Car-following Models: Data-driven Model

CIVE.5490, UMass Lowell

Zhengbing He, Ph.D.
https://www.GoTrafficGo.com

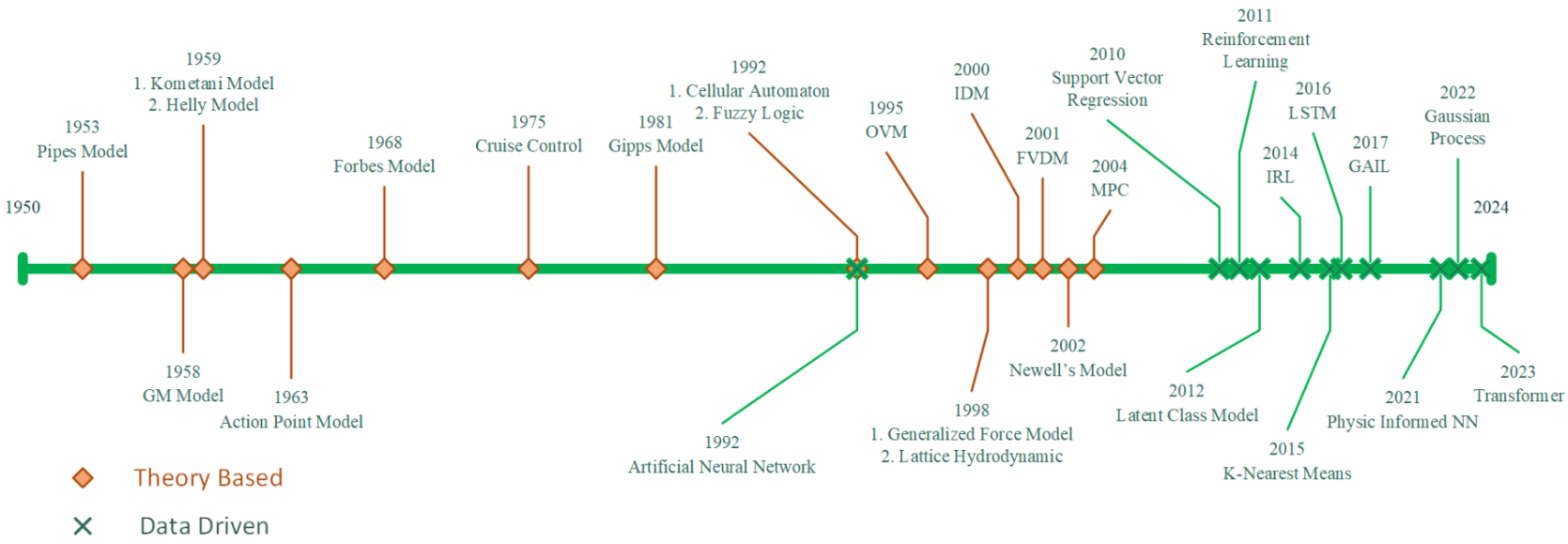
March 29, 2024

Outline

Introduction

- Zhenbing He, Liang Zheng, Wei Guan (2015). A simple nonparametric car-following model driven by field data. *Transportation Research Part B: Methodological*, 80(2015), 185-201

Introduction



Tianya Zhang, et al., Car-Following Models: A Multidisciplinary Review, arXiv:2304.07143v, 2024

Introduction

Existing car-following model, such as Gipps, IDM, Full velocity model

- Ordinary differential equation (ODE) with speed and acceleration variables.
- Fundamental diagrams and driver's behavior parameters.
- Calibration is needed before practical usage.
- Should be able to reproduce major traffic characteristics.

Introduction

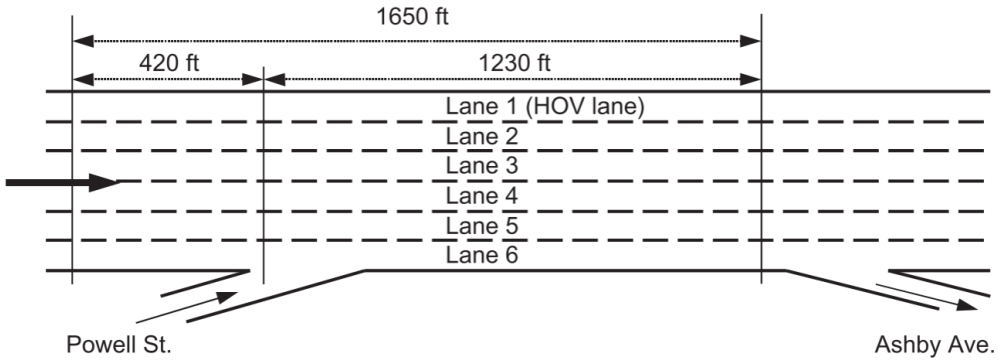
Contributions of this work:

- A data-driven car-following model based on k-nearest neighbor
- Neither mathematical equation nor calibration is needed
- Neither fundamental diagrams nor driver's behavior parameters is assumed
- The model is simple and parsimonious, because there is only one parameter
- The model is able to well reproduce important traffic characteristics

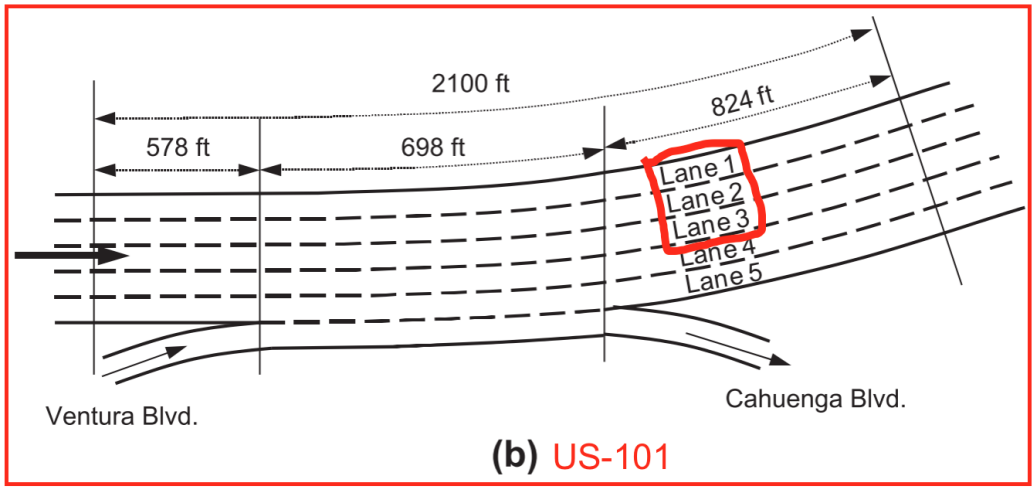
Outline

NGSIM dataset

- Open-source: about 1 hour trajectory dataset, 2005

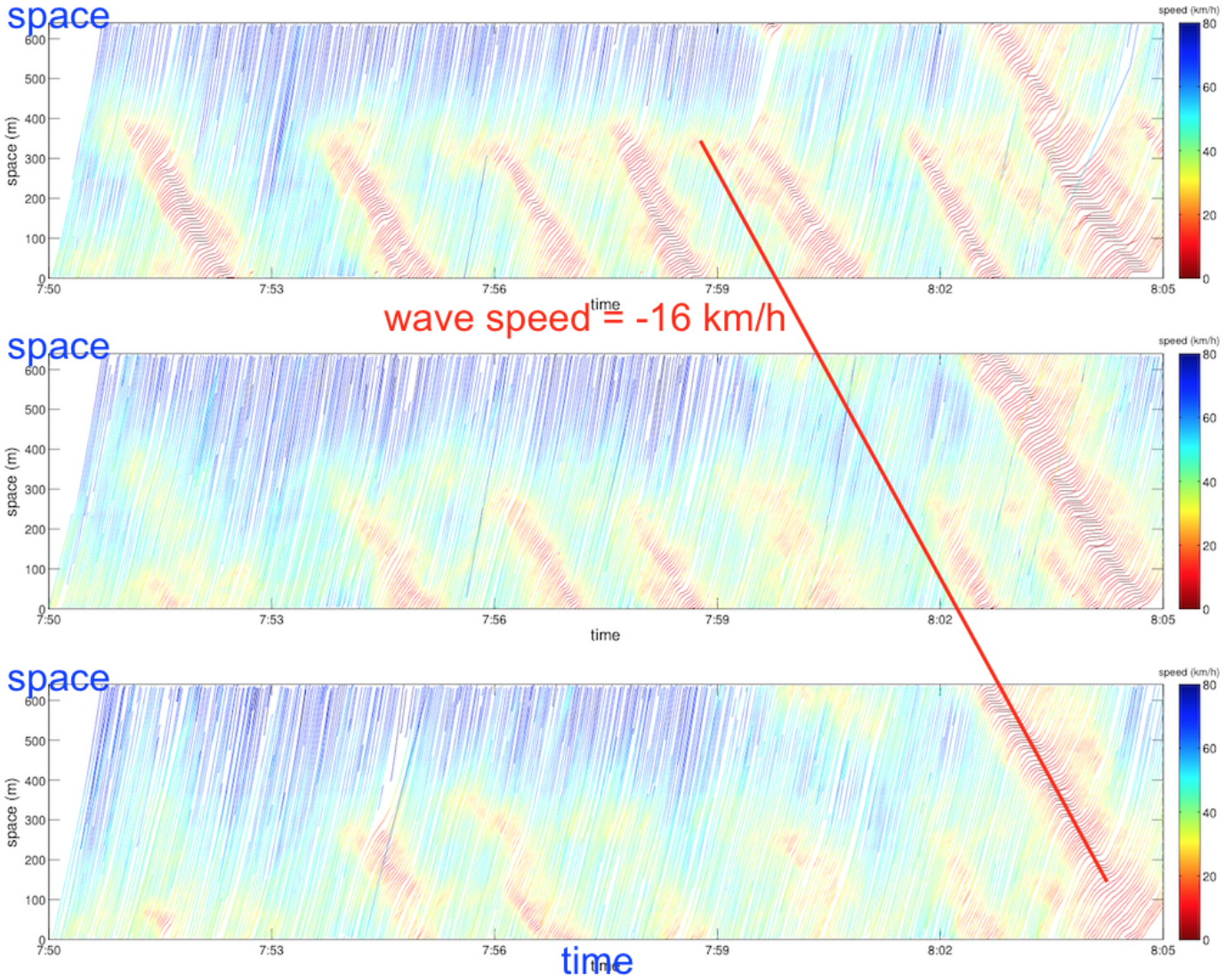


(a) I-80



(b) US-101

NGSIM dataset



A nonparametric approach: k-nearest neighbor

Top 10 Algorithms every Machine Learning Engineer should know

1. Naïve Bayes Classifier Algorithm

2. K Means Clustering Algorithm

3. Support Vector Machine Algorithm

4. Apriori Algorithm

5. Linear Regression Algorithm

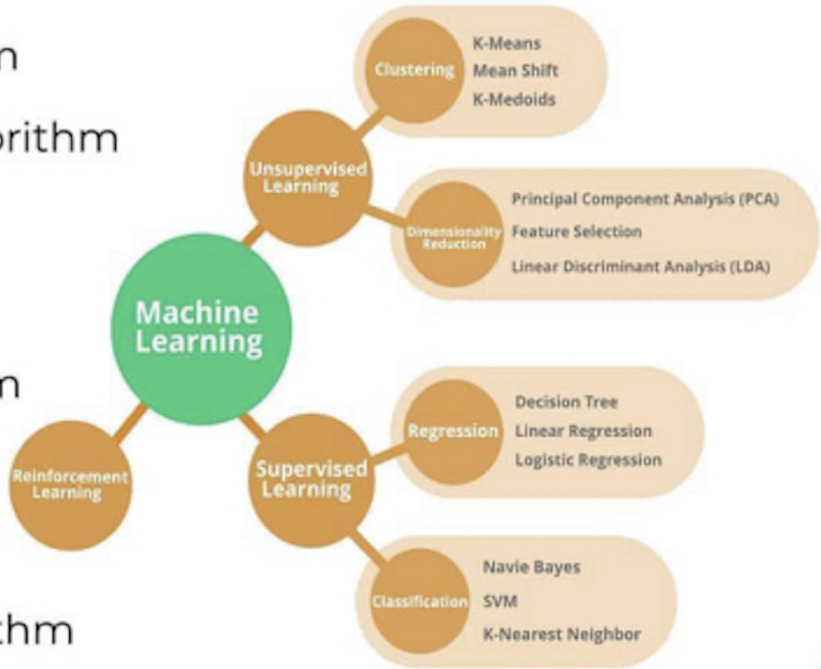
6. Logistic Regression Algorithm

7. Decision Trees Algorithm

8. Random Forests Algorithm

9. K Nearest Neighbours Algorithm

10. Artificial Neural Networks Algorithm



A nonparametric approach: k-nearest neighbor

- Nonparametric approach/model
  - Parametric approach: based on mathematic formulas
  - Nonparametric approach: no model, driven by historical data
- k-nearest neighbor: very simple but works well
  - History is repeating: Most similar conditions (input) highly likely result in similar outcome (output).
  - For example:
    - Wind, temperature, traffic $\rightarrow$ magnitude of haze.
    - Looking up similar days in history based on wind, temperature, traffic today
    - Taking the average magnitudes of haze in these days as the estimator of the haze today.

k-nearest neighbor

- The approach selects the most similar historical cases, and takes the average of their outputs as the estimate of this time.
- Specifically, the approach estimates $\mathbf{y}_0$ in focal $(\mathbf{x}_0, \mathbf{y}_0)$ as follows.

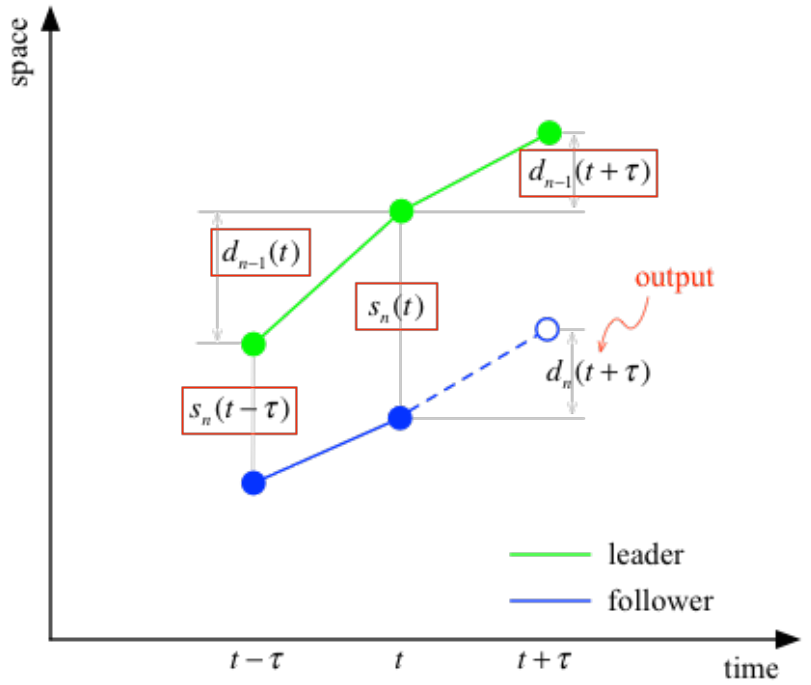
$$\hat{\mathbf{y}}_0 = \frac{\sum_{i=1}^{k} \mathbf{y}_i}{k} \tag{1}$$

where $\mathbf{x}_i$ with respect to $\mathbf{y}_i$ is one of the k most similar samples to $\mathbf{x}_0$.

Outline

The model: illustration

A key to understand the nonparametric model:



input: leader's two-step speed + follower's two-step spacing
output: follower's speed

The model: mathematical expression

- Input:

$$\mathbf{x}_n(t+\tau) = (d_{n-1}(t+\tau),\ d_{n-1}(t),\ s_n(t),\ s_n(t-\tau)). \quad (2)$$

where τ is the simulation time step; $(n-1)$ is the leader of vehicle n; $d_{n-1}(t)$ is the moving distance of the leader between time $(t-\tau)$ and t

- Output: moving distance of vehicle n

$$\mathbf{y}_n(t+\tau) = d_n(t+\tau) \quad (3)$$

Elimination of autocorrelation

Because of strong autocorrelation of time-series trajectory data, it is easy for all k samples coming from a leader-follower pair, in particular when the leader and follower move in a constant speed. Such dominance could reduce the reliability of the model. To overcome this issue, we make all k samples selected from different leader-follower pairs.

Distance between two data samples

- "Ordinary" and simple scaled Euclidean distance, i.e., adjusting the input $x_{ji} \in \mathbf{x}_i$ by its mean $\bar{x}_j$ and standard deviation S_j before calculating Euclidean distance
- The model reads

$$\mathscr{D}(\mathbf{x}_i, \mathbf{x}_0) = \sqrt{\sum_{j=1}^{J} (z_{ji} - z_{j0})^2} \tag{4}$$

where

$$z_{ji} = \frac{x_{ji} - \bar{x}_j}{S_j}, \tag{5}$$

and J is the total number of all elements in an input vector.

Outline

Determination of k: Introduction

- k in kNN: the number of the historical cases that are considered to be similar to the estimated case.
- Usually, k is estimated by experience.
- Here, we determine k by comparing estimation errors under different k-values.
- Employ the scaled Euclidean distance from kth nearest sample to the estimated case, which is the longest distance in all k samples, Denote by $\mathscr{D}_k$.

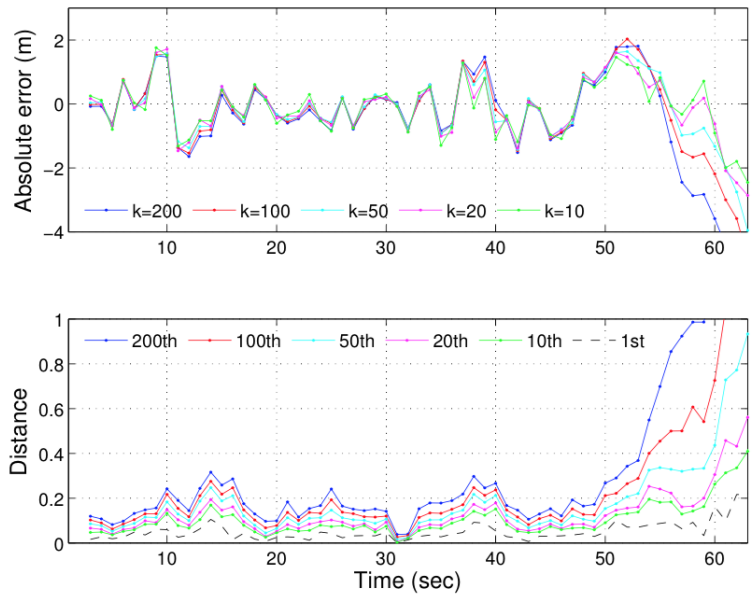
Determination of k: Four databases

We first build three databases with different sizes by using the datasets collected on Lane 2 and 3. Such databases are employed to estimate the movement of the followers on Lane 1.

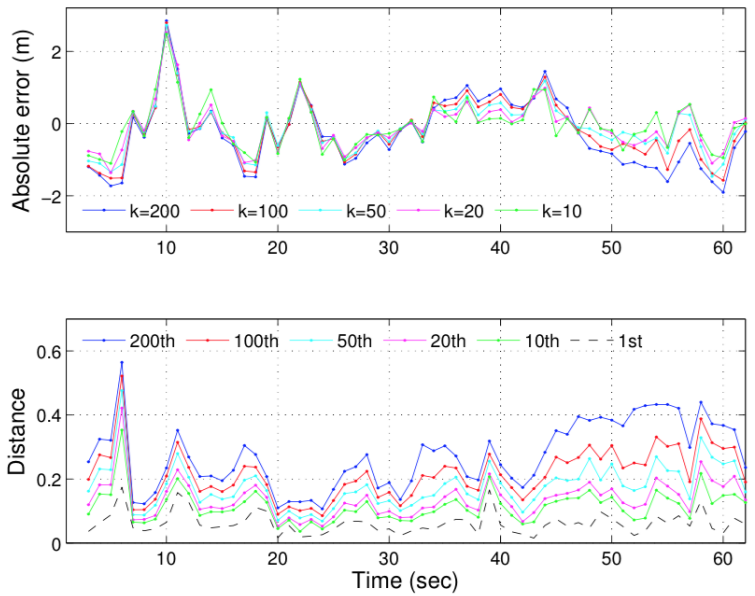
- Database 1: the small-size database containing 22,202 input-output samples, which is built by using the dataset collected on Lane 2 during the first 15 minutes;
- Database 2: the medium-size database containing 78,683 samples, which is built by using the dataset collected on Lane 2 during the all study 45 minutes;
- Database 3: the large-size database containing 152,637 samples, which is built by using the dataset collected on Lane 2 and 3 during the all study 45 minutes.

Determination of k: First scenario

- We estimate the followers of two typical vehicles who traverse stop-and-go oscillations on Lane 1
- Database 3 is used, and the absolute errors with different k-values are compared



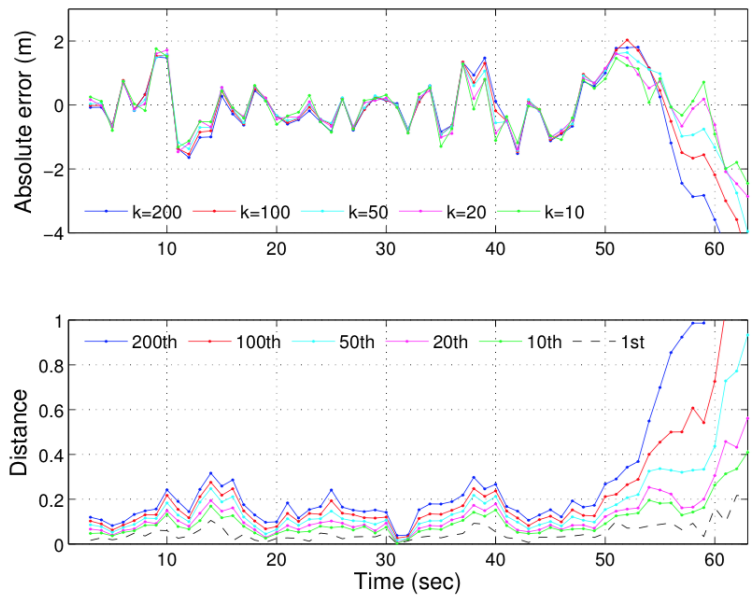
(a) Following Vehicle 422 on Lane 1



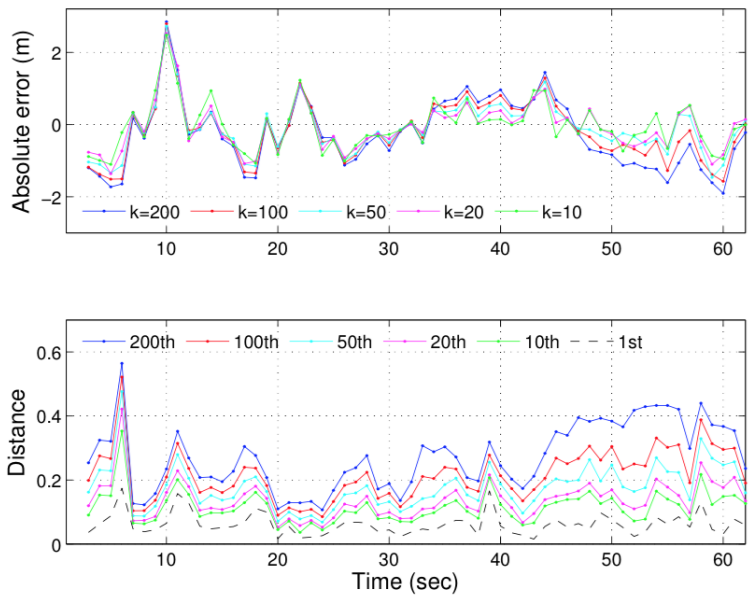
(b) Following Vehicle 1989 on Lane 1

Determination of k: First scenario

- Upper plot (absolute errors): smaller k basically makes smaller errors, but the errors may fluctuate more
  - the closer historical samples better reflect estimated case, but averaging a small number of samples results in instability
  - General tendency but not always true.



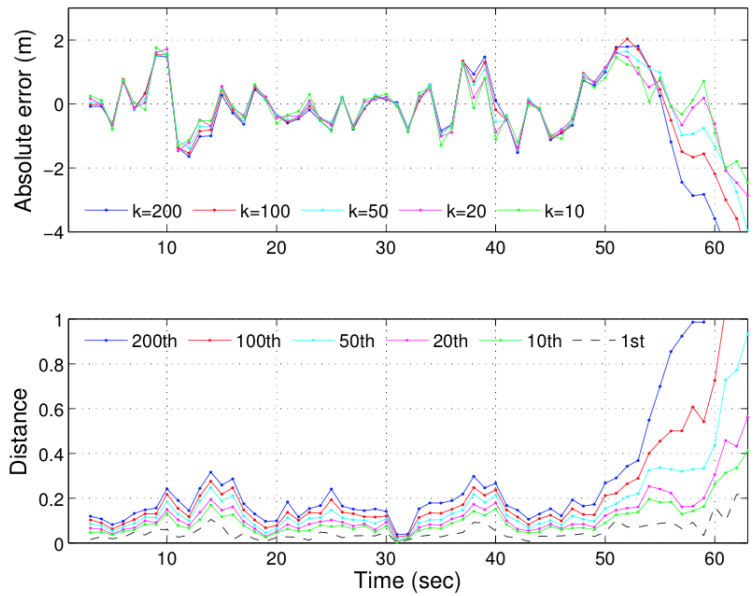
(a) Following Vehicle 422 on Lane 1



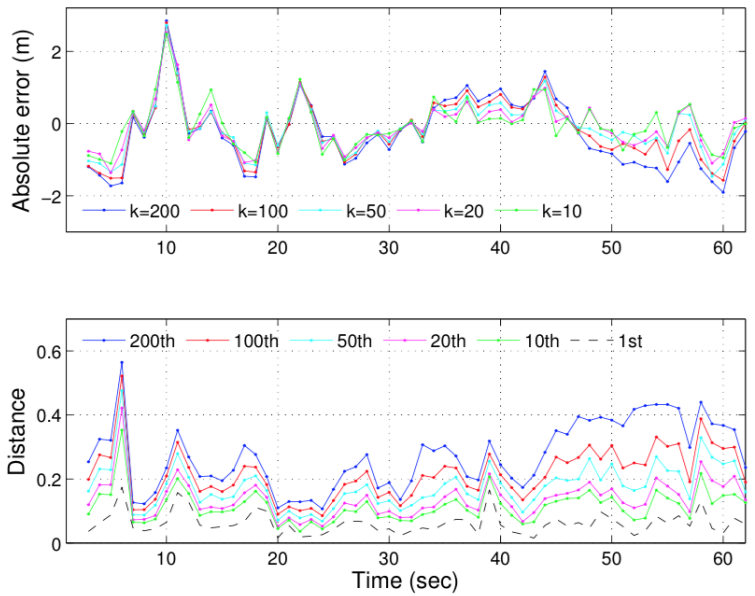
(b) Following Vehicle 1989 on Lane 1

Determination of k: First scenario

- Lower plot (distance $\mathscr{D}_k$): smaller k basically makes smaller distance (more similar)
  - For Database 3, when $k = 10$, the estimations are usually good with the distance smaller than 0.2, i.e. $\mathscr{D}_k < 0.2$



(a) Following Vehicle 422 on Lane 1

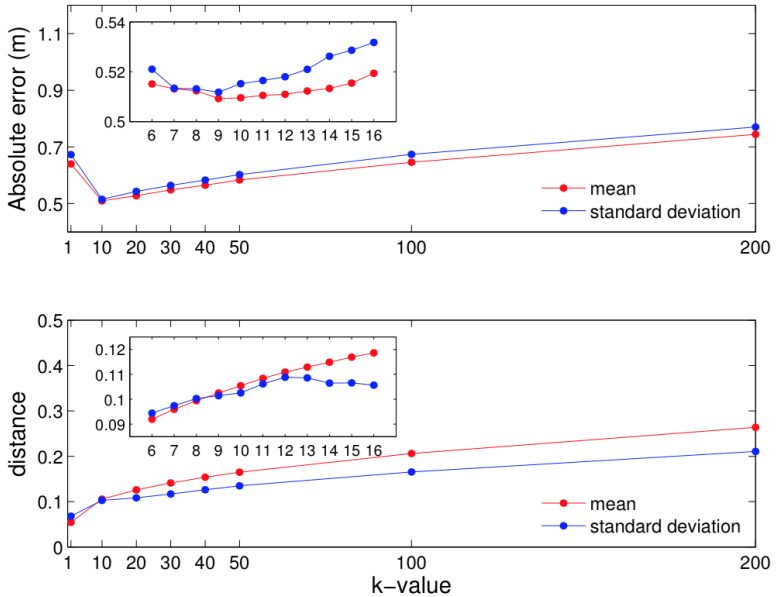


(b) Following Vehicle 1989 on Lane 1

Determination of k: Second scenario

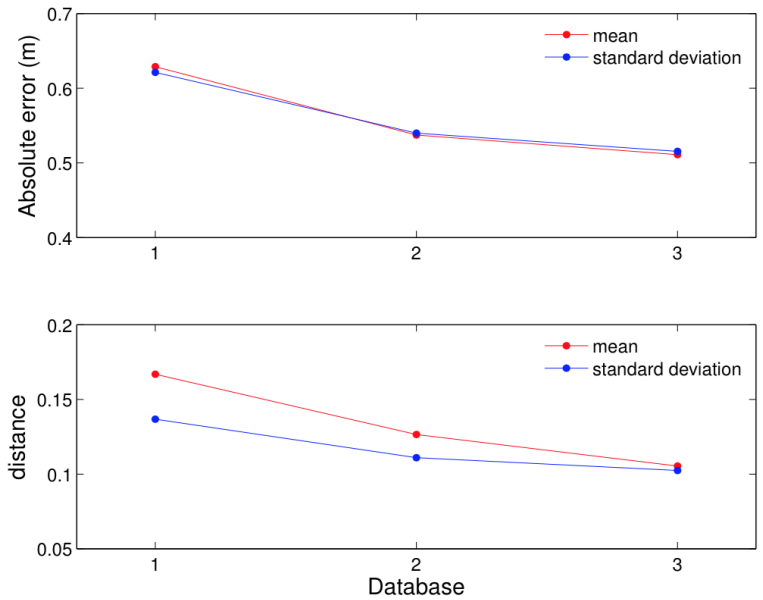
To make a more general conclusion,

- Using different k-values and Database 3, estimate randomly-selected 50 followers.
- Calculate the mean and standard deviation of absolute errors
- It can be seen:
  - Upper plot: for the dataset, the optimal k-value is about 10
  - Lower plot: usually, $\mathscr{D}_k < 0.2$



Determination of k: Second scenario

- Fixing $k = 10$, and using different databases,
- It can be seen
  - The larger Database 3 results in better estimations (i.e., lower mean and standard deviation of the absolute errors)
  - Further shortens the distance $\mathscr{D}_k$.



Determination of k: Conclusion

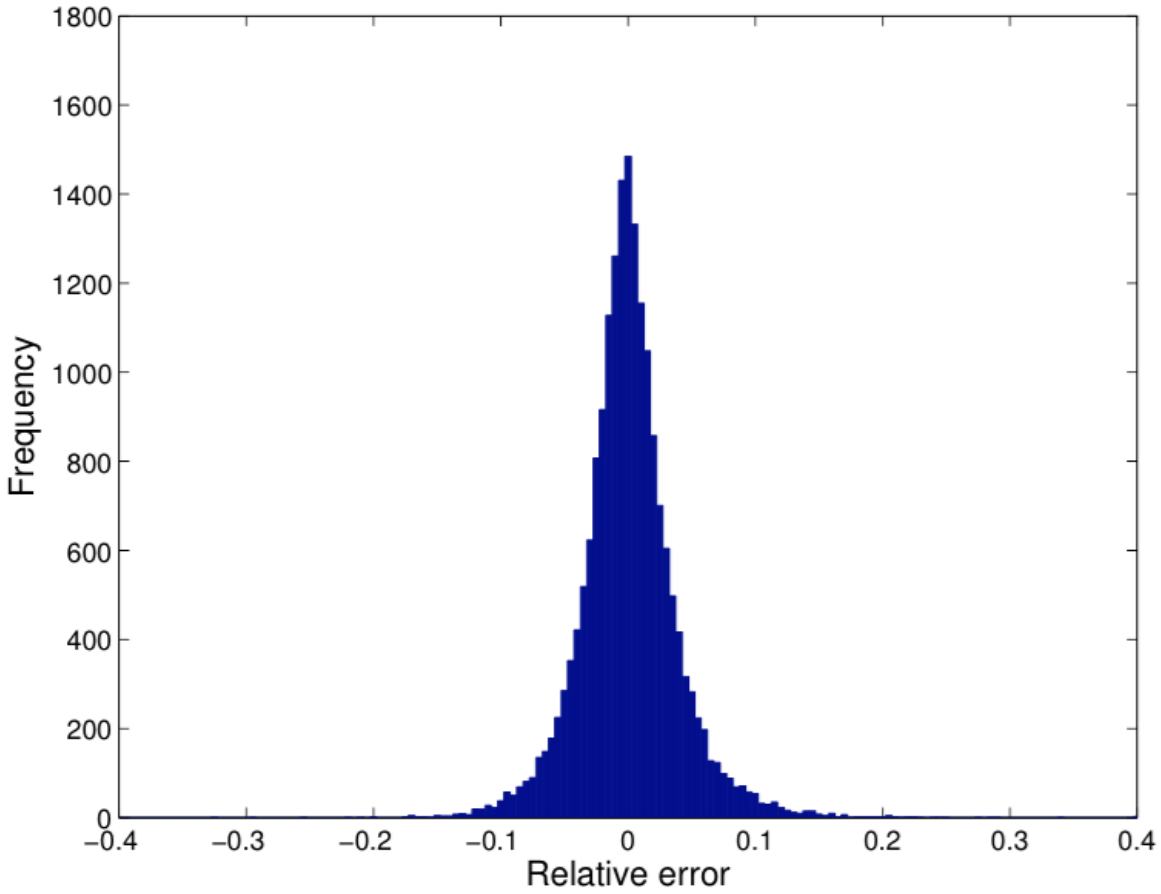
- Specify $k = 10$ for the database built on the US-101 dataset
- An estimation is considered to be satisfied if $\mathscr{D}_k < 0.2$.
- This determination is a premise to apply kNN, because the optimal k-value is highly related to the underlying database.

Outline

Analysis on avoiding collisions

- Generally speaking, averaging similar collision-free historical cases would not lead to a collision.
- However, as a data-driven method, it may be difficult to prove it mathematically.
- To show collision-free statistically, we plot the relative errors between ground-true and estimated space headway.

Analysis on avoiding collisions

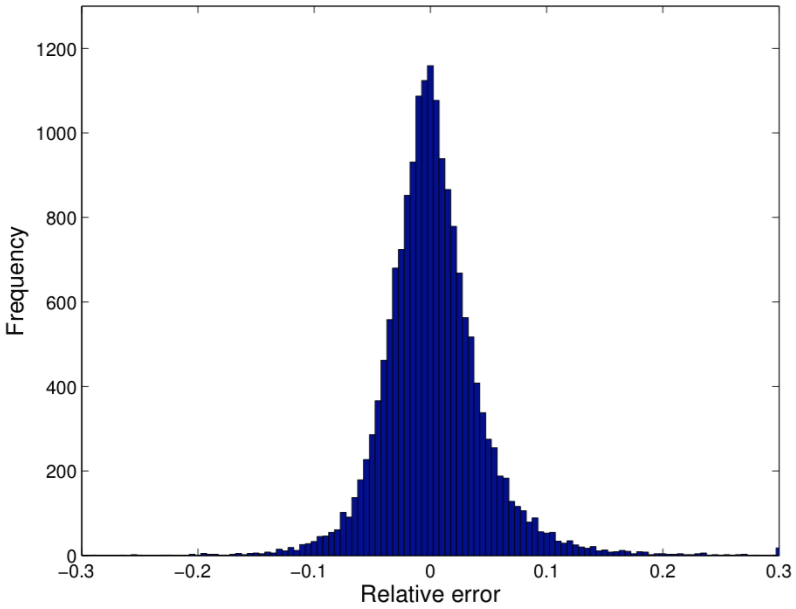


Outline

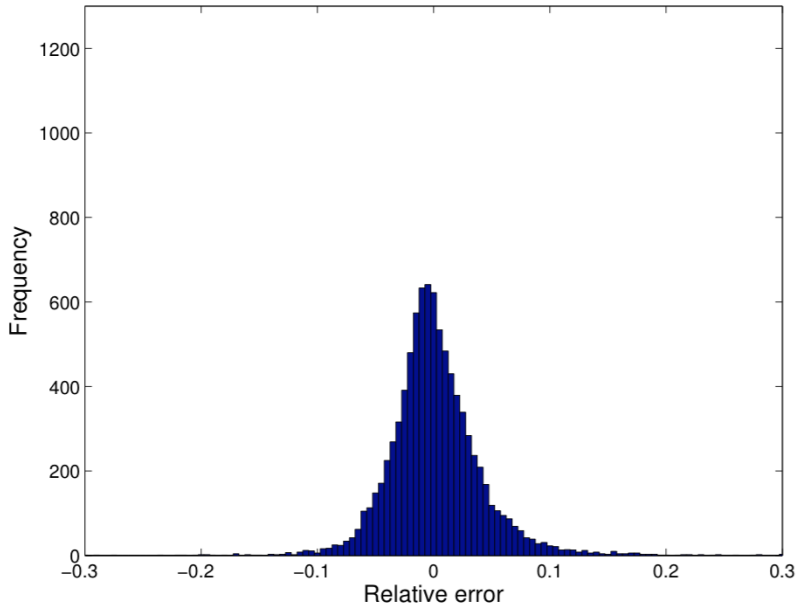
Transferability of the model and the database

- Basic assumption of kNN: drivers repeat their behavior in similar circumstances
- An database could be well transferred to any site with similar circumstances including driving habits, roadway geometry, etc.
- To show this, we estimate vehicles on I-80 using US-101 database.

Transferability of the model and the database



(a) Lane 2, I-80

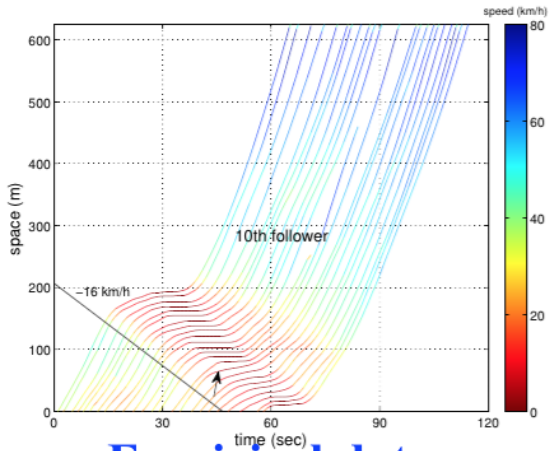


(b) Lane 3, I-80

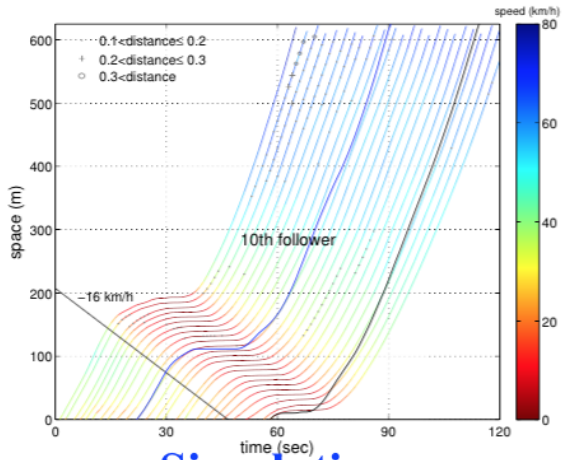
Outline

Outline

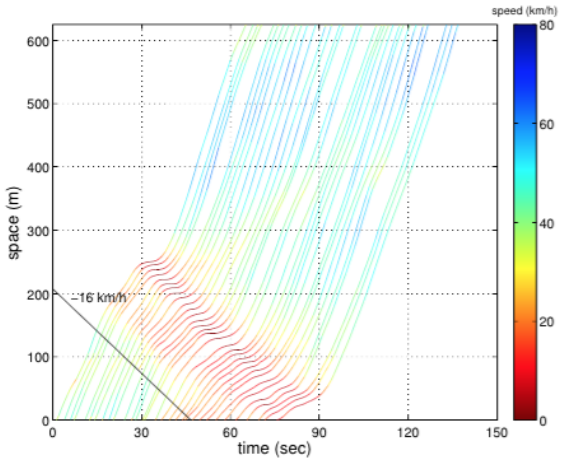
A platoon with real boundary conditions



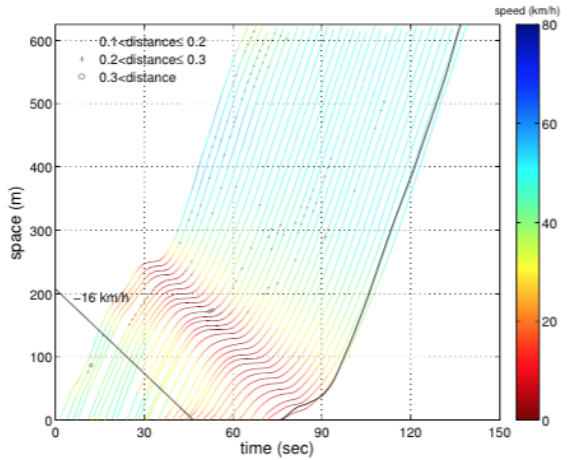
(a) Real platoon following Vehicle 422



(b) Simulated platoon following Vehicle 422

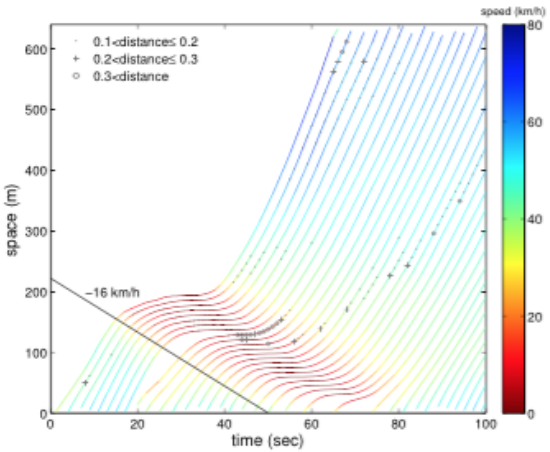


(c) Real platoon following Vehicle 1989

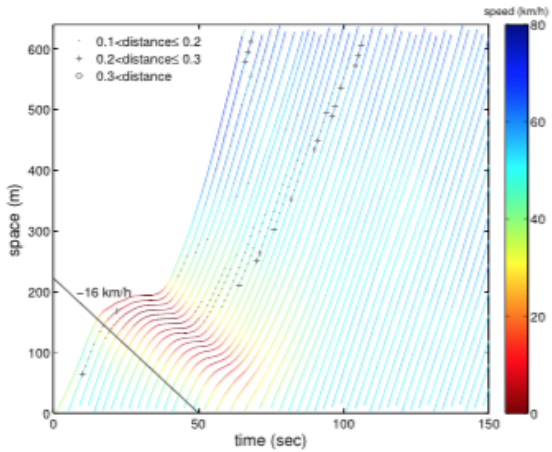


(d) Simulated platoon following Vehicle 1989

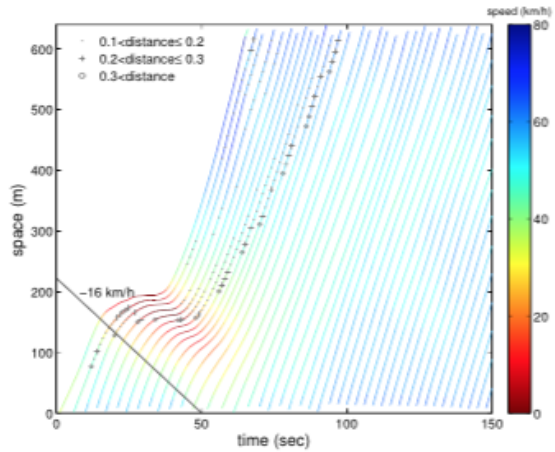
A platoon with different boundary conditions



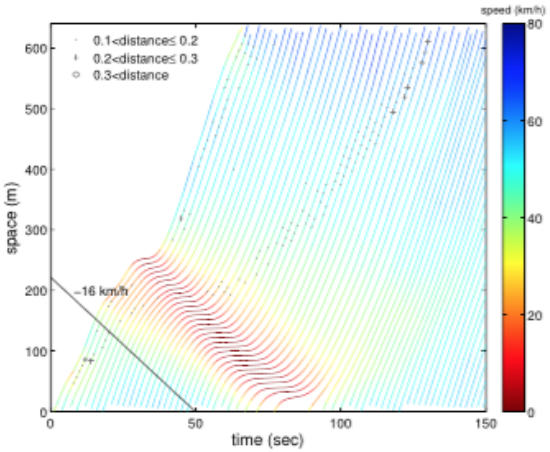
(a) Vehicle 422, entry gap 30 m



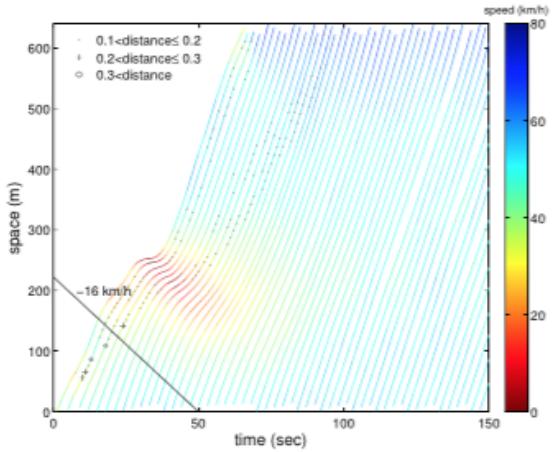
(b) Vehicle 422, entry gap 40 m



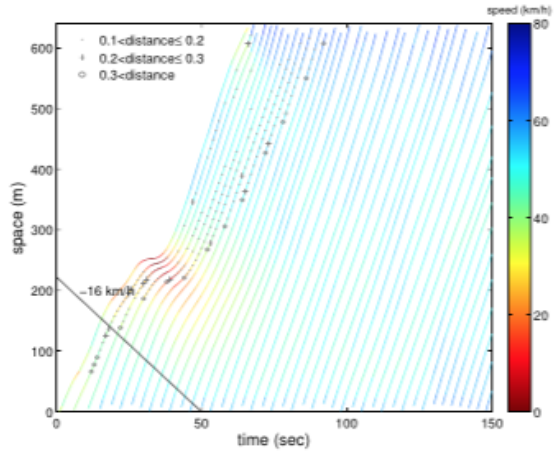
(c) Vehicle 422, entry gap 50 m



(d) Vehicle 1080, entry gap 30 m

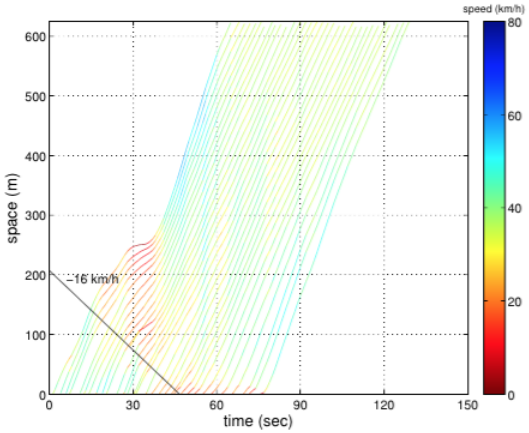


(e) Vehicle 1080, entry gap 40 m

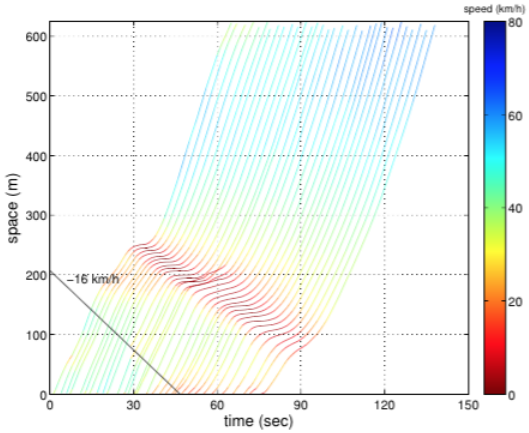


(f) Vehicle 1080, entry gap 50 m

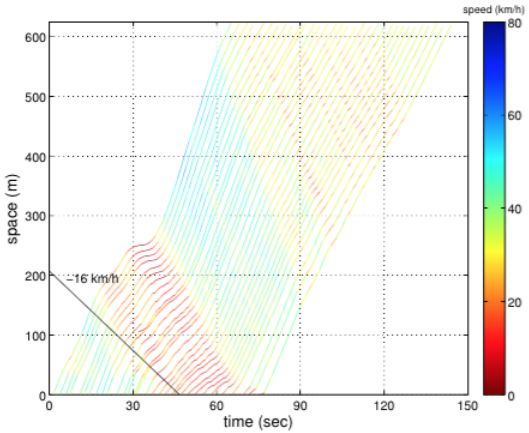
Necessity of each input



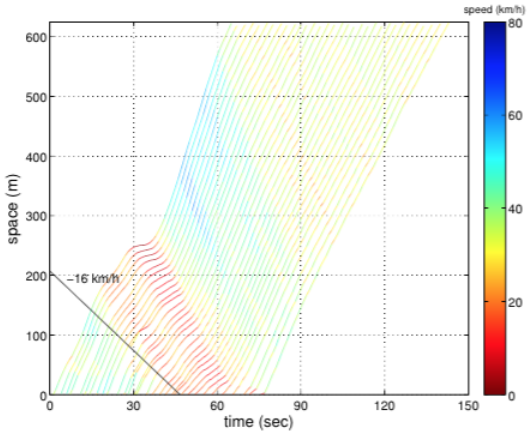
(a) Inputs without $d_{n-1}(t-\tau)$



(b) Inputs without $d_{n-1}(t)$



(c) Inputs without $s_n(t-\tau)$



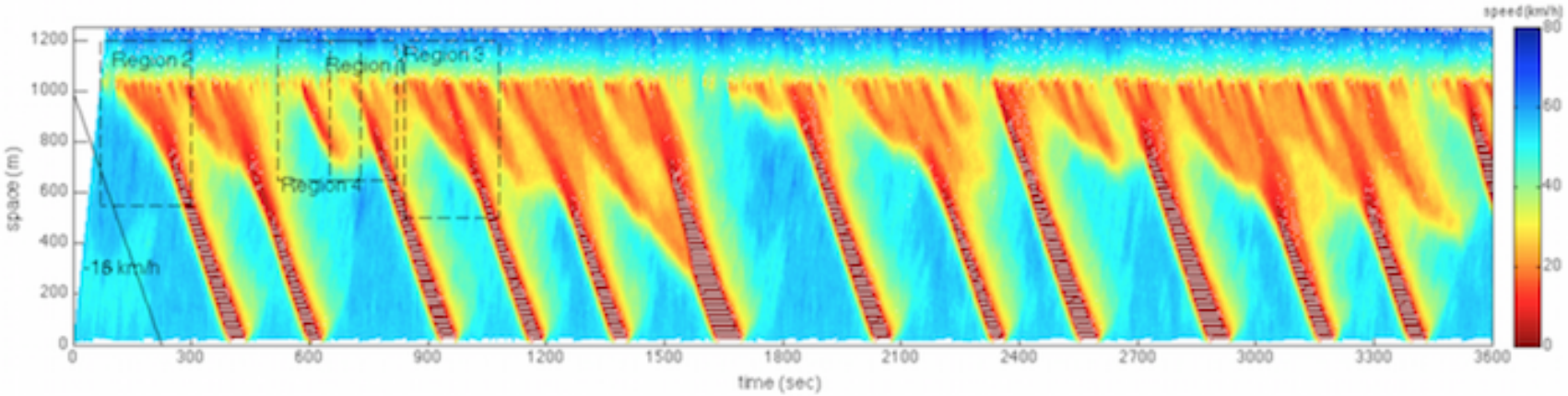
(d) Inputs without $s_n(t)$

Outline

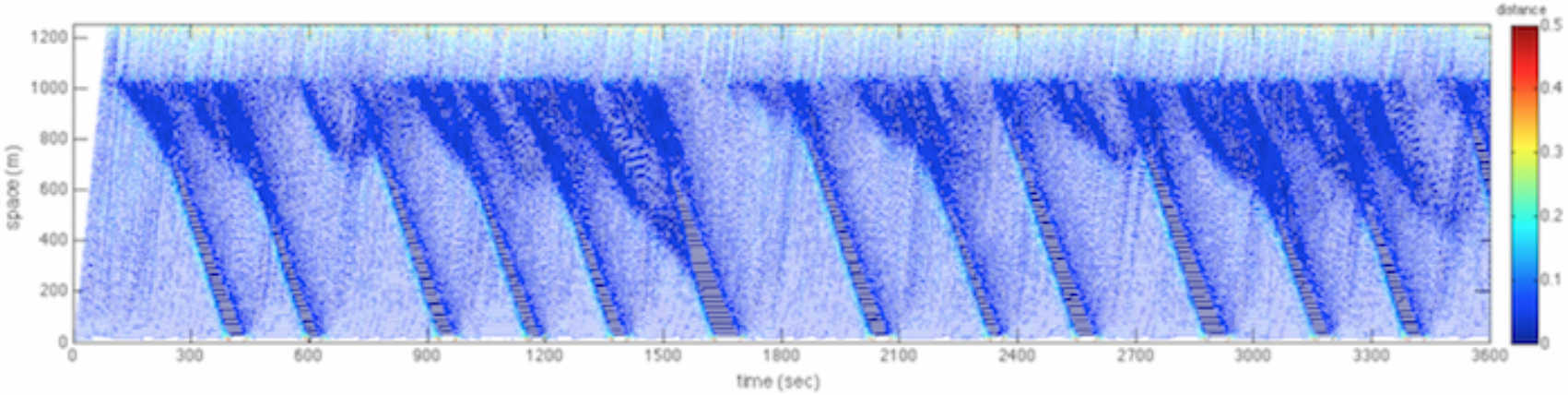
Constructing a rubbernecking scenario

- A 1.25 km one-lane roadway is simulated for 1 hour;
- New vehicle enters with an initial speed of 54 km h^{-1}, when its leader has left the entrance 30 m away;
- The rubbernecking zone is located at section [1, 1.05] km.
- A probability r to rubberneck and then slow down by a percentage of $(1 - p)$.
- If rubbernecking occurs, it will occur at most once.
- The database contains all data collected from Lane 1, 2, and 3 during all 45 minutes.
- No assumption or calibration

Simulation results: Time-space plots

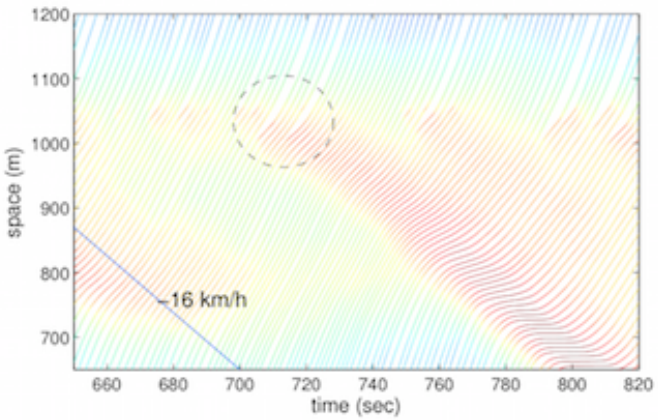


(a) Time-space diagram coloured by speed

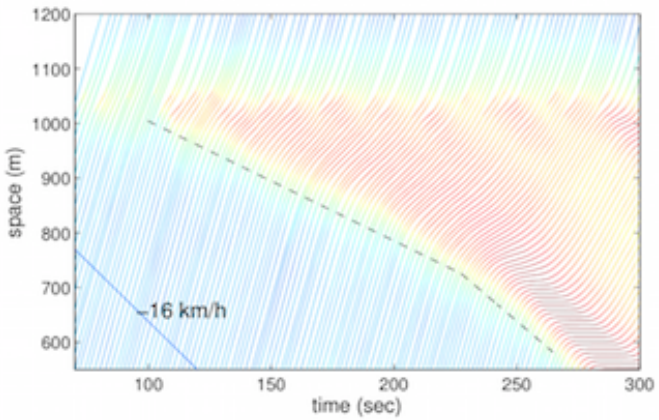


(b) Time-space diagram coloured by distance

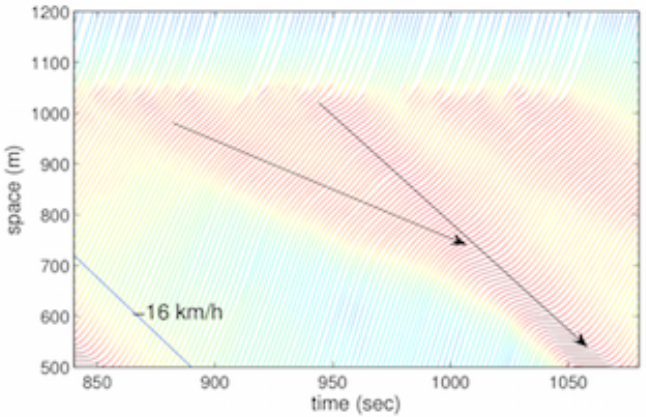
Simulation results: Time-space plots



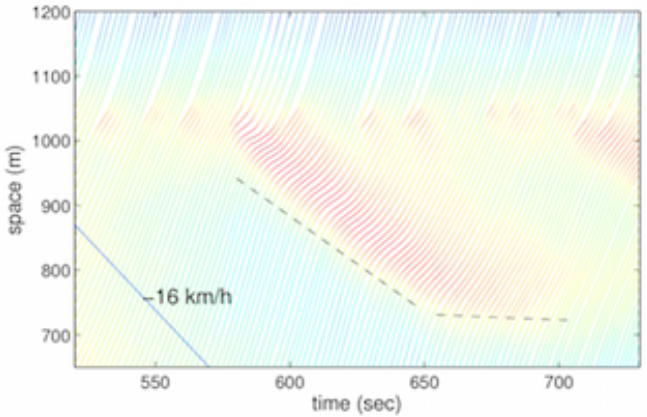
(c) Region 1



(d) Region 2



(e) Region 3



(f) Region 4

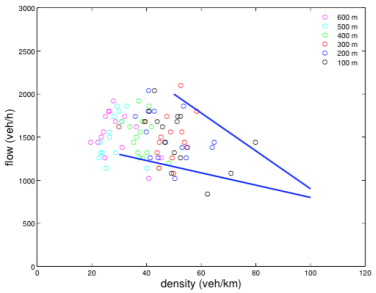
Simulation results: Fundamental diagrams

- Suppose that virtual detectors are installed in the roadside, and the traffic flow q, density ρ, and speed v within a time period T are measured as standard models.

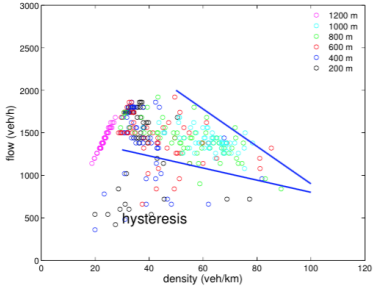
$$q = \frac{N}{T}, \quad \rho = \frac{\sum_{n=1}^{N} \frac{1}{v_n}}{T}, \text{ and } v = \frac{q}{\rho} = \frac{N}{\sum_{n=1}^{N} \frac{1}{v_n}} \tag{6}$$

where N is the count of the vehicles passing the detection location within the time period T, and v_n is the passing speed of a detected vehicle.

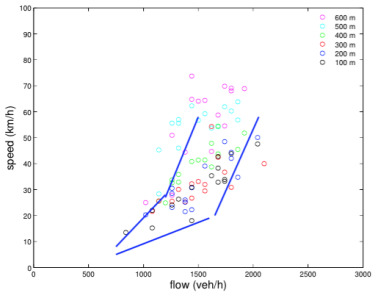
Simulation results: Fundamental diagrams



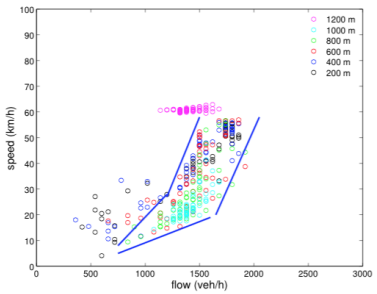
(a) flow-density diagram for empirical traffic



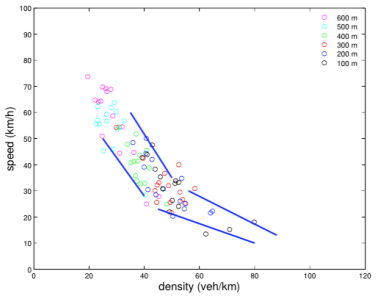
(b) flow-density diagram for simulated traffic



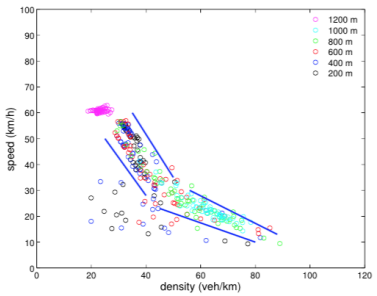
(c) speed-flow diagram for empirical traffic



(d) speed-flow diagram for simulated traffic

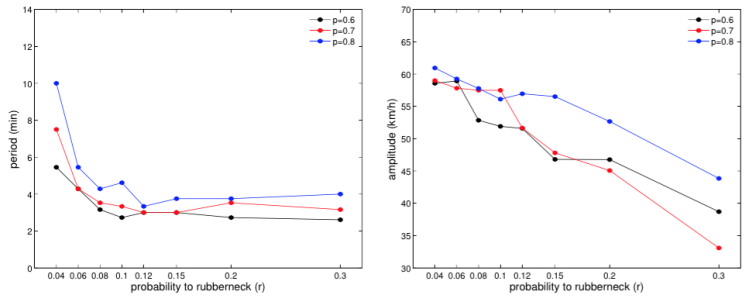


(e) speed-density diagram for empirical traffic



(f) speed-density diagram for simulated traffic

Simulation results: Periods and amplitudes



(a) Period (h=4) (b) Amplitude (h=4)

(c) Period (h=5) (d) Amplitude (h=5)

(e) Period (h=6) (f) Amplitude (h=6)

Outline

Adding white Gaussian noise

- We model the driver errors in a form of a white Gaussian noise with diffusion coefficient σ^2.

$$\tilde{d}_n(t+\tau) = d_n(t+\tau) + W(\varepsilon) \tag{7}$$

where

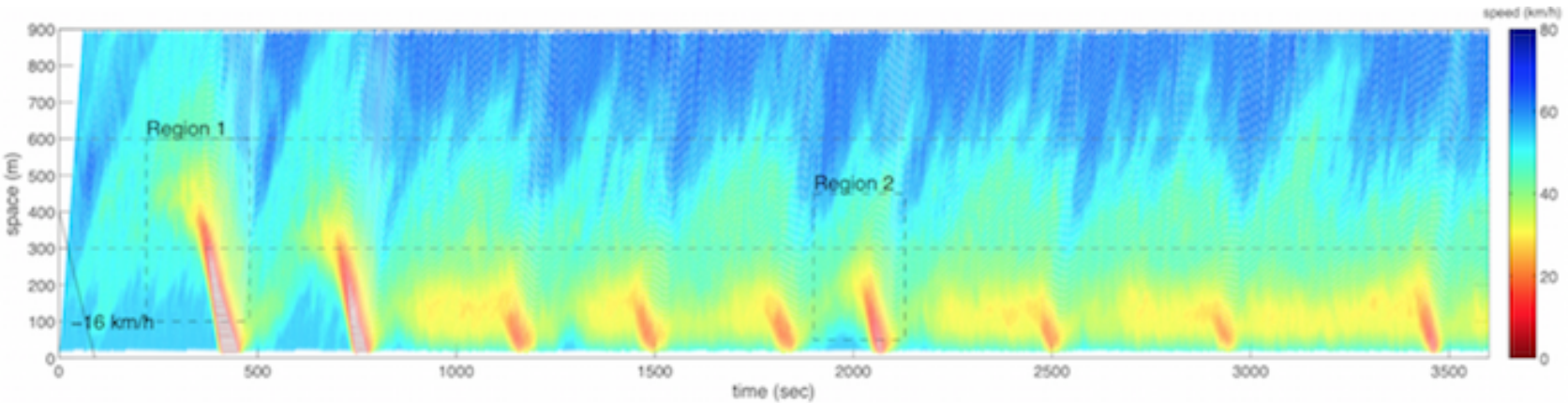
$$W(\varepsilon) = \begin{cases} \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{\varepsilon^2}{2\sigma^2}} & , \quad d_{\text{jam}} < d_n(t+\tau) < d_{\text{free}} \\ 0 & , \quad \text{otherwise} \end{cases} \tag{8}$$

where d_{jam} and d_{free} are the moving distance/speed around jam density and free-flow conditions, respectively.

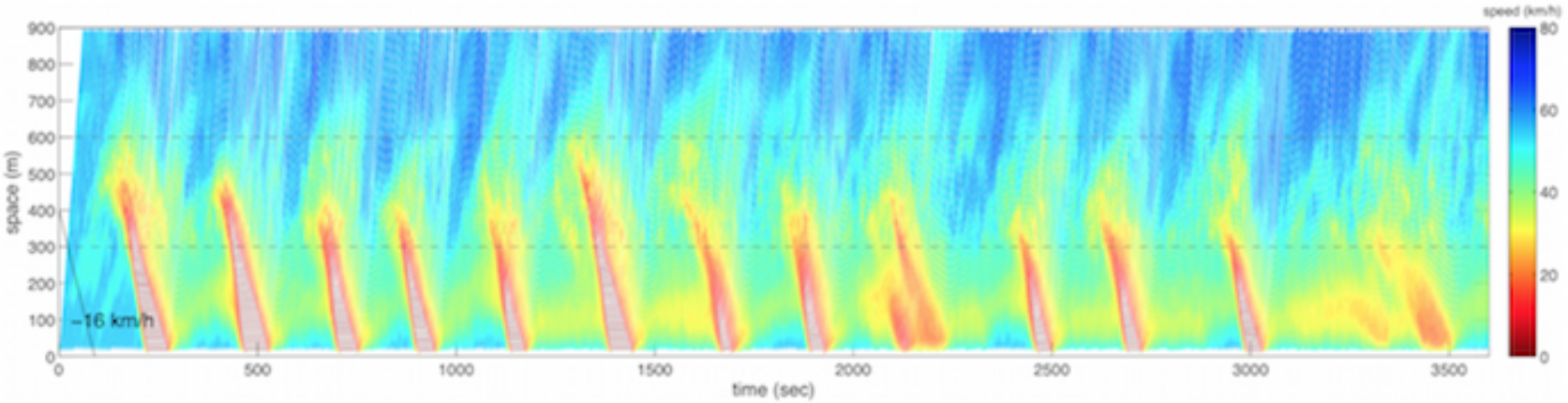
Constructing a driver-error scenario

- A 900 m one-lane roadway is simulated for 1 h.
- The white noise is only added when a vehicle is moving in the section between 300 m and 600 m. This is analogous to an uphill section.
- It is set that d_{free}=54 km h^{-1} and d_{jam}=15 km h^{-1}
- The entry speed and gap are 54 km h^{-1} and 20 m, respectively.

Simulation results: Time-space plots

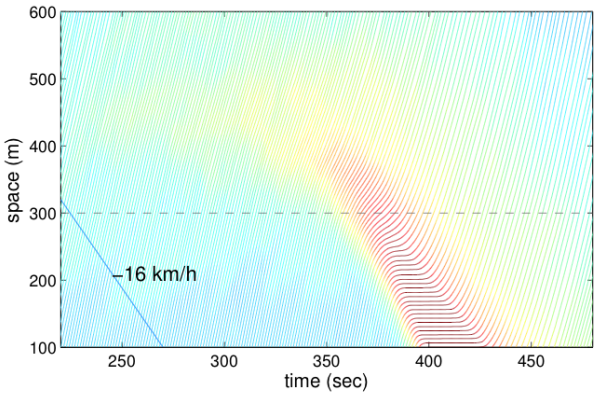


(a) Time-space diagram of trajectories ($\sigma = 0.2$)

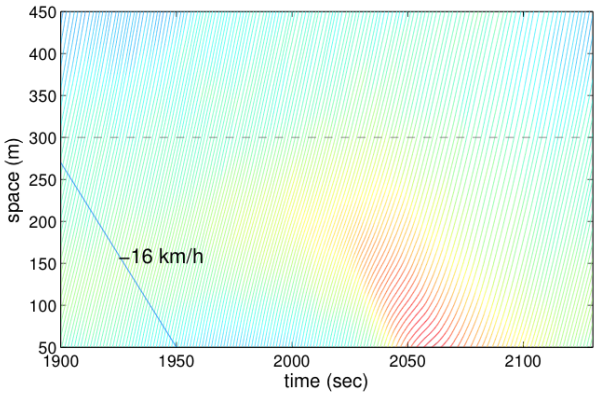


(b) Time-space diagram of trajectories ($\sigma = 0.5$)

Simulation results: Time-space plots



(c) Region 1



(d) Region 2

Conclusion

- Neither mathematical equation nor calibration is needed to be concerned in the model;
- Neither the fundamental diagrams nor driver's behaviour parameters is assumed;
- The model is simple and parsimonious particularly in the conceptual point of view, and the only parameter is k;
- All inputs and outputs are based on vehicle positions, which are straightforward to reproduce traffic dynamics in computer simulations;
- The model is able to well reproduce traffic characteristics contained by the underlying database, such as all stages of stop-and-go oscillations, fundamental diagrams, periods and amplitudes of oscillations.

Thank you!