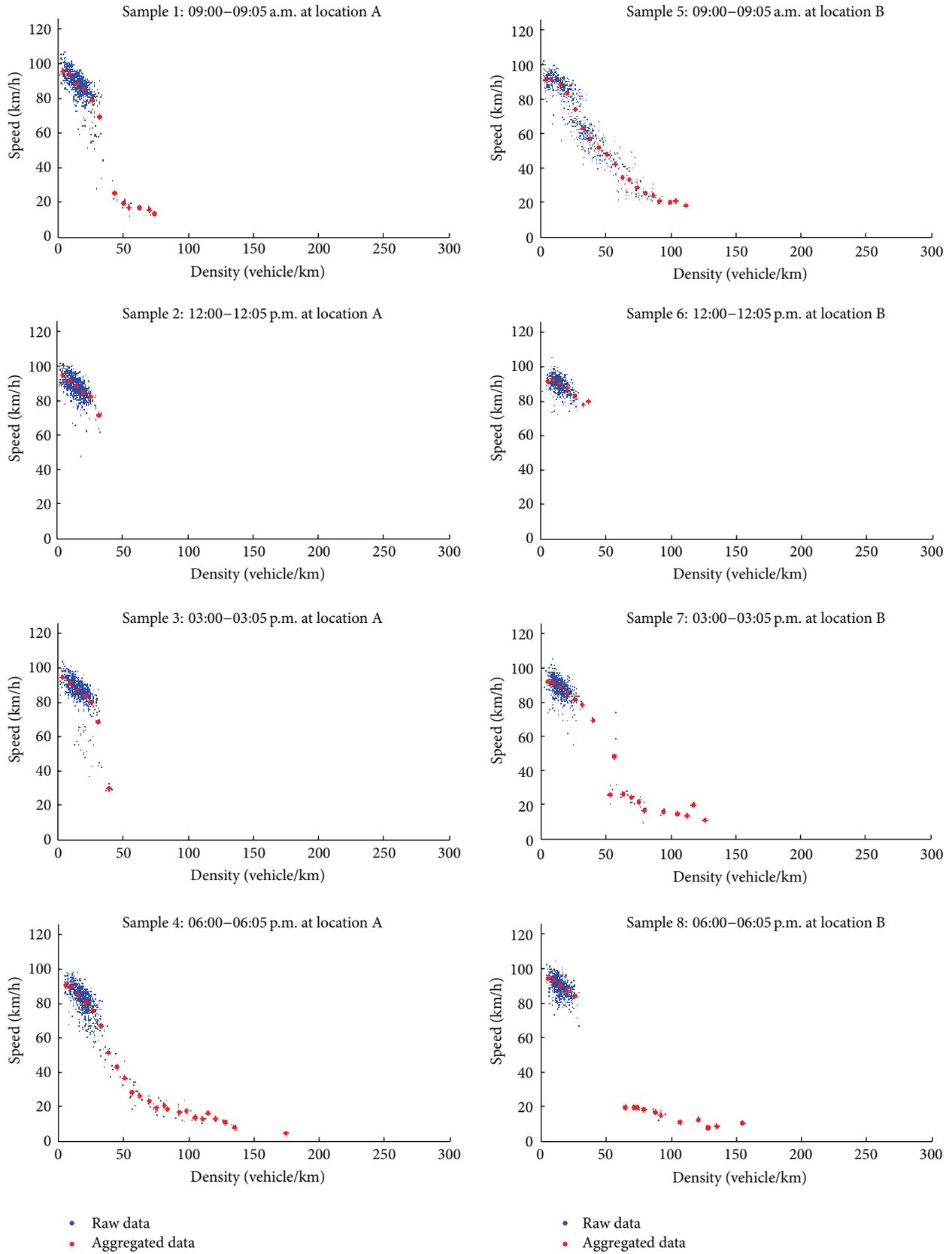


FIGURE 4: Raw and aggregated observation points.

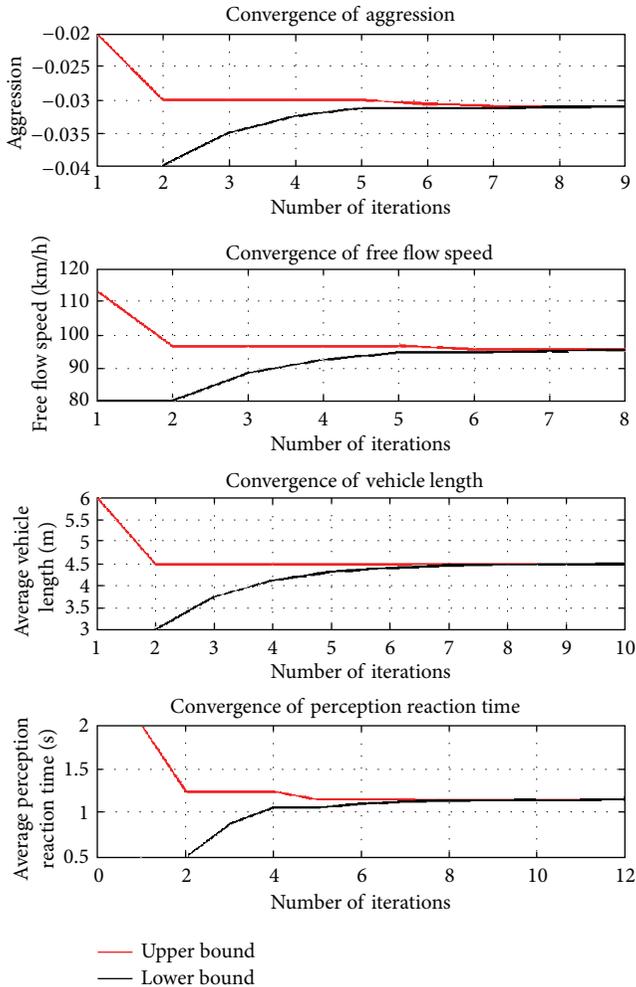


FIGURE 5: Steps for parameters to converge in bisection calibrating method.

aggression at this particular time and location; thus, reset lower bound of aggressiveness as $\gamma_l = (\gamma_l + \gamma_u)/2$, keep other parameters unchanged, and go to step 2. Otherwise, it means we overestimated drivers' aggressiveness so reset upper bound as $\gamma_u = (\gamma_l + \gamma_u)/2$ and go to step 2.

With this bisection algorithm we will finally get a calibrated γ value when sum of signed distances is very near to 0, which means our calibration criterion is approximately satisfied. And repeat this for every model parameter.

5. Verification and Analysis

To verify and analyse the approach introduced in Sections 3 and 4, we pick eight sample sets of 5-minute traffic data to verify and analyse our algorithm. These samples are cut from a set of traffic flow data logs for route 400 in Georgia in July 2003, containing information collected by two roadside cameras. Four of the 5-minute samples derived from logs were collected by camera 1106 near North Old Milton Parkway, Alpharetta, GA (location A), and the other

four are related to logs of camera 1134 near South Hammond Drive, Sandy Springs, GA (location B). These traffic data logs contain raw observations of flow, density, and speed with time stamps. Every raw observation was for a 20-second period, so information contained by an observation is average rate of flow, density, and speed of traffic which went through a camera's shooting area during a specific 20-second period. Thus, this data set is capable of simulating traffic situation in real time except one fact that, size of data log in reality would be much bigger than our data set since we assume real-time data stream is observations for individual vehicles, not averaged traffic during periods. To solve this problem we aggregate data from logs of different days in July into one sample. For example, in order to get a sample for traffic during 9:00–9:05 am at camera 1106, we aggregate all traffic data for this site and this time of day on all days in July into one single file as a simulation of real-time observation of individual vehicles. Since traffic flow characters should be consistent for the same site under the same illumination, it is reasonable to simulate by aggregating.

In order to get samples without any preference, for each site we pick four 5-minute segments of the logs with a 3-hour step; that is, these four samples are log segments of traffic data at 9:00–9:05 am, 12:00–12:05 pm, 3:00–3:05 pm, and 6:00–6:05 pm, respectively.

5.1. Data Aggregation and Weighting. As the first step, data aggregation contributes remarkably to saving running time. 150 slices on k -axis are made on interval $k \in [0, 300]$; however, consider that there are slices where no raw observations fall in; actual number of aggregated observation points is less than 150. As shown in Figure 4, there are 15–70 aggregated points that depend on span of raw data's density value. Also from this figure we see obvious patterns of traffic flow in different time of a day. In the morning majority of traffic travelled at density $k < 30$ vehicles per km; higher density happened sometimes and was not ignorable, especially for camera 1134 at location B. This is just the normal situation on highway; sequences of vehicles come with considerable high internal density but there are long distances between different sequences. And sometimes slow leading vehicles of sequences make traffic slow down. During noon time traffic was more concentrated to density under 30, sparse points exist at higher density and lower speed, so it is still normal situation with minor differences. Traffic situation in afternoon was significantly different. For sample data at 3 pm, there are, but not many, observations with densities higher than 30 vehicles per km, so we consider that most vehicles were travelling in smooth and confident maneuvers. However, traffic at 6 pm at location A was a contrast, with a large proportion of observations distributed with high density and low speed, so this is a rush hour situation.

Patterns for location A and location B are not exactly the same, and this might be because a camera is only for surveillance of one side of highway, as well as the fact that camera 1106 is at the side of location A which is more congested during afternoon rush hour, and camera 1134 occupies the side of location B that accommodates morning crowds.

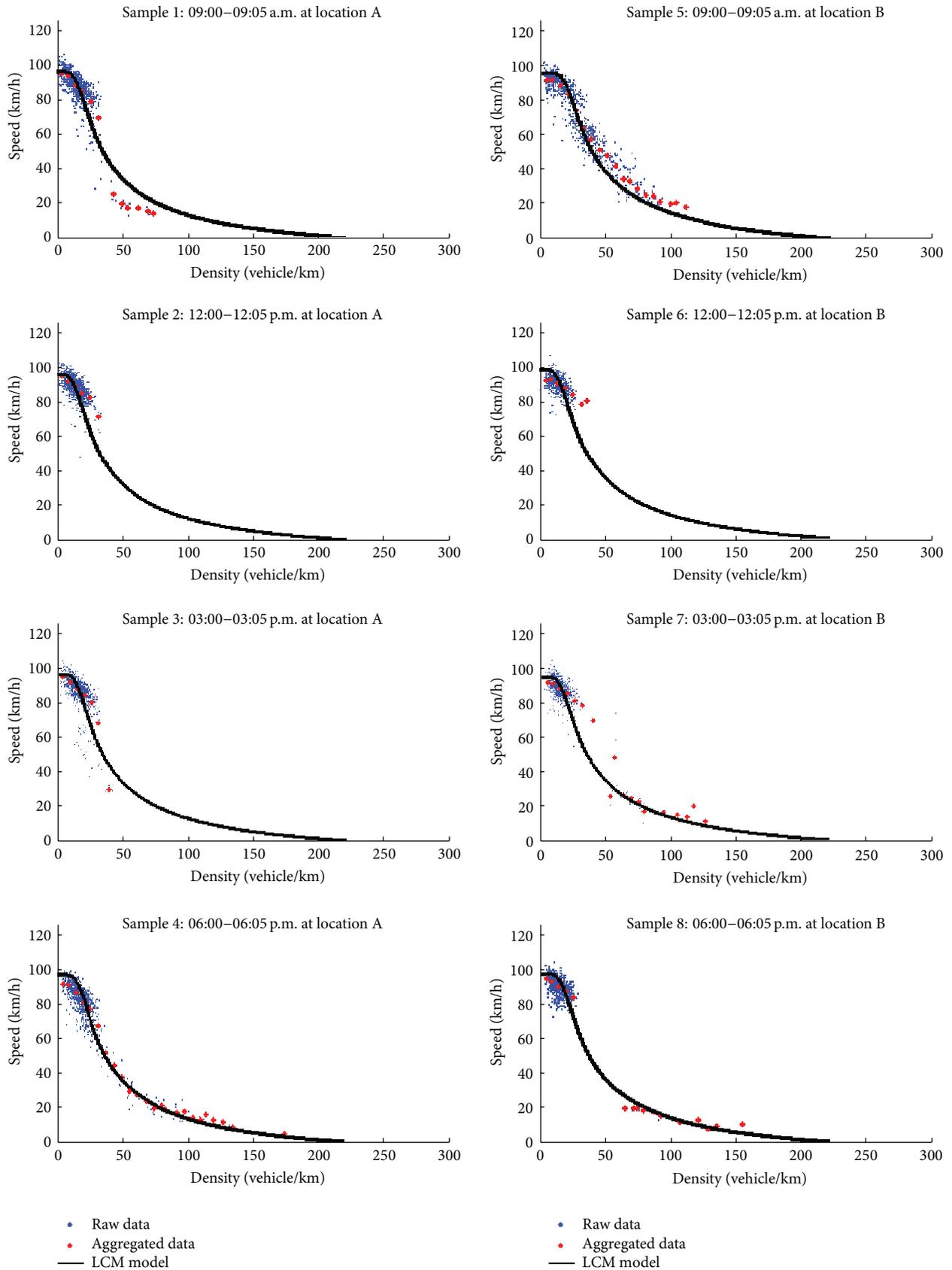


FIGURE 6: Calibration results for sample data sets by bisection based algorithm.

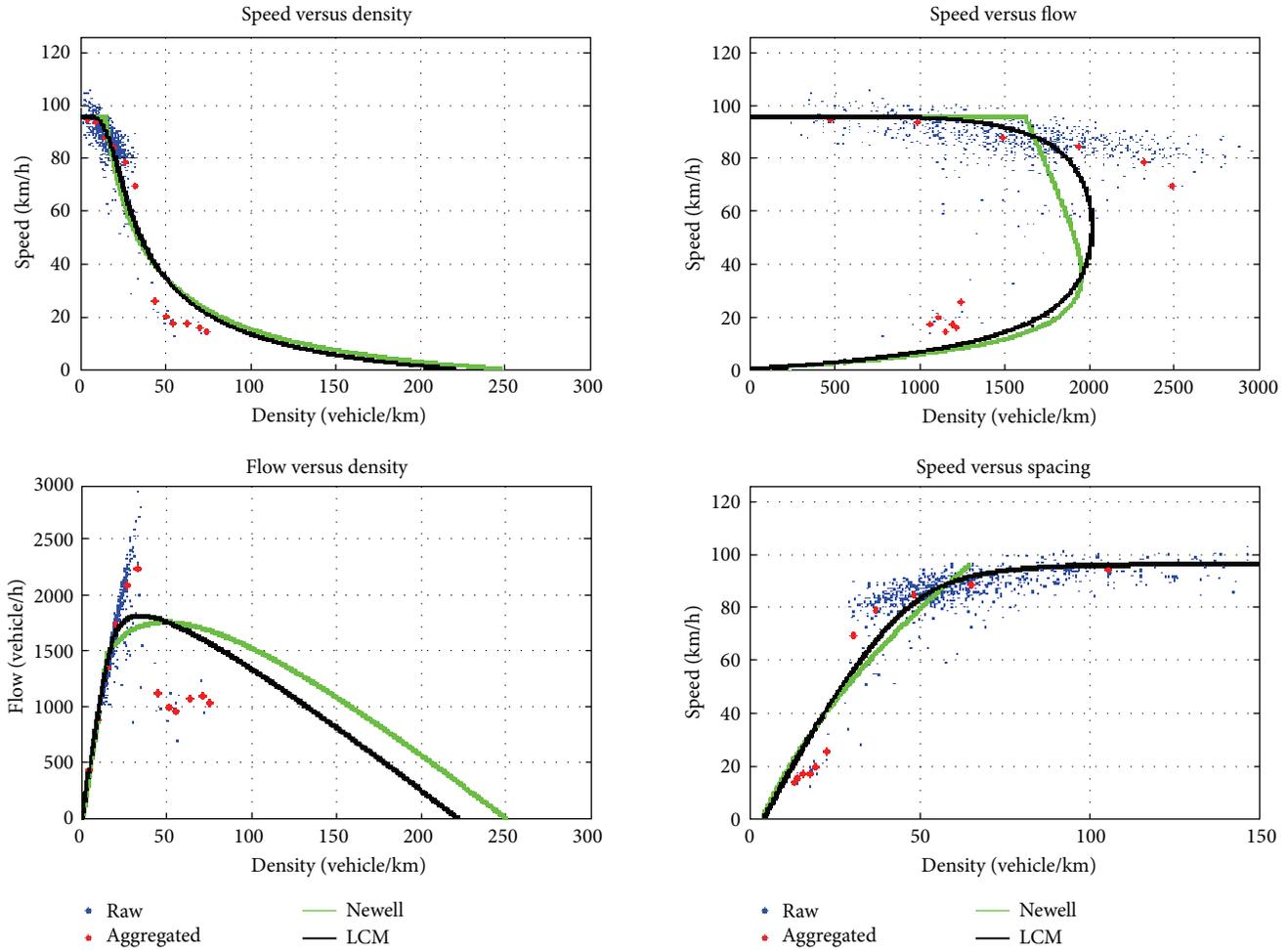


FIGURE 7: LCM and Newell's model calibrated by bisection algorithm.

Also from above Figure 4 we see importance of applying weights to aggregated observation points. Take noon sample at location B as an example; the red point with density at 39 vehicles per km is aggregated from only 1 raw observation, while another red point at 16 vehicles per km is aggregated from 33 raw observations, so intuitively these two points should have different effects on shape of a flow model. In order to generate a weight for an aggregated point, we use the number of raw observations from which this aggregated point is calculated. By adapting this definition of weights for points, inputs from different situations are reasonable.

5.2. Convergence Speed for Bisection Method. Though bisections are not the most advanced numerical analytical method, consider its time complexity as $O(\log_2 n)$; it meets our requirement of performing in real time for transportation engineering and administration purpose. As shown in Figure 5, to calibrate against location B 9 am sample data, every parameter of LCM model converges in at most 12 iterations, so its guaranteed running time is less than a second. Compared to other nonnumerical methods, it is a very fast algorithm for calibrating traffic flow models, and practically this calculating system reacts in real time.

5.3. Calibration Result. In Table 2, we get quantified patterns for sample data sets by calibrating LCM model. As we assumed, parameters of LCM model appear to be consistent for a specific location, and for these two cases, as there are both near-straight sections without any obvious curve, there is no significant difference between values of their parameters. Driving patterns at location A are especially more consistent compared to that of location B, considering that location B is near conjunction of route 285 and route 400, so there is more interference coming from the complicated structure of this conjunction.

Thus, calibration result of our algorithm meets common sense, and Figure 6 demonstrates how speed-density curve of LCM model fits field data after calibration. This figure further proves role of weighting aggregated points as discussed in Section 5.1, especially in sample 1 for location A at 9 am; six red points between density 40 and 75 vehicles per km are considered less important so curve of final result does not go through these points perfectly; however, this makes overall precision of the model optimized.

A more comprehensive visualized result for this bisection based algorithm is shown in a following page in Figure 7. Sample 2 for location B at 9 am is opted for this demonstration

TABLE 2: Calibration results for sample data sets by bisection based algorithm.

Samples	Parameters				
	Free flow speed (km/h)	Vehicle length (m)	P-R time (sec)	Aggression (s^2/m)	
Location A	9:00–9:05 a.m.	96.1628	4.5088	1.2438	-0.0305
	12:00–12:05 p.m.	95.6600	4.5088	1.3331	-0.0311
	3:00–3:05 p.m.	96.4143	4.5029	1.2892	-0.0301
	6:00–6:05 p.m.	96.6655	4.5029	1.1903	-0.0300
Location B	9:00–9:05 a.m.	95.4087	4.4971	1.1544	-0.0310
	12:00–12:05 p.m.	98.1741	4.4912	1.2240	-0.0286
	3:00–3:05 p.m.	95.9115	4.4971	1.2592	-0.0305
	6:00–6:05 p.m.	97.6712	4.5029	1.1720	-0.0290

because its data points are better spread; thus, it is easier to read how calibration results fit field and aggregated data. Also Newell's model is calibrated to prove versatility of our algorithm.

6. Conclusion

Consider its time complexity as $O(\log_2 n)$; this bisection based algorithm has a theoretical high performance. In practice, we run this algorithm on a personal laptop with a T7300 CPU and 2 GB ram; calibration for LCM model against every sample completes in 0.5 seconds, and calibration for Newell's model runs for less than 1 second. Also because of aggregation procedure, this algorithm is not sensitive to size of sample, as it calibrates both models against a 24-hour data set with 56,000 observations in less than 2 seconds. Also by verifying and analysing the algorithm's procedures and results in Section 5, we see it has good accuracy in both calculating parameters of a model and drawing statistical relationships between traffic flow, density, and speed. Thus, it is reasonable to use this calibration algorithm on real-time surveillance of drivers' behaviours.

Conflict of Interests

The authors declare that they have no conflict of interests regarding the publication of this paper.

References

- [1] NHTSA's National Center for Statistics and Analysis, "Traffic safety facts 2009 data, pedestrians part," Tech. Rep., U.S. Department of Transportation, National Highway Traffic Safety Administration, Washington, DC, USA, 2010.
- [2] J. G. Pigman and K. R. Agent, "Highway accidents in construction and maintenance work zones," *Transportation Research Record*, no. 1270, pp. 12–21, 1990.
- [3] Federal Highway Administration, *How Do Weather Events Impact Roads?*, 2011.
- [4] B. D. Greenshields, J. R. Bibbins, W. S. Channing, and H. H. Miller, "A study of traffic capacity," in *Proceedings of the 14th Annual Meeting of the Highway Research Board*, vol. 14, pp. 448–477, Highway Research Board, Washington, DC, USA, 1935.
- [5] G. F. Newell, "Nonlinear effects in the dynamics of car following," *Operations Research*, vol. 9, no. 2, pp. 209–229, 1961.
- [6] G. F. Newell, "A simplified theory of kinematic waves in highway traffic, part II: queueing at freeway bottlenecks," *Transportation Research B*, vol. 27, no. 4, pp. 289–303, 1993.
- [7] R. E. Chandler, R. Herman, and E. W. Montroll, "Traffic dynamics: studies in car following," *Operations Research*, vol. 6, no. 2, pp. 165–184, 1958.
- [8] D. C. Gazis, R. Herman, and R. B. Potts, "Car-following theory of steady-state traffic flow," *Operations Research*, vol. 7, no. 4, pp. 499–505, 1959.
- [9] P. G. Gipps, "A behavioural car-following model for computer simulation," *Transportation Research B*, vol. 15, no. 2, pp. 105–111, 1981.
- [10] I. Kosonen, *HUTSIM—urban traffic simulation and control model: principles and applications [M.S. thesis]*, Helsinki University of Technology, Espoo, Finland, 1999.
- [11] S. Kikuchi and P. Chakroborty, "Car-following model based on fuzzy inference system," *Transportation Research Record*, no. 1365, pp. 82–91, 1992.
- [12] D. Ni, C. Zhang, and H. Wang, "A unified perspective on traffic flow theory, part III: validation and benchmarking," in *Proceedings of the 11th International Conference of Chinese Transportation Professionals (ICCTP '11)*, pp. 4264–4286, ASCE, Nanjing, China, August 2011.
- [13] D. Ni, "A unified perspective on traffic flow theory, part II: the unified diagram," in *Proceedings of the 11th International Conference of Chinese Transportation Professionals (ICCTP '11)*, pp. 4244–4263, ASCE, Nanjing, China, August 2011.
- [14] D. Ni, "A unified perspective on traffic flow theory, part I: the field theory," in *Proceedings of the 11th International Conference of Chinese Transportation Professionals (ICCTP '11)*, pp. 4227–4243, ASCE, Nanjing, China, August 2011.
- [15] R. Herman, E. W. Montroll, R. B. Potts, and R. W. Rothery, "Traffic dynamics: analysis of stability in car following," *Operations Research*, vol. 7, no. 1, pp. 86–106, 1959.
- [16] M. Brackstone and M. McDonald, "Car-following: a historical review," *Transportation Research F*, vol. 2, no. 4, pp. 181–196, 1999.
- [17] G. G. Schultz and L. R. Rilett, "Analysis of distribution and calibration of car-following sensitivity parameters in microscopic traffic simulation models," *Transportation Research Record*, vol. 1876, pp. 41–51, 2004.

- [18] A. Kesting and M. Treiber, "Calibrating car-following models by using trajectory data methodological study," *Transportation Research Record*, vol. 2088, pp. 148–156, 2008.
- [19] S. P. Hoogendoorn and R. Hoogendoorn, "Generic calibration framework for joint estimation of car-following models by using microscopic data," *Transportation Research Record*, vol. 2188, pp. 37–45, 2010.
- [20] H. Rakha and M. Arafah, "Calibrating steady-state traffic stream and car-following models using loop detector data," *Transportation Science*, vol. 44, no. 2, pp. 151–168, 2010.