

A Maneuver-Prediction Method Based on Dynamic Bayesian Network in Highway Scenarios

Junxiang Li¹, Xiaohui Li¹, Bohan Jiang¹, Qi Zhu¹

1. College of Mechatronic Engineering and Automation,
National University of Defense Technology, Changsha, Hunan, 410073, P.R.China
E-mail: junxiangli90@gmail.com;xiaohuilee@outlook.com;
jiangbohan15@nudt.edu.cn;zhuqics@nudt.edu.cn

Abstract: The accurate maneuver prediction for dynamic vehicles can enhance driving safety in complex environments. This paper presents a maneuver prediction method for dynamic vehicles in highway scenarios. The method effectively combines multi-frame vehicle states, road structures and interactions among vehicles. With a novel extraction algorithm of environment feature, the method infers the probability of each driving maneuver by using a Dynamic Bayesian Network. The experimental results demonstrate that our method can predict lane-change maneuvers at least 2 seconds before they occur in real environments with an accuracy of 84.9%.

Key Words: Autonomous driving, maneuver prediction, feature extraction, Dynamic Bayesian Network.

1 INTRODUCTION

Many traffic accidents happen in complex traffic environments. The accurate motion prediction for surrounding vehicles can enhance driving safety in these complex environments. As to autonomous driving, the accurate motion prediction for surrounding vehicles, which are also called as participant vehicles (PVs), will contribute to make safer decisions and plan smoother trajectories. Motion prediction in autonomous driving provides a human-like way of driving that can predict maneuvers of surrounding vehicles. Similarly, accurate motion prediction for PVs in Advanced Driver Assistance Systems (ADAS) is also useful for human drivers.

In recent years, a large number of new techniques have been proposed to predict the motion of participant vehicles. In [1], the approaches of motion prediction are classified into three categories, namely physics-based approaches, maneuver-based approaches and interaction-aware approaches. However, many researches always combine interaction-aware approaches with the other two approaches for more accurate motion prediction results [2]. Moreover, the motion prediction approaches are classified into four categories in [3] according to the prediction number of future behaviors. Different from these two classification methods, we classify the motion prediction approaches into two categories: 1) trajectory prediction and 2) maneuver prediction. Trajectory prediction deals with the prediction in the short term, whereas the maneuver prediction predicts the motion in the long term.

Trajectory prediction predicts the trajectories of PVs. Basically, these approaches use current states of PVs and the kinematic model of vehicles. For instance, continu-

ous Bayes filters (e.g., Kalman Filter) are applied in [4, 5]. Besides, probabilistic methods [2] are also used in trajectory prediction to cope with uncertainty and inaccuracy. In [6, 7], an offline learning trajectory set learned by Gaussian process and an improved RRT are combined to predict trajectories. However, driving trajectories are related to long-term maneuvers, mere trajectory prediction is not accurate in long term.

Maneuver prediction gradually receives more attention in recent years. The maneuver prediction is usually more reliable in the long term. For instance, the approach in [8] can detect a lane-change maneuver averagely 0.6 s earlier than an adaptive cruise control (ACC) system. An object-oriented Bayesian network (OOBN) is proposed to predict maneuvers of PVs in [8]. The naive Bayesian method is extended in [9] for lane-change maneuver prediction. Sivaraman and Trivedi [10] introduce a dynamic probability drivability map (DPDM) for a driver assistant system, which can predict maneuvers of other vehicles. In [11], a method that combines the field map and a Bayesian network is presented, focusing on the interaction-aware maneuver prediction. Recently, a Support Vector Machine (SVM) method is used to predict lane-change maneuvers in [12] with consideration of traffic interactions.

In order to consider interactions among vehicles, Dynamic Bayesian Networks (DBNs) are used in the maneuver prediction. Inspired by [13], this paper presents a novel Dynamic Bayesian Network for the maneuver prediction (DBMP) in highway scenarios. The advantages of our DBMP include two aspects. 1) DBMP predicts maneuvers of PVs based on multi-frame features, the road structure, and interactions among PVs together in a unified framework. 2) DBMP combines the driving experience of humans with machine learning algorithm. Note that although DBMP is focused on the highway scenarios, it can be ex-

This work is supported by National Natural Science Foundation (NNSF) of China under Grant No.61790565.

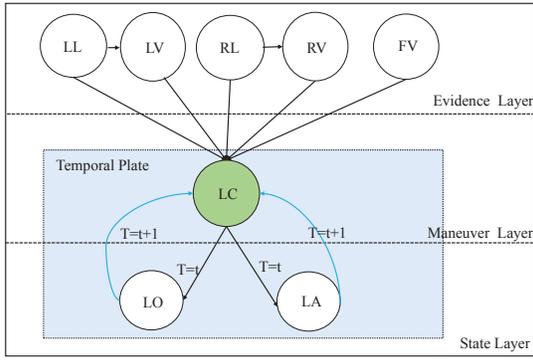


Figure 1: The Dynamic Bayesian Network for maneuver prediction.

tended to other complex environments.

The rest of the paper is organized as follows. In Sec. 2, the model structure of DBMP and the inference of maneuver probabilities are discussed. In Sec. 3, proposed algorithms for feature extraction are introduced, which are the foundation of the inference. Sec. 4 presents experimental results. Finally, Sec. 5 draws conclusions and makes suggestions for future work.

2 MANEUVER PREDICTION USING DYNAMIC BAYESIAN NETWORK

Dynamic traffic environments can provide many clues to predict maneuvers of PVs. These clues, which include the road structure, detectable vehicle states, and traffic interaction, are detected continuously or displayed as multi-frame features. In order to effectively deal with the continuous prediction [14], we developed our maneuver prediction method based on a DBN.

2.1 Model Structure

The structure of our DBN is shown in Fig. 1. Each node represents a detectable clue, which is also called as the feature. Values of nodes can be obtained by feature extraction algorithms, which will be introduced in Sec. 3. The acyclic arrow represents the causal relationship between two features. According to the causal relationship between the lane-change maneuver and other features, we separate all nodes into three layers: 1) an evidence layer; 2) a maneuver layer; 3) a state layer.

1) In the evidence layer, the environment information is considered. The features include: i) the existence of a left lane (LL); ii) the existence of a right lane (RL); iii) the existence of a neighboring vehicle in the left lane (LV); iv) the existence of a neighboring vehicle in the right lane (RV); v) the existence of a neighboring lead vehicle in the same lane (FV). The neighboring vehicle is judged by a distance threshold to the predicting PV. The states (also called as attributes) of the features in this layer are existence or not.

2) In the maneuver layer, the feature only involves the lane change (LC). The final predicted maneuver is calculated by choosing the maximal state value. The states contain lane

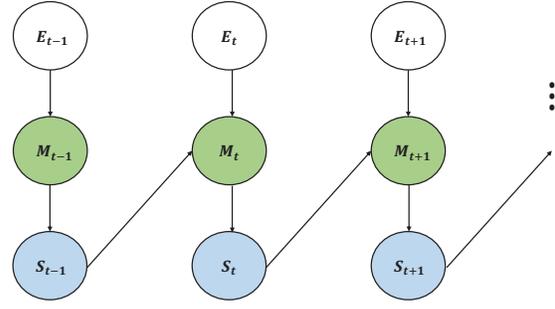


Figure 2: Block diagram of DBMP.

keeping, changing to the left lane, and changing to the right lane.

3) In the state layer, two features are involved: i) the deviation from the centerline of the driving lane (LO); ii) the acceleration (LA). As the states of a PV in the time step t will influence the maneuver in the time step $t + 1$, there is a recursive formulation in the temporal plate to display the dynamic causal relationship. The states of the features in this layer are continuous and discretized into two reasonable values.

2.2 Inference

The final results of predicted maneuvers are inferred by (1). For simplicity, our DBMP can be represented as Fig. 2. Each node represents one layer of the DBN in a certain one-time step. For instance, E_t represents five nodes (i.e., LL, RL, LV, RV, FV) in the time step t .

$$\mathcal{P}(M(t+1)|E(t+1), S(t)) = \frac{\mathcal{P}(M(t+1), E(t+1), S(t))}{\sum_M \mathcal{P}(M(t+1), E(t+1), S(t))} \quad (1)$$

The marginal distribution is expressed as (2) because of the conditional independence.

$$\begin{aligned} \mathcal{P}(M, E, S) &= \mathcal{P}(LL)\mathcal{P}(RL)\mathcal{P}(FV) \\ &\mathcal{P}(LV|LL)\mathcal{P}(RV|RL) \\ &\mathcal{P}(LC|LL, RL, LV, RV, FV) \\ &\mathcal{P}(LO|LC)\mathcal{P}(LA|LC) \end{aligned} \quad (2)$$

Although the prior probability can be learned by learning algorithm, we specify the prior probability by expert experience [15]. The original number of prior probability that we need to specify is 383, which is calculated by:

$$2^7 * 3 - 1 = 383$$

Thanks to the conditional independence, the setup number of prior probability decreases to 77, which is calculated by:

$$1 + 1 + 1 + 2 + 2 + 2^6 + 3 + 3 = 77$$

Note that if $S(t)$ is unknown for the inference in the time step t , the prior distribution is $\mathcal{P}(M(t+1)|E(t+1))$. After

$S(t)$ determined, the prior distribution is $\mathcal{P}(M(t+1)|E(t+1), S(t))$. These two distributions comply to (3).

$$\begin{aligned} & \mathcal{P}(M(t+1)|E(t+1)) \\ &= \sum_{S(t)} \mathcal{P}(M(t+1)|E(t+1), S(t))\mathcal{P}(S(t)) \\ &< \sum_{S(t)} \mathcal{P}(M(t+1)|E(t+1), S(t)) \end{aligned} \quad (3)$$

3 FEATURE EXTRACTION

This section introduces our algorithms for feature extraction in real environments. The purpose of feature extraction is to obtain states of nodes, which are used to infer the probabilities of the maneuver. Thus feature extraction is the foundation of the inference which is discussed in Sec. 2.2. Note that features of all PVs are extracted iteratively.

Firstly, we classify all other features except the feature of the longitudinal acceleration (LA) into two types based on the different relationship, which are features of point-curve relationship and features of point-point relationship. The relationships are shown in Fig. 3. The state of LA can be calculated by the filtering algorithm after detecting continuous velocities of PVs.

Secondly, features of point-curve relationship are extracted, which contain the existence of a left lane (LL), the existence of a right lane (RL), and the lateral deviation from the centerline of the driving lane (LO). The extraction algorithm is listed in Algorithm 1.

Step 1. All lane points, which are detected by sensing systems, are classified and approximated as traffic lines (lines or second-degree polynomial [16]).

Step 2. For each traffic line, the nearest point to the center point P_A of the PV is calculated. Choosing two neighbor points P_{N-1} and P_{N+1} on the traffic line by ΔP . k_1, k_2 are vectors from P_A to P_{N-1} and P_{N+1} respectively.

Step 3. The relative positional relationship (i.e., P_A is on the left side or right side of the traffic line) is calculated by cross product of two vectors k_1, k_2 , with regard to the driving direction.

Step 4. Judge the relative positional relationship of all traffic lines and the PV. The state of LL and RL can be gained by the number of the traffic line on each side. Moreover, we can also ascertain all lane indexes if the number of lanes discussed is changeless.

Step 5. Similar to Step 2, d_{Lmin} is gained by calculating the distance from the centerline of the driving lane to P_A .

Step 6. The state of LO can be gained after discretizing d_{Lmin} into reasonable values.

Thirdly, features of point-point relationship are extracted, which contain the existence of a neighboring vehicle in the left lane (LV), the existence of a neighboring vehicle in the right lane (RV), the existence of a neighboring lead vehicle in the same lane (FV). The extraction algorithm is listed in Algorithm 2.

Step 1. Before iteration, all PVs are stored according to the lane index in an index table, which is shown in Fig. 3. The index table is designed for quicker calculation in the next step.

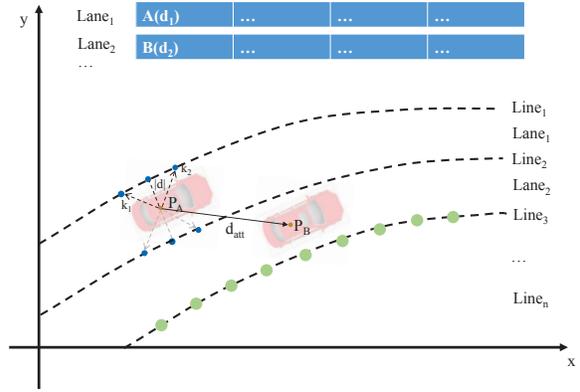


Figure 3: The point-curve relationship and point-point relationship.

Algorithm 1 Extraction Algorithm for features of point-curve relationship

Require: P_A (center point of focused participant vehicle), S_L (lane points)

Ensure: LL, RL, LO (states of LL, RL, LO)

- 1: $Lines \leftarrow ApproximateCurve(S_L)$
- 2: **for** all l in $Lines$ **do**
- 3: $(k_1, k_2) \leftarrow Calculate(l, P_A)$
- 4: **if** $k_1 \times k_2 < (0, 0, 0)$ **then**
- 5: $N_{LL} \leftarrow N_{LL} + 1$
- 6: **else**
- 7: $N_{RL} \leftarrow N_{RL} + 1$
- 8: **end if**
- 9: **end for**
- 10: $N_{LL} > 2 : LL = 1?LL = 0$
- 11: $N_{RL} > 2 : RL = 1?RL = 0$
- 12: $I_A \leftarrow IndexforPV(N_{LL}, N_{RL})$
- 13: $d_{Lmin} \leftarrow Mindist(I_A, Lines)$
- 14: $LO \leftarrow Normalize(d_{Lmin})$

Step 2. As to the states of LV and RV, we make sure whether any vehicles exist in the neighbor lane by the index table. Then, the states of LV and RV are gained if a vehicle is existed and closer to the predicting PV than a predefined distance threshold d_{thre} .

Step 3. Similarly, the state of FV is also extracted by the condition if neighboring vehicles exist in the same lane. In this step, we consider the driving direction.

Finally, the lane-change feature (LC) is inferred after all other features are determined.

4 EXPERIMENT

4.1 Setup

To verify the effectiveness of our proposed method, we conduct two experiments, which are a qualitative experiment and a quantitative experiment. The qualitative experiment is conducted to verify prediction capacity. The quantitative experiment is to compare our DBMP with a naive Bayesian network, which does not consider the time-related lateral deviation and acceleration.

The experiment dataset used is NGSIM-I80 from Ameri-

Algorithm 2 Extraction Algorithm for features of point-point relationship

Require: S_{PV} (lane indexes of all PVs), P_A (center point of focused participant vehicle), d_{thre} (predefined distance threshold)

Ensure: LV, RV, FV (states of LV, RV, FV)

- 1: **for** all P_I in Vehicles **do**
- 2: $T_I \leftarrow LaneIndexTable(S_{PV})$
- 3: **end for**
- 4: **if** $P_B \leftarrow CarInLeftLane(T_I)$ **then**
- 5: $d(P_A, P_B) < d_{thre} : LV = 1?LV = 0$
- 6: **else**
- 7: $LV = 0$
- 8: **end if**
- 9: **if** $P_B \leftarrow CarInRightLane(T_I)$ **then**
- 10: $d(P_A, P_B) < d_{thre} : RV = 1?RV = 0$
- 11: **else**
- 12: $RV = 0$
- 13: **end if**
- 14: **if** $P_B \leftarrow CarInSameLane(T_I)$ **then**
- 15: $d(P_A, P_B) < d_{thre} : FV = 1?FV = 0$
- 16: **else**
- 17: $FV = 0$
- 18: **end if**

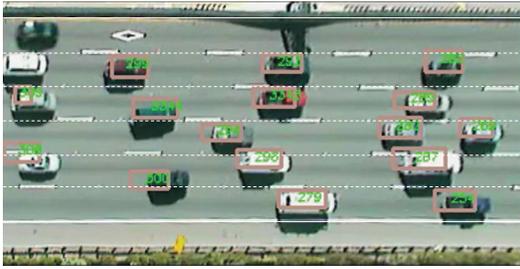


Figure 4: Screenshot in Dataset I-80.

can Federal Highway Administration [17]. NGSIM-I80 is collected from the northbound traffic on I80 in California and used for intelligent transportation systems and validation of prediction algorithms ([12], [18]). Our experimental platform is a PC (Intel(R) Core(TM) i5@2.20GHz, 4.0GB memory). The prediction program has been implemented in C++. The preparation of the dataset contains two steps. Firstly, data of all lane-change vehicles are selected. Secondly, all ground-truth maneuvers are extracted by the relative distance to the centerline of the driving lane. For simplicity, we classify maneuvers in the dataset into two states, which are lane keeping and lane change.

Table 1: Parameters of experiment

Parameter Name	Meaning	Value
d_c	deviation distance from centerline for lane change	0.2m
d_{thre}	threshold distance for filed of view	3m

4.2 Qualitative Results

Maneuvers of all lane-change vehicles are predicted using our DBMP. Parts of qualitative results (i.e., 4 typical pre-

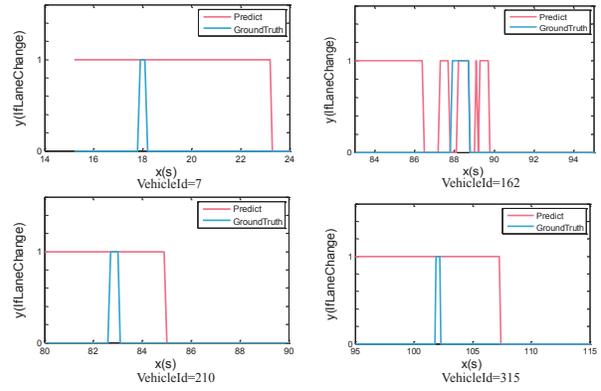


Figure 5: Qualitative Results for Prediction.

dicted results) are shown in Fig. 5. The red curves represent the predicted results by DBMP. The blue curves represent the ground-truth maneuvers extracted from the dataset. When the value of y axis equals to zero, the vehicle is keeping the lane. While the value of y axis equals to one, the vehicle is executing a lane change.

The results show that a lane change maneuver can be predicted by DBMP at least 2 s earlier. However, the shortcoming of our method is the prediction delay (about 3 s) after the lane change. The reason is partly because parameter d_c chosen is small. Moreover, the unstable prediction is also found in the vehicle 162 (i.e., the upper right figure).

4.3 Quantitative Results

The performance of our method is compared to that of naive Bayesian Network (BN) based on correctness in predicting maneuvers. Five vehicles, which totally contains 3012 data in dataset are chosen for prediction. These vehicles execute lane-change maneuvers during the driving process. Besides accuracy measure (4), average precision P , recall R (5) and F_1 score (6) are also used as our performance criteria. Because there are more lane-keeping data than lane-change data in the dataset, we make true positive case TP when it predicts a lane-keeping maneuver correctly. Similarly, TN, FP, FN represent true negative case, false positive case and false negative case respectively.

$$Accuracy = \frac{TP + TN}{TP + TN + FN + FP} \quad (4)$$

$$P = \frac{TP}{TP + FP}, R = \frac{TP}{TP + FN} \quad (5)$$

$$F_1 = 2 \cdot \frac{P \cdot R}{P + R} \quad (6)$$

The results in Table 2 show a significant, 37.1% increase in precision, 0.9% increase in recall, 29.2% increase in F_1 by using DBMP than Naive BN. The prediction accuracy by using DBMP reaches to 84.9%. The performance improves because vehicle states and traffic interaction are used as multi-frame features in DBMP.

Table 2: Comparison of naive BN and DBMP

Approach	Accuracy[%]	Precision[%]	Recall[%]	F_1 [%]
Naive BN	49.1	50.0	96.3	62.6
DBMP	84.9	87.1	97.2	91.8

5 CONCLUSIONS

We have presented a maneuver predictor DBMP for highway driving scenarios. The algorithm is based on the Dynamic Bayesian Network, which combines structural prediction and dynamic features. The method has been validated using open dataset I-80. The results demonstrate that the proposed DBMP can effectively predict lane keeping and lane change over a long-term horizon. Moreover, the prediction criteria will greatly increase using proposed DBMP than naive BN. For the future work, we will investigate the motion prediction in other complex environments (e.g., urban intersections). The problem of unstable predictions will be solved in future studies.

6 ACKNOWLEDGE

We would like to show our sincere gratitude to Han-wool Woo from the University of Tokyo. The discussion with him about feature extraction of the open dataset inspires us a lot. The library for Bayesian inference we used is SMILE, which is developed by BayesFusion, <http://www.bayesfusion.com/>.

REFERENCES

- [1] S. Lefèvre, D. Vasquez, and C. Laugier, "A survey on motion prediction and risk assessment for intelligent vehicles," *ROBOMECH Journal*, vol. 1, no. 1, p. 1, 2014.
- [2] T. Gindele, S. Brechtel, and R. Dillmann, "A probabilistic model for estimating driver behaviors and vehicle trajectories in traffic environments A Probabilistic Model for Estimating Driver Behaviors and Vehicle Trajectories in Traffic Environments," in *IEEE International Conference on Intelligent Transportation Systems*, no. May 2014, 2013, pp. 1066–1071.
- [3] M. Althoff and S. Magdici, "Set-Based Prediction of Traffic Participants on Arbitrary Road Networks," *IEEE Transactions on Intelligent Vehicles*, vol. 1, no. 2, pp. 187–202, 2016.
- [4] W. Xu, J. Pan, J. Wei, and J. M. Dolan, "Motion Planning under Uncertainty for On-Road Autonomous Driving," in *2014 IEEE International Conference on Robotics & Automation(ICRA)*, 2014, pp. 2507–2512.
- [5] J. Li, B. Dai, X. Li, C. Li, and Y. Di, "A real-time and predictive trajectory-generation motion planner for autonomous ground vehicles," in *2017 9th International Conference on Intelligent Human-Machine Systems and Cybernetics (IHMSC)*, vol. 2, Aug 2017, pp. 108–113.
- [6] G. S. Aoude, J. Joseph, N. Roy, and J. P. How, "Mobile Agent Trajectory Prediction using Bayesian Non-parametric Reachability Trees," *Proc. of AIAA Infotech Aerospace*, no. March, pp. 1587–1593, 2011.
- [7] G. S. Aoude, B. D. Luders, J. M. Joseph, N. Roy, and J. P. How, "Probabilistically safe motion planning to avoid dynamic obstacles with uncertain motion patterns," *Autonomous Robots*, vol. 35, no. 1, pp. 51–76, 2013.
- [8] D. Kasper, G. Weidl, T. Dang, G. Breuel, A. Tamke, and W. Rosenstiel, "Object-oriented Bayesian networks for detection of lane change maneuvers," *Intelligent Transportation System Magazine*, vol. 4, no. 1, pp. 673–678, 2012.
- [9] J. Schlechtriemen, A. Wedel, J. Hillenbrand, G. Breuel, and K.-d. Kuhnert, "A Lane Change Detection Approach using Feature Ranking with Maximized Predictive Power," no. Iv, 2014.
- [10] S. Sivaraman and M. M. Trivedi, "Dynamic probabilistic drivability maps for lane change and merge driver assistance," *IEEE Transactions on Intelligent Transportation Systems*, vol. 15, no. 5, pp. 2063–2073, 2014.
- [11] M. Bahram, C. Hubmann, A. Lawitzky, M. Aeberhard, and D. Wollherr, "A Combined Model- and Learning-Based Framework for Interaction-Aware Maneuver Prediction," *IEEE Transactions on Intelligent Transportation Systems*, vol. 17, no. 6, pp. 1538–1550, 2016.
- [12] H. Woo, Y. Ji, H. Kono, Y. Tamura, Y. Kuroda, T. Sugano, Y. Yamamoto, A. Yamashita, and H. Asama, "Dynamic potential-model-based feature for lane change prediction," *2016 IEEE International Conference on Systems, Man, and Cybernetics, SMC 2016 - Conference Proceedings*, pp. 838–843, 2017.
- [13] M. Schreier, V. Willert, and J. Adamy, "An Integrated Approach to Maneuver-Based Trajectory Prediction and Criticality Assessment in Arbitrary Road Environments," *IEEE Transactions on Intelligent Transportation Systems*, vol. 17, no. 10, pp. 2751–2766, 2016.
- [14] K. P. Murphy, *Dynamic bayesian networks: representation, inference and learning*. University of California, Berkeley, 2002.
- [15] M. Schreier, "Bayesian Environment Representation, Prediction, and Criticality Assessment for Driver Assistance Systems," Ph.D. dissertation, Technische Universität Darmstadt, 2015.
- [16] H. Woo, Y. Ji, H. Kono, Y. Tamura, Y. Kuroda, T. Sugano, Y. Yamamoto, A. Yamashita, and H. Asama, "Lane-Changing Feature Extraction Using Multisensor Integration," no. Iccas, pp. 3–6, 2016.

- [17] F. H. Administration, “Next generation simulation,” <http://ops.fhwa.dot.gov/trafficanalysisistools/ngsim.htm>.
- [18] M. Montanino and V. Punzo, “Trajectory data reconstruction and simulation-based validation against macroscopic traffic patterns,” *Transportation Research Part B: Methodological*, vol. 80, pp. 82–106, 2015.